



DURBAN UNIVERSITY OF TECHNOLOGY
INYUVESI YASETHEKWINI YEZOBUCHWEPHESHE

FACULTY OF ACCOUNTING AND INFORMATICS
DEPARTMENT OF INFORMATION TECHNOLOGY

2023
YEAR END MAIN EXAMINATION

INSTRUCTIONAL PROGRAMME : Bachelor of ICT/Advanced Diploma in ICT

INSTRUCTIONAL OFFERING : *Machine Intelligence 3*

PAPER NUMBER : 1

SUBJECT CODE/S : MAIN301/MCHI301

DATE : 22 NOVEMBER 2023

DURATION : 3 HOURS

TIME : 14H00 - 17H00

TOTAL MARKS : 100

NUMBER OF PAGES : 11 (including cover sheet)

EXAMINER/S : MRS S HOOSEN

MODERATOR/S : MR L VORSTER

INSTRUCTIONS/REQUIREMENTS:-

1. Answer all questions in the answer book
2. Calculators are permitted
3. No notes will be permitted.

Do not turn the page until permission is given

QUESTION 1 [15 marks]

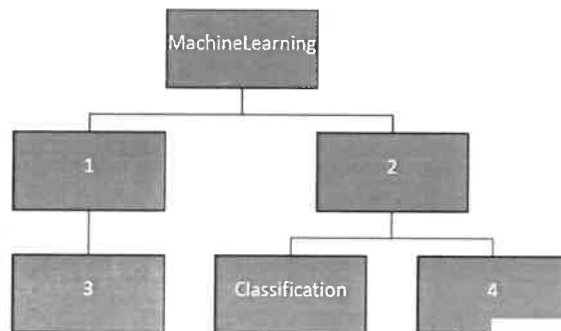
Write down the numbers 1.1 to 1.15 and next to each write down only the alphabet of the choice you have selected.

1.1 What type of learning needs to be applied to the following problem statement?

"You want to train a machine to help you predict how long it will take you to drive home from your workplace."

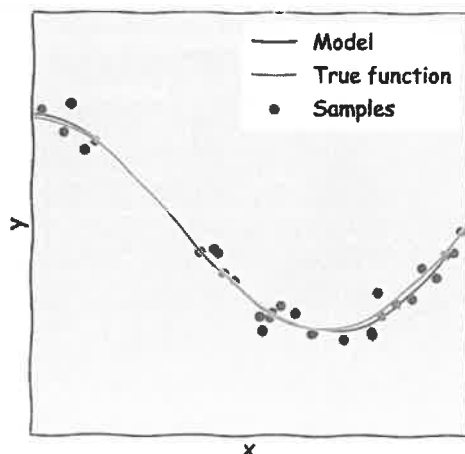
- A) Unsupervised Learning with Clustering
- B) Supervised Learning with Regression
- C) Supervised Learning with Classification
- D) Reinforcement Learning with Agents

1.2 The diagram below represents an overview of the types of machine learning. Study this diagram and then select the correct option by matching the numbers in the diagram with the corresponding type of learning.



- A) 1: Supervised; 2: Unsupervised; 3: Regression; 4: Clustering
- B) 1: Unsupervised; 2: Supervised; 3: Clustering; 4: Regression
- C) 1: Supervised; 2: Unsupervised; 3: Clustering; 4: Regression
- D) 1: Unsupervised; 2: Supervised; 3: Regression; 4: Clustering

1.3 What best describes the model function illustrated below?



- A. This is an example of a model with high bias
- B. The model function has the right complexity to fit the true function
- C. The model function does not have enough complexity to fit the true function and underfits the model
- D. The model function has too many complexities and over fits the model

1.4 In the snippet of code below which method compares features in a dataset and generates a threshold value for the comparisons, and which method locates individual elements in the dataset, respectively.

```
In [134]: def get_correlation(data, threshold):
            corr_column = set()
            corrmatrix = data.corr()
            for i in range(len(corrmatrix.columns)):
                for j in range(i):
                    if abs(corrmatrix.iloc[i, j]) > threshold:
                        colname = corrmatrix.columns[i]
                        corr_column.add(colname)
            return corr_column
```

- A) corrmatrix, i
- B) .corr(), set()
- C) .corr(), .iloc[]
- D) set(), iloc[]

1.5 This type of learning is used to identify different segments of customers and group them into categories like gender, age and location.

- A) Reinforcement learning with agents
- B) Unsupervised learning with clustering
- C) Supervised learning with regression
- D) Supervised learning with classification

1.6 The algorithm that finds the best fit model by automatically selecting the best features in a dataset and handles missing data is _____?

- A) Neural Networks
- B) SVM
- C) Random Forest
- D) Decision Trees

1.7

	A	B	C	D	E	F	G	H	I
Type	Flour	Milk	Sugar	Butter	Eggs	Baking Powder	Vanilla	Salt	
Muffin	55	28	3	7	5	2	0	0	
Muffin	47	24	12	6	9	1	0	0	
Muffin	47	23	18	6	4	1	0	0	
Muffin	45	13	17	17	8	1	0	0	
Muffin	50	25	12	6	5	2	1	0	
Muffin	55	27	3	7	5	2	1	0	
Muffin	54	27	7	5	5	2	0	0	
Muffin	47	26	10	10	4	1	0	0	
Muffin	50	17	17	8	6	1	0	0	
Muffin	50	17	17	11	4	1	0	0	
Cupcake	39	0	26	19	14	1	1	0	
Cupcake	42	21	16	10	8	3	0	0	
Cupcake	34	17	20	20	5	2	1	0	
Cupcake	39	13	17	19	10	1	1	0	
Cupcake	38	15	23	15	8	0	1	0	
Cupcake	42	18	25	9	5	1	0	0	
Cupcake	36	14	21	14	11	2	1	0	
Cupcake	38	15	31	8	6	1	1	0	
Cupcake	36	16	24	12	9	1	1	0	
Cupcake	34	17	23	11	13	0	1	0	

`x_train,x_test,y_train,y_test=train_test_split(X, y, test_size=0.3)`

Given the dataset and the code above, what will be the output for `x_train.shape`, `x_test.shape` ?

- A) (12, 8), (8,)
- B) (14,8) (14,)
- C) (14,) (6, 8)
- D) (14, 8), (6, 8)

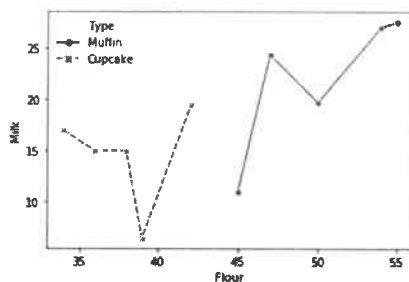
1.8 This module implements several loss, score, and utility functions to measure classification performance of algorithms when applied to data.

- A) sklearn. scores
- B) sklearn. metrics
- C) sklearn.models
- D) sklearn.accuracy

1.9 These types of machine learning algorithms are based on supervised learning and they model target prediction values based on independent variables. They mainly used for finding out the relationship between variables and forecasting.

- A) Decision Trees
- B) Random Forest
- C) Linear Regression
- D) SVM

1.10 What will be the correct coding with parameter listings for the plots below:



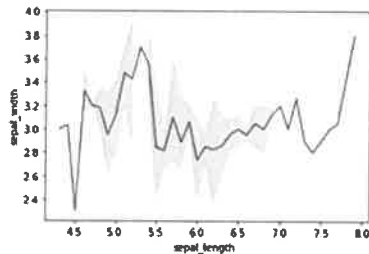
- A) `ax=sns.lineplot(x='Flour',y='Milk',data=raw_data,hue='Type', ci=False, markers=True, style='Type')`
- B) `ax=sns.line(x='Flour',y='Milk',data=raw_data,hue='Type', ci=True, markers=True, style='Type')`
- C) `ax=sns.lineplot(x='Flour',y='Milk',data=raw_data,hue='Type', ci='False', markers=True, 'style'='Type')`
- D) `ax=sns.lineplot(x=Flour,y=Milk,data=raw_data,hue='Type', ci=False, markers=True, style='Type')`

1.11 Given the following code:

```
import seaborn as sns

data = sns.load_dataset("iris")
```

Select the line of code that will plot the graph below:



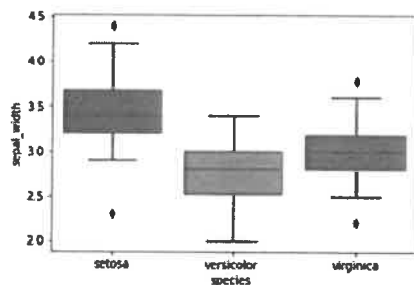
- A. `sns.lineplot(x="sepal_length", y="sepal_width", data=data, hue='Type')`
- B. `sns.lineplot(x="sepal_length", y="sepal_width", data=data, ci="True")`
- C. `sns.lineplot(x="sepal_width", y="sepal_length", data=data)`
- D. `sns.lineplot(x="sepal_length", y="sepal_width", data=data)`

1.12 Given the following code:

```
import seaborn as sns
import matplotlib.pyplot as plt
data = sns.load_dataset("iris")
```

Select the code that will display the plot below:

Output:



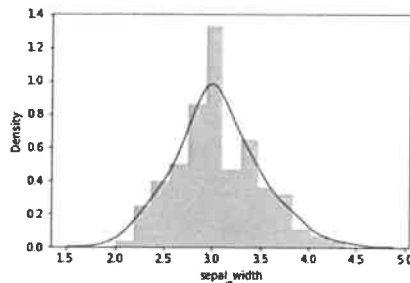
- A) `plt.boxplot(x='species', y='sepal_width', data=data)`
`plt.show()`
- B) `sns.barplot(x=species, y=sepal_width, data=data)`
`plt.show()`
- C) `sns.boxplot(x='species', y='sepal_width', data='data')`
`plt.show()`
- D) `sns.boxplot(x='species', y='sepal_width', data=data)`
`plt.show()`

1.13 Given the code below:

```
import seaborn as sns
import matplotlib.pyplot as plt
data = sns.load_dataset("iris")
```

Select the code that will display the plot below:

Output:

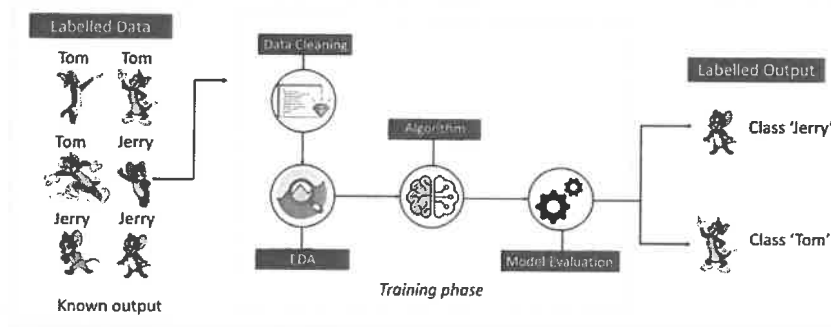
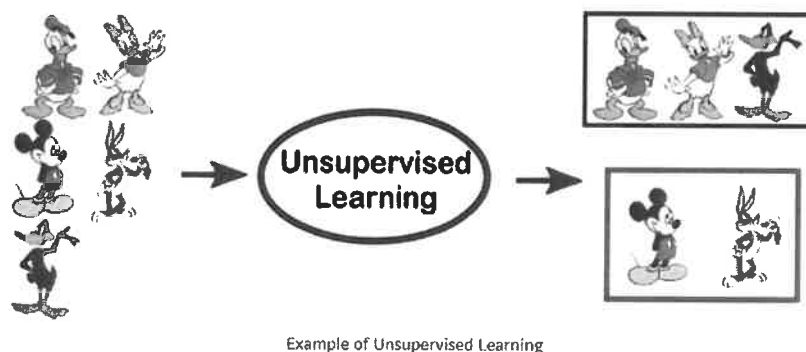


- A) `sns.histplot(data['sepal_width'])`
 - B) `sns.distplot(data['Density'])`
 - C) `sns.distplot(['sepal_width'])`
 - D) `sns.distplot(data['sepal_width'])`
- 1.14 This is a type of bar plot where the X-axis represents the bin ranges while the Y-axis gives information about frequency.
- A) Histogram
 - B) Line Graph
 - C) Scatter Plot
 - D) Bar chart
- 1.15 Colouring different points of a scatter plot can be done through the use of:
- A) `ci`
 - B) `markers`
 - C) `Hue`
 - D) `Matplotlib inline`

(15)

QUESTION 2[20 marks]

2.1. Study the diagrams A and B below and give a clear and detailed explanation of what each illustrates with regards to Machine Learning.

Diagram ADiagram B

Example of Unsupervised Learning

(6)

2.2 **“Ever since the technical revolution, we’ve been generating an immeasurable amount of data. As per research, we generate around 2.5 quintillion bytes of data every single day! It is estimated that by 2020, 1.7MB of data will be created every second for every person on earth.”**

With regards to the statement above, discuss the need and importance of Machine Learning as the most in demand technology in today’s market. (8)

2.3 Data Collection is regarded as a very important stage in the ML workflow process. From your understanding of Machine Learning why does data play such a vital role. (3)

2.4 “Data collected from the real world is transformed to a clean dataset.”

List 3 ways on how one could clean the raw data in a dataset. (3)

QUESTION 3 [15 marks]

Study the dataset given below and answer the questions that follow:

Age	Height (in foot)	Weight (in kgs)
5	3.5	20
7	3.11	25
9	4.1	26
11	4.7	32
13	4.11	35
15	5.1	40
17	5.2	45
19	5.3	48
21	5.5	50
23	5.55	51
25	5.55	55

- 3.1 What can you conclude about the relationship between the 3 feature attributes given and explain the impact this may have on the ML model during the training phase of the ML workflow process. (5)
- 3.2 What statistical method can we implement in ML to help measure the strength of the relationship between 2 variables. (2)
- 3.3 With regards to ML, list 4 methods of cleaning a dataset and for each method explain what is the reason for removing these feature attributes. (8)

QUESTION 4 [10 marks]

Explain the working of each of the following ML algorithms using a diagram as well as the general Python code for implementation.

- 4.1 Linear Regression Algorithm (5)
- 4.2 Support Vector Machine (5)

QUESTION 5 [10 marks]

- 5.1 The code below implements a Decision Tree classifier to train and test a ML model using the muffins vs cupcakes dataset. Complete the code by filling in the missing information.

```
from 5.1.1 import 5.1.2
model= 5.1.3
5.1.4 (ingredients,type_label)
new =[[47, 26, 10, 10, 4, 1, 0, 0]]
predict= 5.1.5 (new)
if(predict)==0:
    print("muffin")
else:
    print("cupcake")
```

(5)

- 5.2 What is your understanding of the algorithm below. Exactly what is it used for in machine learning and how does it work. (5)

```
def get_correlation(data, threshold):
    corr_column=set()
    corrmatrix=data.corr()
    for i in range(len(corrmatrix.columns)):
        for j in range(i):
            if abs(corrmatrix.iloc[i,j])>threshold:
                colname=corrmatrix.columns[i]
                corr_column.add(colname)
    return corr_column
```

QUESTION 6 [15 marks]

- 6.1 Splitting your dataset is essential for an unbiased evaluation of prediction performance. The code below illustrates the application of the train, test, split method, with 60% of data used for training and 40% for testing. The program also uses evaluation measures of precision, accuracy and recall.

Study the incomplete program below and fill in the parts that are missing.

```
from sklearn.model_selection import 6.1.1
from 6.1.2 import accuracy_score, precision_score, 6.1.3
from sklearn.ensemble import 6.1.4
x_train, x_test, y_train, y_test=train_test_split(ingredients, type_label,
6.1.5 =0.4)

model=RandomForestClassifier()
model.fit(x_train, 6.1.6)
predict=model.predict(6.1.7)
accuracy= 6.1.8 (y_test, predict)
precision= 6.1.9 (y_test, predict)
recall= 6.1.10 (y_test, predict)
accuracy, precision, recall
```

(10)

- 6.2 Splitting a dataset might also be important for detecting if your model suffers from one of two very common problems, called under fitting and overfitting. What is your understanding of each problem and explain how it affects the created model? (5)

QUESTION 7 [15 marks]

Study the following segment of Python code on data visualisations and answer the questions that follow.

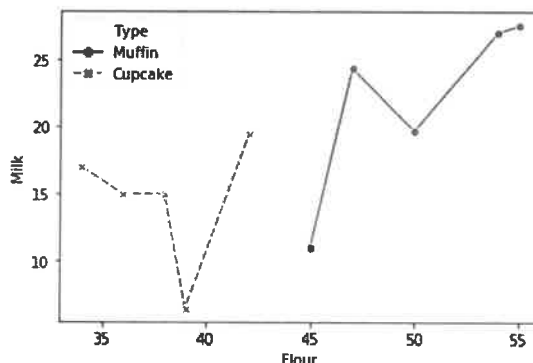
```
In [1]: #Plots and visualisations in ML
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
%matplotlib inline

In [3]: raw_data=pd.read_csv('muffins_cupcakes.csv')
raw_data.head()
```

```
Out[3]:
```

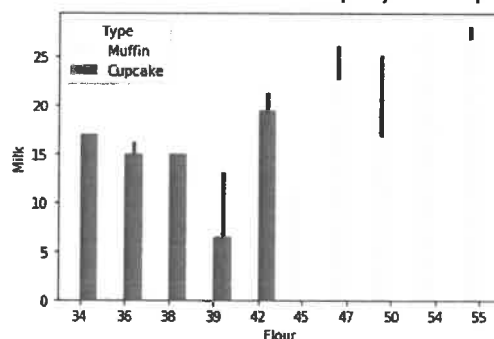
	Type	Flour	Milk	Sugar	Butter	Egg	Baking Powder	Vanilla	Salt
0	Muffin	55	28	3	7	5	2	0	0
1	Muffin	47	24	12	6	9	1	0	0
2	Muffin	47	23	18	6	4	1	0	0
3	Muffin	45	11	17	17	8	1	0	0
4	Muffin	50	25	12	6	5	2	1	0

7.1 Write the code that will display a lineplot as follows:



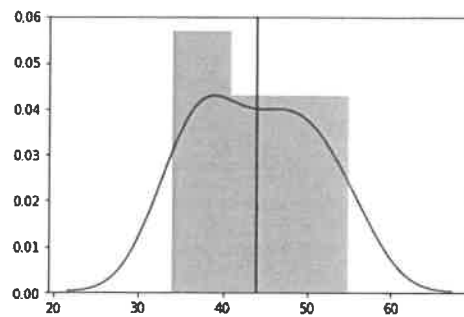
(5)

7.2 Write the code that will display a bar plot, in red, as follows:



(5)

- 7.3 Write the code that will display a histogram as follows. Note that the distribution plot is in red and the vertical mean axes line is in blue.



THE END

TOTAL=100

Research Article

Waste Management System Using IoT-Based Machine Learning in University

Tran Anh Khoa ¹, **Cao Hoang Phuc**,² **Pham Duc Lam** ³, **Le Mai Bao Nhu**,²
Nguyen Minh Trong,² **Nguyen Thi Hoang Phuong**,² **Nguyen Van Dung**,² **Nguyen Tan-Y** ²,
Hoang Nam Nguyen ¹ and **Dang Ngoc Minh Duc** ⁴

¹Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical & Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam

²Faculty of Electrical & Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam

³Faculty of Mechanical, Electrical, Electronic and Automotive Engineering, Nguyen Tat Thanh University, Ho Chi Minh City 700000, Vietnam

⁴School of Graduate Studies, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam

Correspondence should be addressed to Pham Duc Lam; pdlam@ntt.edu.vn

Received 5 September 2019; Revised 2 December 2019; Accepted 19 December 2019; Published 27 February 2020

Academic Editor: Nathalie Mitton

Copyright © 2020 Tran Anh Khoa et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Along with the development of the Internet of Things (IoT), waste management has appeared as a serious issue. Waste management is a daily task in urban areas, which requires a large amount of labour resources and affects natural, budgetary, efficiency, and social aspects. Many approaches have been proposed to optimize waste management, such as using the nearest neighbour search, colony optimization, genetic algorithm, and particle swarm optimization methods. However, the results are still too vague and cannot be applied in real systems, such as in universities or cities. Recently, there has been a trend of combining optimal waste management strategies with low-cost IoT architectures. In this paper, we propose a novel method that vigorously and efficiently achieves waste management by predicting the probability of the waste level in trash bins. By using machine learning and graph theory, the system can optimize the collection of waste with the shortest path. This article presents an investigation case implemented at the real campus of Ton Duc Thang University (Vietnam) to evaluate the performance and practicability of the system's implementation. We examine data transfer on the LoRa module and demonstrate the advantages of the proposed system, which is implemented through a simple circuit designed with low cost, ease of use, and replace ability. Our system saves time by finding the best route in the management of waste collection.

1. Introduction

The Internet of Things (IoT) is a new and promising technology, which has the potential to globally change human life in a positive way, thanks to its diverse connectivity. IoT provides exchange and linkage between low-energy devices and interactions through the Internet. Many applications around the world have been implementing different operations, based on the background of IoT, to offer novel services for smart cities [1–3] and optimize

energetic efficiency. For examples, energy efficiency was considered in [4, 5], cars were connected by IoT methods in [6–8], and water management in smart agriculture was investigated in [9–13], among a long list of other IoT application areas.

One important application is that IoT technology has become a practical and efficient tool to build smart cities. According to [14], a critical issue for a smart city is the increase of waste generation with accelerated population growth in cities. Two significant challenges for waste

management are waste collection and path collection. First, waste collection is a daily task in urban areas entailing the planning of waste truck routes, in which environmental, economic, and social factors must be considered. Second, the path length should be shortened, in order to avoid high fuel costs and reduce the work amount, by applying graph theory [14–17]. Some solutions have introduced the use of IoT devices to estimate the fill level of inboxes and send this data over the Internet to a server for decision-making [18–23].

Machine learning (ML) provides effective solutions, such as regression, classification, clustering, and correlation rules perception [24, 25], for IoT-based waste management. There are three main reasons for this: firstly, in IoT applications, all of the devices are connected, and an immense amount of data is collected every day. Furthermore, they may be programmed to trigger some events, based either on some predefined conditions or exciting feedback from the collected data. Secondly, computer systems can learn to perform certain tasks, such as classification, clustering, predictions, and pattern recognition. Furthermore, these systems are trained using numerous algorithms and statistical models to analyze sample data. Thirdly, measurable characteristics (called features) usually characterize the sample data; some ML algorithms attempt to find correlations between the features and some output values (called labels). The information obtained through training is then used to identify patterns or make decisions based on new data.

However, human intervention is typically required to analyze the collected data, extract meaningful information, and create smart applications. IoT devices must not only collect data and transfer it to other devices but also be self-sufficient. They must be able to make context-based decisions and learn from their collected data [25]. Hence, waste management systems using IoT-based machine learning should consider more variables for task prediction. On the other hand, regression is a mathematical model which can represent or recognize the connection between two or more variables. The dependent or response variable is the system's output. By applying a regression model to a system, the relationships between the dependent variables and the independent ones can be detected. Logistic regression (LR) is a kind of regression analysis used to explain the connection between a dependent variable and one or more independent variables. LR is appropriate when the dependent variable takes binary values. As such, it is suitable for prediction tasks in ML and IoT application [24–29].

IoT-based waste management models perform a vital function in improving the standard of living and human well-being by increasing energy-efficiency, enhancing governance, and reducing cost. In Vietnam, Ton Duc Thang University has set a goal to become an elite research university in the world's top 500 universities. With the successful model and dream of a top Vietnamese university, the application of IoT technology to waste management is one aspect of this model.

This article presents a novel platform for smart trash control at the university, which is able to keep the university

clean at low cost, requiring low labour resources. A novel microcontroller system is designed with a sensor module for measuring the filling height of garbage using ultrasound and geolocation of collected data based on LoRa technology. Furthermore, the paper presents a new method for predicting the probability of the filling level of each trash bin by applying LR in ML; furthermore, a graph theory-based optimization solution is proposed to compute the paths of waste collection with different time schedules in order to minimize the environmental and socio-economic impacts, as well as supporting the workers of the university.

The contributions of our work are:

- (i) Previous articles have mostly evaluated results on a test board; our work introduces the design of a single microcontroller board, which is of low cost and straightforward, with an ultrasound sensor which can measure the filling height of a garbage can and send information using LoRa E32 technology.
- (ii) We present a novel IoT-based machine learning method, which is employed to predict the probability of collecting waste in the real environment based on the historical input data.
- (iii) Our article is the first to propose the use of the sigmoid function for predicting the probability of waste collection and to apply Dijkstra's algorithm to optimize the path for waste collection from trash bins.
- (iv) The critical success of our article is in testing the performance of our system in the realistic environment of Ton Duc Thang University (Vietnam). Therefore, the advantages of employing smart trash bins for the trash collection task can be illustrated. Our algorithm has applied the results to implementing various filling height thresholds for determining the assembly of a garbage can, increasing profits, and optimizing the number of workers to be used.

The paper is organized as follows: Section 2 summarizes the related literature. Section 3 presents all components and hardware design. Section 4 explains the developed algorithm. Section 5 tests the prototype in a real environment to perform different operations. Finally, Section 6 concludes and estimates future works.

2. Related Works

Considering the advantages of IoT technologies, many researchers have investigated and developed new applications for smart cities, especially for waste management. To save power consumption and maximize operational time, a simple system that identifies the fullness of trash bins was presented, which collected data and delivered it through a wireless mesh network [5]. However, the idea has still some ambiguous problems in the system. To improve waste management, platform software for smart cities has been introduced in [14]; however, they only concentrated on the

collection of data, and their platforms were comprised of technologies from other companies.

On the other hand, some approaches have developed waste management strategies based on the optimization to achieve an efficient system. In [15], the authors presented a waste control and management stage to be applied in rural areas using LoRaWAN technology and route optimization. Additionally, an implementation based on IoT was set up, but the system did not provide clarity about communication and optimization for all trash bins in the system. Based on logistic regression (LR) and genetic algorithm (GA) methods, the authors in [16] presented a new method to check the status of smart trash bins and select a collection path in Philadelphia, USA. Moreover, they did not provide any technologies for the transmission of data from the trash bin to the other devices in the system. In particular, optimization algorithms have been clearly defined for IoT-based waste management, such as the nearest neighbour search, colony optimization, genetic algorithm, and particle swarm optimization methods [17, 18]. In [19], the authors proposed a solution to manage a garbage system integrated with IoT technology, which was an autonomous line-following vehicle with a robotic hand for garbage collection, in which they did not apply any algorithms to optimize the waste collection. In [20], an IoT platform for an automated waste collection system provided by the project allowed real-time monitoring and interface with the system. However, the aim of this paper was to present an IoT cloud solution combining device connection, data processing, and control, rather than the design and optimization of waste collection. A food waste collection method was presented in [21], where the information was collected using radio frequency identification (RFID) technology and transmitted using a wireless mesh network. However, the disadvantages of this technology were severe in the long range, especially considering the aim of the smart city is management with a large area. Finally, the results of the optimization algorithm became too vague and could not be applied to a real system, such as a city.

For practical waste management, an impressive architecture was proposed for a sensor node in [22, 23], which used a microcontroller (ATMega328P), an ultrasonic sensor (SRF05), and a LORA E32 TTL—100 433 MHz module [30–32]. Nonetheless, they only tested the test board as a platform to provide sensor nodes and did not apply any methods for waste management in a smart city, such as optimized waste collection. While the target of this article was to propose an IoT application, with their design, the microcontroller board was very complicated; moreover, certain performances and functionalities are needed for each particular application. Our article applies machine learning and graph theory to optimize the waste collection processes, avoid overfilled bins, and reduce the work load.

In this paper, we consider algorithms, based on heuristic models or graph theory, from which we can find ways to minimize the distance of waste collection. The primary purpose is to reduce the total cost of transport, transfer, save labour, and reduce the dependency on used vehicles, while maximizing service quality, as well as improving general

quality of life. The other algorithm optimizes waste collection, instead of considering low cost and energy savings, and supports the university's waste management network to perform with high efficiency by applying ML and graph theory methods. We summarize the features of algorithms considered in this part and compare them with our proposal, as shown in Table 1.

3. System Architecture

The system under consideration consists of smart trash bins with a real-time monitoring system which integrates multichoice, such as ultrasound distance, along with a LoRa E32 TTL-100 433 MHz transmission module. Low energy use was considered throughout the design process. Each node is consequently supposed to be powered by multiple sources; for instance, solar energy or batteries. For flexibility, we have designed hardware that can use either energy source, as trash bins are often put in places where direct sunlight is not available. Selection of the best electronic components for their interconnection and for energy efficiency strategies during the employment of the methods was considered. Furthermore, technical solutions will enable turning off nodes (or parts of them) when they do not work. Once principal components of the node had been chosen, the overall design was targeted at their integration. The proposed system architecture is shown in Figure 1, where the data collected are sent using LoRa to a server, where they are stored and processed. The data are used to track and predict the status of the trash over each period. Furthermore, they will be used to calculate the optimal path, accordingly. The predicted state of each trash bin can be examined, based on assigned training data. It is, then, reviewed to refresh the appropriate waste fill level, which is an essential input parameter of the optimal path algorithm.

Besides these main targets, low-cost containment and high efficiency were secondarily considered. One feature of smart cities is that the requirements of low power consumption imply that there is no connection to the electrical network. The most critical component affecting the energy spread of a sensor node is the transmitting module. So, the design of such architecture must focus on not only the structure of the sensor node, but also the meaning of the system structure. Finally, the description of efficient information transfer technology in large areas should be considered adequately. Contrarily, an IoT node can estimate the volume data of trash bins, and then, the proper sensor to perform this analysis must be determined. Therefore, the selection of a sensor which considers the weight of trash bins was not considered, as a high weight does not imply a high trash level, for the following reasons: first, metallic materials can be weighty, yet leave the trash bin mostly empty, while a large quantity of paper can load the trash bin but weigh little [23]. Second, the authors in [16] recommended that, in some cases, the weight to the trash bin should not be considered, but, instead, the volume data and weight need to be combined. After all of the abovementioned specifications, the best selection solution for the ultrasound sensor was decided to be the SRF05 [30].

TABLE 1: Comparison of the principal characteristics of the proposed algorithms to the state of the art.

Ref	Algorithm	Goal	Hardware design	GIS	Transmission module	Mobile app
[5]	No	Low cost, energy saving	Yes	GPS	Yes	No
[15]	Graph theory	Optimal waste collection	No	GSM/ GPS	Yes	No
[16]	Graph theory	Smart waste collection	Yes	LoRa	Yes	No
[17]	No	Automated waste collection	No	No	No	No
[18]	LR and GA	Optimal waste collection	No	NA	No	No
[19]	Heuristic	Optimal waste collection	No	NA	Yes	No
[20]	No	Management	No	No	No	Yes
[21]	No	Management	No	GSM	Yes	No
[22]	No	Low cost, energy saving	Yes	RFID	Yes	Yes
[23]	No	Low cost, energy saving	No	LoRa	Yes	No
[24]	No	Low cost, energy saving	Yes	LoRa	Yes	Yes
Our proposal	Graph theory and machine learning	Optimal waste collection, low cost, and energy saving	Yes	LoRa	Yes	Yes

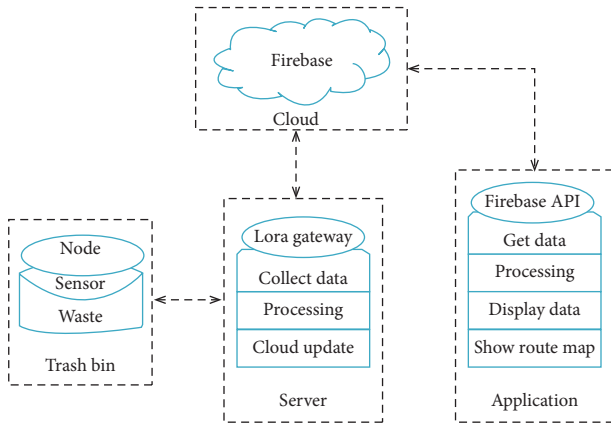


FIGURE 1: Illustration of the overall smart waste collection system.

LoRa is designed to work in the appropriate band for each country or region because of license-free bands in each different country. The LoRa module depending on the physical layer used can be classified as in the 433, 868, or 915 MHz frequency bands [33]. In Asia, only the 433 MHz bands can be used [34]. We used LoRa E32 TTL-100 433 MHz module which aims at providing communication in the range of up to 2 km, low latency, mobility support, multilayer battery life, AES 128-bit shared key encryption support, and a data rate up to 167 kbit/s [35]. The LoRaWAN is defined as the MAC layer protocol and network system architecture based on the LoRa technology. The topology of LoRaWAN network architecture is star, where the end devices can only communicate with LoRaWAN gateways and not directly with each other [33]. The LoRaWAN gateways are used to respond to forwarding raw data packets from end nodes towards the network server. In our network architecture, we use the LoRaWAN MAC layer, class A, that provides the medium access control mechanism that enables the communication between multiple devices and the LoRaWAN gateway.

Based on previous solutions, in our proposal, firstly, the IoT node architecture, shown in Figure 2(a), is composed of three types of components: ATmega328P, LoRa E32 TTL-100 433 MHz module, and SRF-05. Energy specifications of

main active components of the board are listed in Table 2, along with the price (which is below \$20) for each IoT node. Secondly, the communication between the IoT gateway node using LoRa E32 TTL-100 433 MHz and the ESP8266 module is shown in Figure 2(b). Regarding the LoRa module and ESP8266, data collected from the commercial IoT node will be assigned timestamps and sent to the cloud. The progression description in Figure 3 summarizes the main interactions by step-by-step construction. Moreover, the IoT nodes are connected directly to a gateway through the LoRa MAC protocol that forwards their data. Then, the IoT gateways consist of the transmission communication modules, which not only send the data to the server by applying FireBase API/Host for ESP8266 but also send to users/application. The IoT gateway node is responsible for establishing communications with all the nodes in 24 hours. Its hardware requirements are actually simple, since it only has to be able to host and send the data for the server and offer a communication interface with the node. Besides the hardware previously mentioned, the most important is the power system. For flexibility, the system can be chosen to power the nodes with Li-ion 18650 batteries or solar energy, as shown in Table 3, such that the input voltage is within an acceptable range for operating the LoRa E32 TTL-100 433 MHz module.

Considering the capacity of four Li-ion 18650 batteries or a battery and solar panel kit, with settings as in Tables 2 and 3, we can estimate of the lifetime of the batteries as

$$t = \frac{W_t}{W}, \quad (1)$$

where t is the lifetime, W_t is the power throughout the node (W/h or W/day), and W is the total power (W/h or W/day), which is calculated by equation $W = (1/T) (\int_0^{T_1} W_1 + \int_0^{T_2} W_2)$, with W_1, T_1 and W_2, T_2 being power and time duration (s) for the ON and SLEEP mode, respectively. The lifetime approximately corresponds to 39 days for the Li-ion 18650 batteries and 13 days for solar panel and battery kit. Moreover, as the average cost of one node is around \$20, it is very relevant in the university setting.

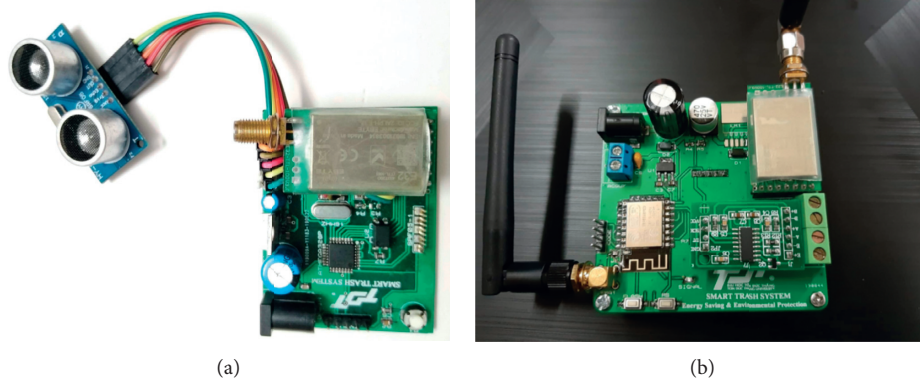


FIGURE 2: (a) The final system architecture of the IoT node. (b) The final system architecture of the LoRa gateway.

TABLE 2: The main active components of the board.

Modules	Amount	Modes	Current capacity (μ A)	Operating voltage (V)	Power (W)	Total power	Running time	Prices (\$)
ATmega328	1	On	16	5	0.08	0.08	3600	2.0
E32 TTL-100	1	On Sleep	106	5	0.53 20	8.853	5 3540	12.6
SRF-05	1	On	4	5	0.02	0.02	3600	3.0

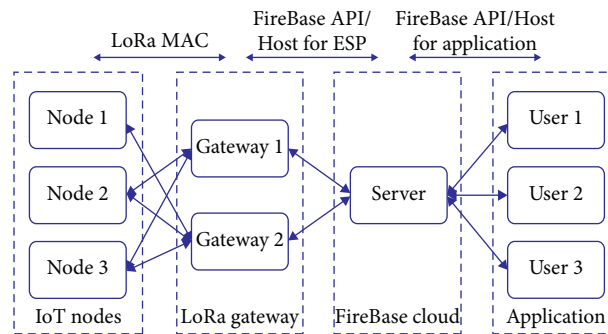


FIGURE 3: Subsystems that form the connection of the IoT node with the application [34, 35].

TABLE 3: The main active power system with the lifetime.

Modules	Amount	Capacity (mAh)	Voltage (V)	Power (W/h)	Lifetime (h)
Li-ion 18650	4	6800	3.7	100.64	931
Solar panel	1	500	20.0	10	
Battery	1	5000	6.8	34	314

After reviewing the components of the system, they were embedded in a board and tested in a real environment to perform different experiments (see the next section).

4. Optimal Path Planning Algorithm for Waste Collection

In this section, we discuss the waste collection data, along with their states, positions, and system design, and test real data to verify the output result. First, the obtained information is transmitted through the communication link to the server, where data are processed, saved, and forwarded to the cloud, as studied in the next section. Second, we apply Algorithm 1 to predict the status and route of each trash bin

daily, which will be explained in Section 4.3. Note that we consider the location of the aggregate waste collection location of each building, instead of each small trash bin on each floor of the university campus; furthermore, the system will enable a node that does not work to be turned off.

4.1. Data Collection. To collect the dataset of waste, we observed several classes for all buildings within working hours (from 6:50 A.M. to 21:00 P.M. from Monday to Sunday) according to the Ton Duc Thang University academic calendar in the second semester of the year 2018-2019 for 4 months. The obtained dataset is shown in Table 4. For instance, on Monday, at building A (node A), there are 91


```

Input:  $\mathcal{F} = \{S, C, W, B, D, \tau, n, \lambda, \eta, nepoches\}$ 
Output:  $\mathcal{O} = \{O, P, L\}$ 
 $ep = 0$ 
while  $ep < nepoches$  do
   $ep += 1$ 
   $\mathbf{w} \leftarrow \mathbf{w} - \eta((\sigma(\mathbf{x}_i^T \mathbf{w}) - y_i)\mathbf{x}_i)$ 
  Sigmoid function  $P$  based on equation (2)
  if  $(\|\mathbf{w}_{new} - \mathbf{w}_{old}\| < tol)$  then
    break
   $\mathbf{w}_{old} = \mathbf{w}$ 
  end if
end while
 $P$  function
for  $i = 1$  to  $n$  do
  Calculate  $P_i$ 
  if  $(P_i \geq \tau)$  then
    Check  $S_i, C_i, L_i, D_i, W_i$ 
    if  $(s_i, l_i \geq \tau)$  then
       $L_i = 1$ 
       $d_i^{min} = Dijkstra(C_i, W_i)$ 
       $O_i \leftarrow d_i$ 
    else if  $(s_i, l_i \leq \tau)$  then
      continue
    end if
  end if
end for

```

ALGORITHM 1: Optimal path planning algorithm for waste collection.

TABLE 4: The classes of Ton Duc Thang University academic year 2018-2019.

Building	A	B	C	D	E	F	G	H	I	J	K	L	M
<i>Classes schedule for the 2018-2019 academic year of Ton Duc Thang university</i>													
Monday	91	234	156	81	252	97	21	7	62	2	23	4	14
Tuesday	95	281	149	120	231	117	21	8	56	0	38	3	24
Wednesday	97	293	174	110	287	134	21	7	48	0	30	0	19
Thursday	99	280	167	91	259	117	21	7	42	2	27	0	29
Friday	90	269	146	95	273	118	21	7	42	2	25	0	23
Saturday	95	165	61	68	226	74	21	8	44	0	15	0	14
Sunday	54	34	8	8	6	1	0	4	0	0	0	0	18
<i>The number of classes each building spent collecting, and whether they collected (1) or not collected (0)</i>													
Monday	1	1	1	1	1	0	0	0	1	0	0	0	0
Tuesday	1	1	1	1	1	0	0	0	0	0	0	0	0
Wednesday	1	1	1	1	1	1	0	0	0	0	0	0	0
Thursday	1	1	1	1	1	1	0	0	0	0	0	0	0
Friday	1	1	1	1	1	1	0	0	0	0	0	0	0
Saturday	1	1	1	1	1	1	0	0	0	0	0	0	0
Sunday	0	0	0	0	0	0	0	0	0	0	0	0	0
<i>The probability of collecting the waste versus the number of classes by applying equation (5)</i>													
Monday	0.91	0.99	0.99	0.87	0.99	0.93	0.24	0.13	0.72	0.10	0.26	0.11	0.18
Tuesday	0.93	0.99	0.99	0.97	0.99	0.97	0.24	0.14	0.65	0.09	0.43	0.10	0.27
Wednesday	0.93	0.99	0.99	0.96	0.99	0.99	0.24	0.13	0.55	0.09	0.33	0.09	0.22
Thursday	0.94	0.99	0.99	0.91	0.99	0.97	0.24	0.13	0.48	0.10	0.30	0.09	0.32
Friday	0.91	0.99	0.99	0.93	0.99	0.97	0.24	0.13	0.48	0.10	0.28	0.09	0.26
Saturday	0.93	0.99	0.70	0.77	0.99	0.82	0.24	0.14	0.50	0.09	0.19	0.09	0.18
Sunday	0.63	0.38	0.14	0.14	0.13	0.10	0.09	0.11	0.09	0.09	0.09	0.09	0.21

classes with study hours. Additionally, the weight of waste is increased in case of more students in classes. We conclude that the dataset of waste is depended by the number of

students. Therefore, we put one large waste bin on the ground floor to handle 13 buildings (nodes) in the university campus area.

4.2. Logistic Regression Model. A description of logistic regression can start with an expression of the regular logistic function used for classification problems. It is a predictive analysis algorithm, based on the idea of probability. The logistic function is a *sigmoid* function, with $t \in \mathcal{R}$ and an output between (0,1) for any value. The function, called $f(s)$, is given by

$$f(s) = \frac{1}{1 + e^{-s}} \triangleq \sigma(s). \quad (2)$$

In this subsection, Figure 4(a) describes the relationship between classes and the status of the trash bin. The green line indicates the direction of the trash bin's level. Taking into consideration the classes, if the green line is near to the state 1, that level of the trash bin will update to 1. Figure 4(b) displays the key algorithm employed in this article, logistic regression.

The primary reason we decided to use this model is due to its excellent characteristics:

- (i) The yellow line is performing linear regression, which fails to represent the true state as it could have a value greater than 1 or less than 0, which is not possible, as per the hypothesis of logistic regression. This is a not sensible choice for our solution.
- (ii) The red line (naturally separate from the perceptron learning algorithm's (PLA's) activation function, in which the two classes are 0 and 1, instead of -1 and 1) is the hard threshold. The PLA does not work in this problem, as the data is not linearly separable.
- (iii) The blue and green lines fit our problem better and have some of the following essential properties:
 - (a) The function is a continuous real value, with range (0,1).
 - (b) Assuming that the point considered is equal to 0.5 as the threshold, the closer the point is to the left, the closer it is to 0; and the closer the point is to the right, the closer it is to 1. This is in accordance with the observation that the more the classes are, the more the students will be able to collect waste and vice versa.

4.3. Optimal Path Planning Algorithm for Waste Collection. The collection paths are the moving sequences, including all of the trash bins within the Tan Phong campus, Ton Duc Thang University. The optimization of specific sequences is a combinatorial optimization problem. Considering several paths, we use Dijkstra's algorithm [36], which is active in implementing near-optimal clarifications. As the employees need time to check all of the trash bin, it is very joyful if the percentage filling level of waste is predicted. Therefore, the next prediction, the system is going to suggest which one should be collected to check the overload appearance. In this article, the LR algorithm is used to predict the status of each trash bin, based on its actual data. The overall system is presented in Algorithm 1, and the settings are described in Table 5. For each hour t in a working day, an LR algorithm is applied to predict the status of each bin. If the probability

status of each trash bin is higher than a given threshold τ , the filling height of the trash bin is checked by the ultrasonic sensor. If the filling level is higher than 50%, the status is updated as 1. Then, Dijkstra's algorithm is applied to find the shortest distance for collecting waste from all full trash bins, turning back to the offices. Consequently, the system detects the optimal paths within the university, which helps the employees to get the collected waste more efficiently.

5. Operation Tests

Our proposal was tested at Ton Duc Thang University with three following considerations. First, we provide and test the algorithm to find the shortest distance in the simulation. Second, we show that these coefficients are entered into the logistic regression equation to predict the probability of collecting the waste in the trash bins. Finally, we design a mobile application to show the effectiveness of the sensor for analysis of the filling level with the data by a wireless communication link. The mobile application also shows the probability prediction and optimal path collection.

5.1. Testing of the Algorithm to Find the Shortest Distance. To generate the nodes as uniformly-distributed points, the three steps mentioned below are used, as generating points using a distance uniformly distributed between 0 and R , and using an angle with uniform distribution from 0 to 2π causes the points to be more dense closer to the origin.

Step 1. Generate a random position inside the circle with radius R (where R is the coverage of the LoRa channel and the co-ordinate of the center is $(0,0)$) with polar co-ordinates $(r_{\text{Node}}, \theta_{\text{Node}})$.

$$\begin{aligned} \theta_{\text{Node}} &= 2\pi \times \text{rand}(), \\ r_{\text{Node}} &= R \times \sqrt{\text{rand}()}, \end{aligned} \quad (3)$$

where $\text{rand}()$ denotes a uniformly distributed random variable from 0 to 1.

Step 2. The point with polar co-ordinates $(r_{\text{Node}}, \theta_{\text{Node}})$ is converted to Cartesian co-ordinates by

$$\begin{aligned} x &= r_{\text{Node}} \times \cos(\theta_{\text{Node}}), \\ y &= r_{\text{Node}} \times \sin(\theta_{\text{Node}}). \end{aligned} \quad (4)$$

Step 3. Go to Step 1 to define the next node position.

After the distributed node positions were generated, the optimal routes are found using Dijkstra's algorithm, based on the filling level and co-ordinates of trash bins. To understanding the operation of Dijkstra's algorithm, we give an example under the assumption in Table 6 with 6 nodes and 11 edges. The considering numbers are distances between two sensor nodes. The relationship between sensor nodes is drawn in Figure 5. Consider that the source node is node 1 and the destination node is node 2, as shown in Figure 5. There are any ways from node 1 to node 2. However, under minimizing the distance path between node 1 and node 2,

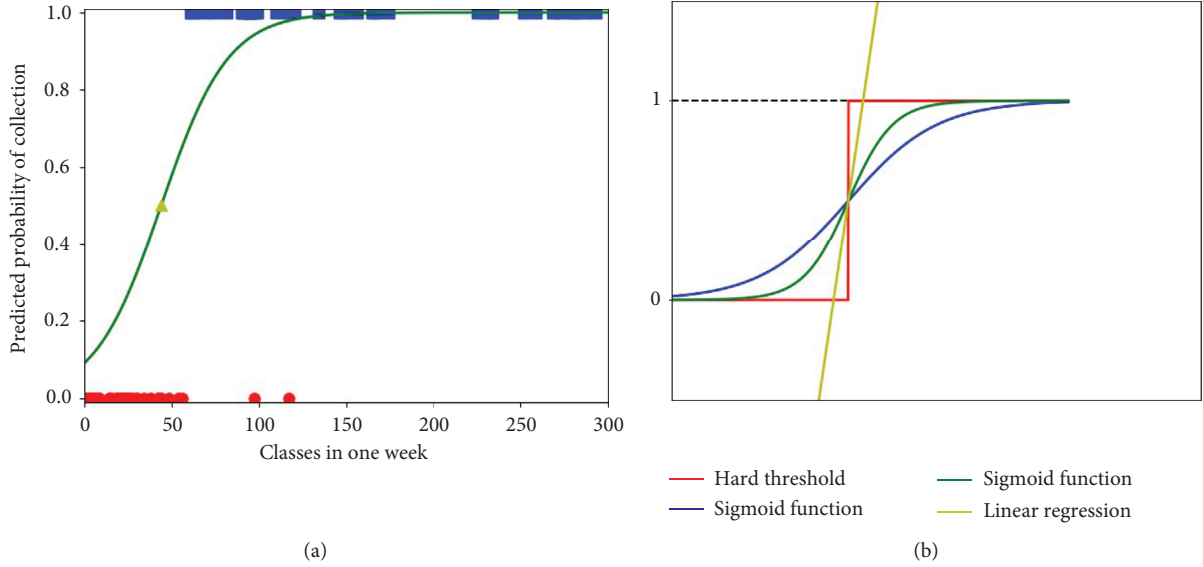


FIGURE 4: (a) Graph of a logistic regression curve showing the probability of collecting waste versus classes in each building over one week. (b) A type of activation function for logistic regression.

TABLE 5: Simulation/numerical parameters.

Variables	Description
$S = \{s_i i \in (1, n)\}$	Status (HIGH, MIDDLE, and LOW) level of trash bin i
$C = \{c_i i \in (1, n)\}$	Co-ordinate of trash bin i
$L = \{l_i i \in (1, n)\}$	Filling level of trash bin i
$W = \{W_i i \in (1, n)\}$	Number of classes i in a day
$O = \{O_i i \in (1, n)\}$	Optimal path of trash bin i in the system
$D = \{D_i i \in (1, n)\}$	Data set from Table 4
$\mathcal{S} = \{S, C, L, W, \tau, \eta, n\}$	Input data
$O = \{O, P, L\}$	Output data
$P = \{p_i i \in (1, n)\}$	Predicted probability of collection waste, P function
τ	Threshold of trash bin i , (0.5)
n	Number of trash bins in the university
η	Learning rate (0.001)
$nepochs$	Number of epochs (10000)
d_i^{\min}	Shortest distance (m)

the shortest path from node 1 to node 2 is $1 \rightarrow 6 \rightarrow 2$. To achieve the fast path, Dijkstra's algorithm is used. The operation is shown in Table 7.

5.2. Predicted Probabilities of Each Node. The considered problem is to build a model to assess the ability of garbage collection, based on the number of classes used during the day in different buildings. The relationship between the number of classes per day at each building in the university campus is depicted in Table 4. From the table, it can be seen that the more the classes were used, the higher the number of students was and, consequently, the more likely it was that the accumulated waste would lead to increased waste collection, although there were some cases where the class was abundant and the waste was below the threshold. Nonetheless, there was no threshold on the number of classes which accurately distinguished waste collection, and so, the threshold was set as

0.5. Therefore, we can predict the probability of waste collection based on the number of classes.

Considering Table 4, the number of classes in each building had a corresponding status for the trash bins as *collected* (1) or *not collected* (0). The LR model was chosen as the predictive modelling algorithm to be applied, as the output variable was a binary classification; that is, the status of the trash bin: *collected* (1) or *not collected* (0). We aimed to define a mathematical equation which can be utilized to predict the probability of the case *collected* (1). Once the equation is estimated, it can be applied to predict the output variables when only the data are known. The *sigmoid* function is shown in equation (2).

Applying equations (2) and (A.11) in Appendix A, the solution of the logistic regression was $(-2.20619801 \ 0.05095177)$. The output indicates that classes are significantly associated with the probability of collecting the waste:

$$P \approx \text{sigmoid}(-2.20619801 + 0.05095177) \times \text{classes}. \quad (5)$$

Now, we apply equation (5) to Table 4. For example, for a building A, which had 91 classes on Monday, setting the value in equation (5) gives an estimated probability for waste collection of 0.91. Similarly, for building I, which had 62 classes, the estimated probability of waste collection was 0.72. Another example is building F, which had 117 classes on Tuesday, for which the estimated probability of waste collection was 0.97. In general, the higher the number of classes, the higher the chance of collecting waste. Equation (5) also indicates that the probability of waste collection increases as the number of classes increases. As *sigmoid* is a covariate function, Table 4 shows the probability of collecting waste for several values of numbers of classes. Based on these predicted probability values in Table 4, the threshold and false positive rate (FPR)/true positive rate (TPR) of the Receiver Operating Characteristic (ROC) curve are estimated in Table 8.

TABLE 6: Distances between all the nodes.

Nodes	(4,1)	(6,2)	(2,3)	(5,3)	(6,3)	(3,4)	(5,4)	(1,5)	(2,5)	(1,6)	(4,6)
Distance (m)	368.2	1619	801	817.2	1258.4	602.3	215.4	582.3	1190.5	1010.6	661.3

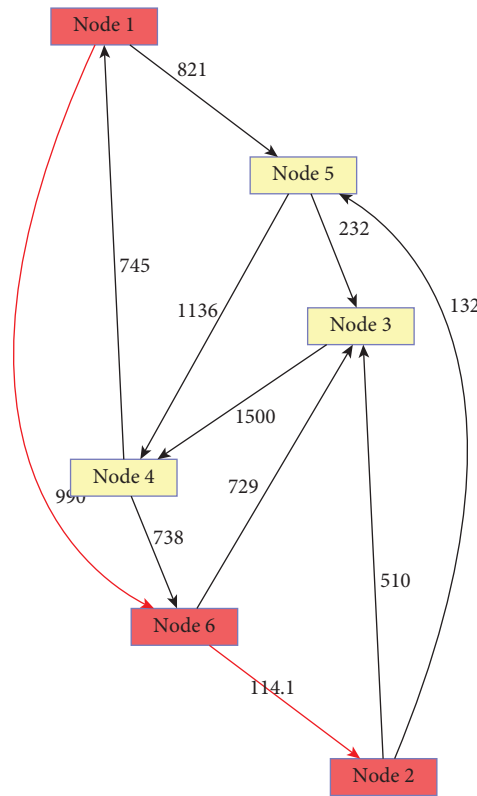


FIGURE 5: The optimal path router of an IoT node.

TABLE 7: Operation of Dijkstra's algorithm.

	Node 1	Node 4	Node 5	Node 6	Node 3	Node 2
Node 1	0	745	821	996	∞	∞
Node 4	0	745	1483	∞	∞	∞
Node 5	0	∞	821	996	∞	∞
Node 6	0	∞	821	1053	∞	∞
Node 3	0	∞	∞	996	1725	1110.1
Node 2	0	∞	∞	996	∞	1110.1

Figure 6(a) shows that the Area Under the Curve (AUC) was 0.99, using threshold = 0.5, FPR = 0.1, and TPR = 1; see Table 8. The closer the ROC curve was to 1 and the larger the AUC was, the more effective the model was. For each threshold value in 7, we thus get a pair (FPR, TPR), which represents points (FPR, TPR) on the graph for which the limit changes from 0 to 1. Note that the range of thresholds does not necessarily range from 0 to 1 in general problems, and it should be ensured that the TPR/FPR receives the largest or smallest value it can achieve. A model is effective when there is a low FPR, and a high TPR means that there exists a point on the ROC curve close to the point with coordinates (0, 1) on the graph (upper left corner).

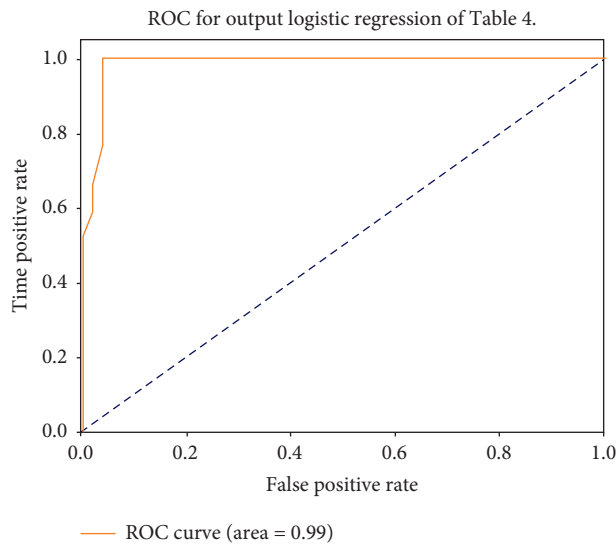
The graph in Figure 6(b) shows the loss function; if the values of the loss function are small, then the evaluation provides useful results. In a functional classification problem,

it is understandable that a few data points are misclassified. Our results in the figure are, thus, consistent with the loss function model. The loss function and its optimization are presented in Appendix A.1 and A.2, respectively.

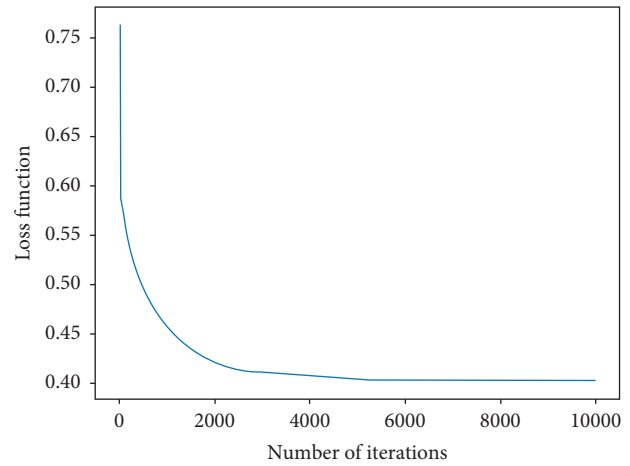
5.3. Test in a Real Environment. This section explains the network structure that was designed and implemented in the actual campus at Ton Duc Thang University over an area of 30 hectares with buildings (see Figure 7). The number of classes of each building is described in Table 4. In each building, a centralized waste storage area was placed with a gateway and data collection. The authentic structure had three levels for each trash bin: HIGH, MIDDLE, and LOW. As the threshold was 0.5, if the trash bin's level is HIGH (alternatively, MIDDLE), its filling level is 1; otherwise, it is 0. Based on the status, the employees require updating the optimal paths every day toward getting the high-level bins. By applying Algorithm 1, based on the input parameters, the system can determine the probability of waste collection based on the logistic regression function and optimize the choice of moving between buildings within the campus. Figure 8 depicts the results of class data for Monday on a week of the second semester of the 2018-2019 academic year. We can also determine the path for waste collection with the optimal router. The color set for the percentages in the

TABLE 8: Outcome of the model for varying thresholds.

No.	Thresholds	False positive rate	True positive Rate
1	1.99	0	0
2	0.99	0	0.5
3	0.97	0.01	0.6
4	0.94	0.01	0.65
5	0.93	0.03	0.77
6	0.91	0.03	0.85
7	0.7	0.03	1.0
8	0.5	0.1	1.0
9	0.48	0.14	1.0
10	0.27	0.26	1.0
11	0.26	0.3	1.0
12	0.24	0.41	1.0
13	0.19	0.46	1.0
14	0.18	0.5	1.0
15	0.14	0.58	1.0
16	0.13	0.66	1.0
17	0.11	0.69	1.0
18	0.1	0.78	1.0
19	0.09	1.0	1.0



(a)



(b)

FIGURE 6: The ROC metric to evaluate the classifier output and value of the loss function.



FIGURE 7: Map of the area of Ton Duc Thang University with road map and distance between all of the buildings.

mobile application in Figure 8 is as follows: 0–20% for place-green, 21–40% for lime, 41–60% for yellow-green, 61–80% for ochre, and 81–100% for rust.

6. Conclusions and Future Works

In this work, an optimal algorithm combining graph theory and LR has been described, with the possibility of assessing the probability of a trash bin being fully based on the number of classes in the university. This algorithm presents many advantages, as compared with the old waste collection methods. In addition, this study also provides improvements over

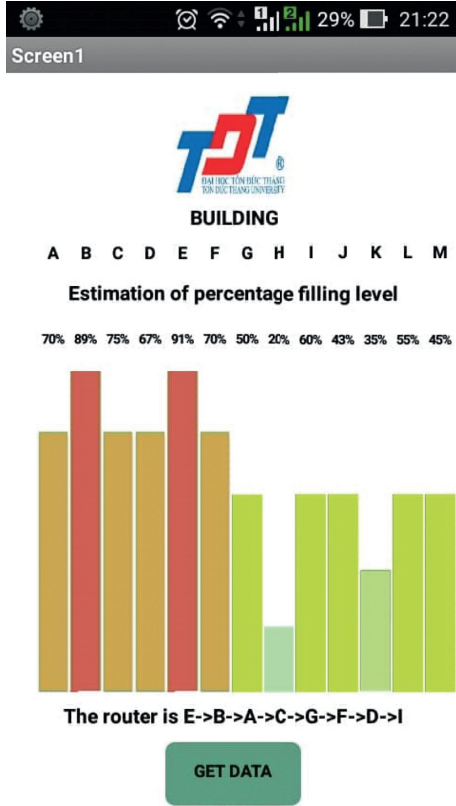


FIGURE 8: The percentage of the trash bins in Ton Duc Thang University.

classical algorithms. The algorithm is integrated into the system with a low-cost design circuit and LoRa technology, enabling its application in practical use-cases, in which changing the sensor components can be done quickly. This study also presented three experiments: first, a test was conducted by simulating the location of arbitrary trash bins and finding the shortest path between the trash bins. Second, the logistic regression equation was applied to estimate the probability of collecting the waste. Finally, a practical-use case of the waste collection process at Ton Duc Thang University (Vietnam) was tested. The algorithm used the database of classes used in each university building and data received from smart buckets (e.g., occupancy rate) as input data. In summary, this system provides better operations for optimizing employee use, saving operating costs, and collecting data on time. The system can be built cheaply, simply, and effectively and will be extensively applied in all campuses of the Ton Duc Thang University, a Smart University, to unify the filling level of the trash bins by using an ultrasonic sensor. Again, machine learning will be used, in particular, using multiclassification methods.

Abbreviations

IoT:	Internet of Things
WSN:	Wireless sensor network
ML:	Machine learning
AI:	Artificial intelligence
GA:	Genetic algorithm
LoRa:	Long range radio third-generation technology

LR:	Logistic regression
LPWAN:	Low power wide area network
PLA:	Perceptron learning algorithm
SGD:	Stochastic gradient descent
FPR:	False positive rate
TPR:	True positive rate
ROC:	Receiver operating characteristic
AUC:	Area under the curve.

Appendix

A. The Logistic Sigmoid Regression Model

A.1. Loss Function. With the model as an activation function $f(s) \in (0, 1)$, assume that the probability that the data x_i has the first label $f(x_i^T w)$ and the last label is $1 - f(x_i^T w)$. Then,

$$p(y_i = 1 | x_i; w) = f(x_i^T w), \quad (\text{A.1})$$

$$p(y_i = 0 | x_i; w) = 1 - f(x_i^T w), \quad (\text{A.2})$$

where $p(y_i = 1 | x_i; w)$ represents the probability that the first label $y_i = 1$ occurs, as we know the parameter model w and the data input x_i . The aim of this purpose is to find the w parameter $f(x_i^T w) \approx y_i$ using the training data.

Denoting $a_i = f(x_i^T w)$, equations (5) and (A.1) can be written more compactly as

$$p(y_i | x_i; w) = a_i^{y_i} (1 - a_i)^{1 - y_i}. \quad (\text{A.3})$$

This equation is equivalent to equations (5) and (A.1); since $y_i = 0$, the first component is 1. To make the model fit the data, we need to find w such that the probability is to the maximum.

Consider all training data with the data matrix $X = [x_1, x_2, \dots, x_N] \in \mathbb{R}^{d \times N}$ and with the corresponding label equivalent $y = [y_1, y_2, \dots, y_N]$. Then, we need to solve the optimization

$$w = \arg \max_w p(y | X; w). \quad (\text{A.4})$$

Assuming that the training data were generated independently, we can write the likelihoods of the parameters as

$$p(y | X; w) = \prod_{i=1}^N p(y_i | x_i; w) = \prod_{i=1}^N a_i^{y_i} (1 - a_i)^{1 - y_i}. \quad (\text{A.5})$$

Note that a_i is an equation in w and x_i . This equation is the loss function of the logistic regression. As the sign changes after taking a logarithm, we need to find the minimum value $J(w)$ of w .

$$J(w) = -\frac{1}{N} \log p(y | X; w) = -\frac{1}{N} \sum_{i=1}^N (y_i \log a_i + (1 - y_i) \log (1 - a_i)). \quad (\text{A.6})$$

A.2. Optimizing the Loss Function. The problem of optimizing the loss function in logistic regression can be solved using SGD. For each loop, w will be updated to one random variable point. The loss function of logistic regression with one point (x_i, y_i) and gradient is

$$J(w; x_i, y_i) = -(y_i \log a_i + (1 - y_i) \log(1 - a_i)), \quad (\text{A.7})$$

$$\nabla_w J(w; x_i, y_i) = -\left(\frac{y_i}{a_i} - \frac{1 - y_i}{1 - a_i}\right) (\nabla_w a_i) = \frac{a_i - y_i}{a_i(1 - a_i)} (\nabla_w a_i). \quad (\text{A.8})$$

Set $x_i^T(w)$. Then, we have

$$\nabla_w a_i = \frac{\delta a_i}{\delta z_i} (\nabla_w a_i) = \frac{\delta a_i}{\delta z_i} x_i. \quad (\text{A.9})$$

Assume that we need to find the function $a = f(z)$. As

$$\frac{\delta a}{\delta z} = a(1 - a), \quad (\text{A.10})$$

equation (A.9) is equivalent to that shown on the next page, where C is a constant. Choosing $C = 0$, we have $a = f(w^T w) = \sigma(z)$. This is a *sigmoid* function. Logistic regression with a sigmoid function is named logistic sigmoid regression or logistic sigmoid.

Thus, substituting (A.8) and (A.9) into (A.7), we have

$$\nabla_w J(w; x_i, y_i) = (a_i - y_i) x_i = (\sigma(x_i^T w - y_i)) x_i. \quad (\text{A.11})$$

Based on Stochastic Gradient Descent (SGD), the root of the logistic regression is

$$w \leftarrow w - \eta(a_i - y_i) x_i = w - \eta(\sigma(x_i^T w - y_i)) x_i, \quad (\text{A.12})$$

where η is the learning rate.

This completes the proof.

$$\begin{aligned} \frac{\delta a}{a(1 - a)} &= \delta z, \\ \left(\frac{1}{a} + \frac{1}{1 - a}\right) \delta a &= \delta z, \\ \log a - \log(1 - a) &= z + C, \\ \log\left(\frac{a}{1 - a}\right) &= z + C, \\ \frac{a}{1 - a} &= e^{z+C}, \\ \frac{a}{1 - a} &= e^{z+C}, \\ a &= e^{z+C} (1 - a), \\ a &= \frac{e^{z+C}}{1 + e^{z+C}} = \frac{1}{1 + e^{-z-C}} = \sigma(z + C). \end{aligned} \quad (\text{A.13})$$

Data Availability

(1) The nature of the data is collected by TDT university campus according to the number of classes and their

corresponding days. (2) The survey data used to support the findings of this study are included within the supplementary information file(s). (3) There are no restrictions on data access. Requests for data, 6 months after publication of this article, will be considered by the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] B. N. Silva, M. Khan, and K. Han, "Towards sustainable smart cities: a review of trends, architectures, components, and open challenges in smart cities," *Sustainable Cities and Society*, vol. 38, pp. 697–713, 2018.
- [2] M. Alvarez-Campana, G. López, E. Vázquez, A. Villagrà, and J. Berrocal, "Smart CEI Moncloa: an IoT-based platform for people flow and environmental monitoring on a Smart University Campus," *Sensors*, vol. 17, no. 12, p. 2586, 2017.
- [3] A. Bagula, L. Castelli, and M. Zennaro, "On the design of smart parking networks in the smart cities: an optimal sensor placement model," *Sensors*, vol. 15, no. 7, pp. 15443–15467, 2015.
- [4] E. Vinagre, F. De Paz, T. Pinto, Z. Vale, M. Corchado, and O. Garcia, "Intelligent energy forecasting based on the correlation between solar radiation and consumption patterns," in *Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, Athens, Greece, December 2016.
- [5] F. Folianto, Y. Low, and W. Yeow, "Smart waste management system," in *Proceedings of the IEEE 10th International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, Singapore, April 2015.
- [6] T. Dinh, Y. Kim, and W. Yeow, "A comprehensive system for a novel location-centric IoT-cloud based on-street car parking violation management system in smart cities," *Sensors*, vol. 16, pp. 1–23, 2016.
- [7] R. Sukjaimuk, Q. N. Nguyen, and T. Sato, "A smart congestion control mechanism for the green IoT sensor-enabled information-centric networking," *Sensors*, vol. 18, no. 9, p. 2889, 2018.
- [8] S. Ahn and J. Choi, "Internet of vehicles and cost-effective traffic signal control," *Sensors*, vol. 19, no. 6, p. 1275, 2019.
- [9] M. S. Munir, I. S. Bajwa, M. A. Naeem, and B. Ramzan, "Design and implementation of an IoT system for smart energy consumption and smart irrigation in tunnel farming," *Energies*, vol. 11, no. 12, p. 3427, 2018.
- [10] D. Bandur, B. Jakšić, M. Bandur, and S. Jović, "An analysis of energy efficiency in wireless sensor networks (WSNs) applied in smart agriculture," *Computers and Electronics in Agriculture*, vol. 156, pp. 500–507, 2019.
- [11] J. Muangprathub, N. Boonnam, S. Kajornkasirat, N. Lekbangpong, A. Wanichsombat, and P. Nillaor, "IoT and agriculture data analysis for smart farm," *Computers and Electronics in Agriculture*, vol. 156, pp. 467–474, 2019.
- [12] A. Goap, D. Sharma, A. Shukla, and C. Krishna, "An IoT based smart irrigation management system using machine learning and open source technologies," *Computers and Electronics in Agriculture*, vol. 155, pp. 41–49, 2018.
- [13] T. Anh Khoa, M. M. Man, T.-Y. Nguyen, V. Nguyen, and N. Hoang Nam, "Smart agriculture using IoT multi-sensor: a novel watering management system," *Journal of Sensor and Actuator Networks*, vol. 8, no. 3, p. 45, 2019.

- [14] K. Pardini, J. Rodrigues, S. A. Kozlov, N. Kumar, and V. Furtado, "IoT-based solid waste management solutions: a survey," *Journal of Sensor and Actuator Networks*, vol. 8, no. 1, p. 5, 2019.
- [15] M.-V. Bueno-Delgado, J.-L. Romero-Gázquez, P. Jiménez, and P. Pavón-Mariño, "Optimal path planning for selective waste collection in smart cities," *Sensors*, vol. 19, no. 9, p. 1973, 2019.
- [16] L. Álvaro, J. Caridad, J. De Paz, G. V. González, and J. Bajo, "Smart waste collection system with low consumption LoRaWAN nodes and route optimization," *Sensors*, vol. 18, no. 5, pp. 1804–1282, 2018.
- [17] D. Vu and G. Kaddoum, "A waste city management system for smart cities applications," in *Proceedings of the 2017 IEEE Advances in Wireless and Optical Communications (RTUWO)*, IEEE, Riga, Latvia, November 2017.
- [18] M. Akhtar, M. A. Hannan, R. A. Begum, H. Basri, and E. Scavino, "Backtracking search algorithm in CVRP models for efficient solid waste collection and route optimization," *Waste Management*, vol. 61, pp. 117–128, 2017.
- [19] M. Hannan, M. Akhtar, R. A. Begum, H. Basri, A. Hussain, and E. Scavino, "Capacitated vehicle-routing problem model for scheduled solid waste collection and route optimization using PSO algorithm," *Waste Management*, vol. 71, pp. 31–41, 2018.
- [20] A. A. J. Jim, R. Kadir, M. A. A. Mamun, A.-A. Nahid, and M. Y. Ali, "A noble proposal for Internet of garbage bins (IoGB)," *Smart Cities*, vol. 2, no. 2, pp. 214–229, 2019.
- [21] C. Popa, G. Carutasu, C. Cotet, N. Carutasu, and T. Dobrescu, "Smart city platform development for an automated waste collection system," *Sustainability*, vol. 9, no. 11, p. 2064, 2017.
- [22] I. Hong, S. Park, B. Lee, J. Lee, D. Jeong, and S. Park, "IoT-based smart garbage system for efficient food waste management," *The Scientific World Journal*, vol. 2014, Article ID 646953, 13 pages, 2014.
- [23] A. Lundin, A. Ozkil, and J. Schuldt-Jensen, "Smart cities: a case study in waste monitoring and management," in *Proceedings of the 50th Hawaii International Conference on System Sciences*, Waikoloa, HI, USA, January 2017.
- [24] M. Cerchecci, F. Luti, A. Mecocci, S. Parrino, G. Peruzzi, and A. Pozzebon, "A low power IoT sensor node architecture for waste management within smart cities context," *Sensors*, vol. 18, no. 4, p. 1282, 2018.
- [25] V. H. Tiep, "Machine learning cơ bản (The Fundamental of Machine Learning)," 2019, <https://machinelearningcoban.com/>.
- [26] F. Zantalis, F. Koulouras, S. Karabetsos, and D. Kandris, "A review of machine learning and IoT in smart transportation," *Future Internet*, vol. 11, no. 4, p. 94, 2019.
- [27] I. A. Najm, A. K. Hamoud, J. Lloret, and I. Bosch, "Machine learning prediction approach to enhance congestion control in 5G IoT environment," *Electronics*, vol. 8, no. 6, p. 607, 2019.
- [28] A. L. Diedrichs, F. Bromberg, D. Dujovne, K. Brun-Laguna, and T. Watteyne, "Prediction of frost events using machine learning and IoT sensing devices," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4589–4597, 2018.
- [29] M. Hasan, M. Islam Milon, M. I. I. Zarif, and M. M. A. Hashem, "Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches," *Internet of Things*, vol. 7, Article ID 100059, 2019.
- [30] "Microchip. ATmega328 complete datasheet," 2018, <https://www.microchip.com/wwwproducts/en/ATmega328>.
- [31] "SRF05—ultra-sonic ranger technical specification," 2019, <https://www.robot-electronics.co.uk/htm/srf05tech.htm>.
- [32] LoRa E32-TTL-1W, "Module, Ebyte SX1278 wireless module E32 series, user manual," 2017, https://img.filipeflop.com/files/download/E32_User+Manual_EN_v1.00.pdf.
- [33] J. Haxhibeqiri, F. Van den Abeele, I. Moerman, and J. Hoebeke, "LoRa scalability: a simulation model based on interference measurements," *Sensors*, vol. 17, no. 6, p. 1193, 2017.
- [34] LoRaWAN, "What is it? A technical overview of LoRa and LoRaWAN," 2019, <https://lora-alliance.org/>.
- [35] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of LoRa: long range & low power networks for the internet of things," *Sensors*, vol. 16, no. 9, p. 1466, 2016.
- [36] "Dijkstra's algorithm," 2019, https://en.wikipedia.org/wiki/Dijkstra_algorithm.