E-DORM

Jou-Chi Huang,

Tamkang University

Group Members:

Jou-Chi Huang, Feng-Hsuan Chang,

Hsing-Hui Lin, Shao-Chun Weng,

Liao-Chung Hao, Chien-Yi Cheng

Contents

Introduction

1. Purpose of The Investigation

The goal of E-Dorm mobile APP is to make dormitory life more convenient and manageable. Accordingly, E-Dorm includes complete functions for students and administrators. Though existing community and dormitory management APPs cover a wide range of functions, there are still many inconveniences in school dormitory lives. For example, long queues waiting for public machines, paper-delivered notifications of package are easily ignored and miss garbage truck. As a result, students often encounter problems such as competing for shared machines, long waiting in their spare time, package being returned and garbage accumulation. To be more precise, we differentiated all the existing problems into three categories which are washing machine, package signing process and garbage truck reminders respectively. We hoped to eliminate the above problems by efficient and eco-friendly means. Moreover, we planned to adapt inexpensive and accessible equipment so that dormitories and communities can implement the idea without paying much costs.

2. Problems being Investigated

(1) Washing Machines

Problems.

Dormitory students need to go to the site to check status of the washing machine in person, so they can know whether there is available machine. Furthermore, most of the students often do laundry in peak hour. Therefore, lining up on the site becomes a critical problem for students. Students also confronted with using machines on the same floor, whereas there are idle machines on other floors. Besides, the previous malfunction machines reporting system is time-consuming and lengthy. It took at least five days for the faulty machine to be repaired. Still, paper-delivered reporting applications are easily lost leading to cope with the repair event out of time. In this case, students suffer from having a few available machines and repeated use broken machines. Additionally, administrators are unable to analyze machine's related data to maximize the usage of the machines. For instance, increasing machines in high-usage areas, arranging regular maintenance and fix up reported machines efficiently.

Solutions.

E-Dorm sorts out the problems through detecting the number of available machines on each floor, displaying current status (In use, Idle, Broken) of machines and predicting the remaining time of operating machines. The improvements are that students can view machine status remotely and enhance the availability of every machines. Further, student can set notification on the machine. When the washing process is finished (condition switch from In use to Idle), APP will notify the student. Meanwhile, E-Dorm make statistics on the number of students using washing machines in each period from Monday to Sunday; then visualize usage rate in bar chart helping students avoid peak hour. Administrators can employ long-term statistical data to make properly plans. As for new malfunction reporting system, students can input the number and description of broken machine on E-Dorm and the reports will be sent to administrators immediately. After the machine has been repaired, administrators can update the machine status to "idle" (usable) on the APP. The electronic reporting system solves complicated procedure and promotes paperless concepts.

(2) Package Signing Process

Problems.

Previously, administrators must write paper of package notifications one by one bringing out problems of time-consuming and human errors. To be more specific, administrators may fill in incorrect student number, repeated records, inconsistent information provided on the packages, etc. Moreover, paper-delivered notifications can be ignored or lost, and students did not actually receive the packages. What's more, the package information was not recorded after signing up. If the students had relevant questions, they could not reach clues from the past package records.

Solutions.

E-Dorm implements electronic package signing process to solve the above problems. When a student has package, administrator will send personal notification to the student, and the student's mobile phone will receive a message reminder. Beside, student can also look up for self-package records. Apart from electronic notifications, when administrator enters student's information, the APP will query the most relevant information found in the database. Thus, E-Dorm not only prevents human typing errors but also ensures information accuracy. After the administrator confirms that the student has signed for the package, administrator can click on the "sign up" button to accomplish the process. Importantly, administrator can search all the package records. Hence, if students lost their package, they could ask administrator to inspect package records.

(3) Garbage Truck Reminders

Problems.

If there is no public garbage disposal area, students need to wait for the garbage truck within the specified time. However, due to the busy schedule and after-school clubs, students tend to forget or miss the garbage truck.

Solutions.

Students can set personal garbage reminders on E-Dorm and determine what time before the garbage truck arrive they will receive the reminder.

3. Background of the Problems

After a campus questionnaire survey of 121 people, it shows that more than 90% of the dormitory students often faced the dilemma of not having washing machine available and having to line up for long while. This survey helps us to confirm that detecting the status of washing machine is a necessary requirement for the users. There was no prior development on this issue because no one has figured out the solution. Although existing community management APPs in Taiwan have function of electronic package signing up process, the APPs have not yet combined the application of machine detection. Accordingly, while improving human life, this development has also made breakthroughs in technology field. To conclude, we anticipate this application will helps future developers stand on a higher starting point than predecessors, and then evolved a more progressive dormitory management APP.

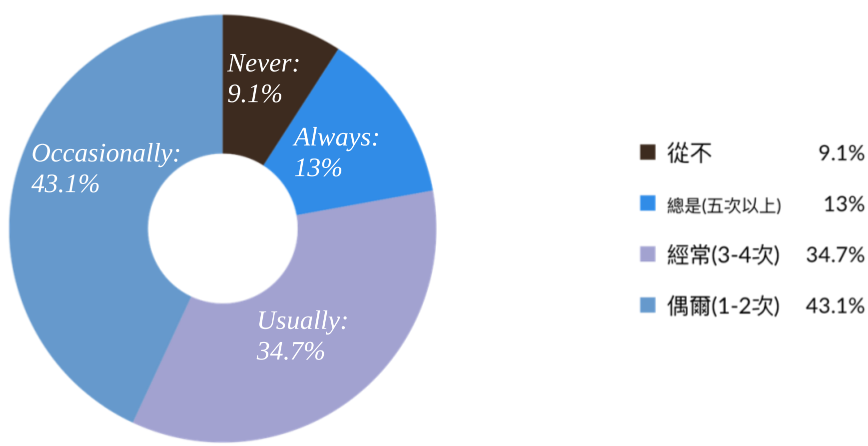

Figure 1.

4. Thesis and General Approaches

(1) Diagram of Back-end Management Architecture

Server includes the following three routing:

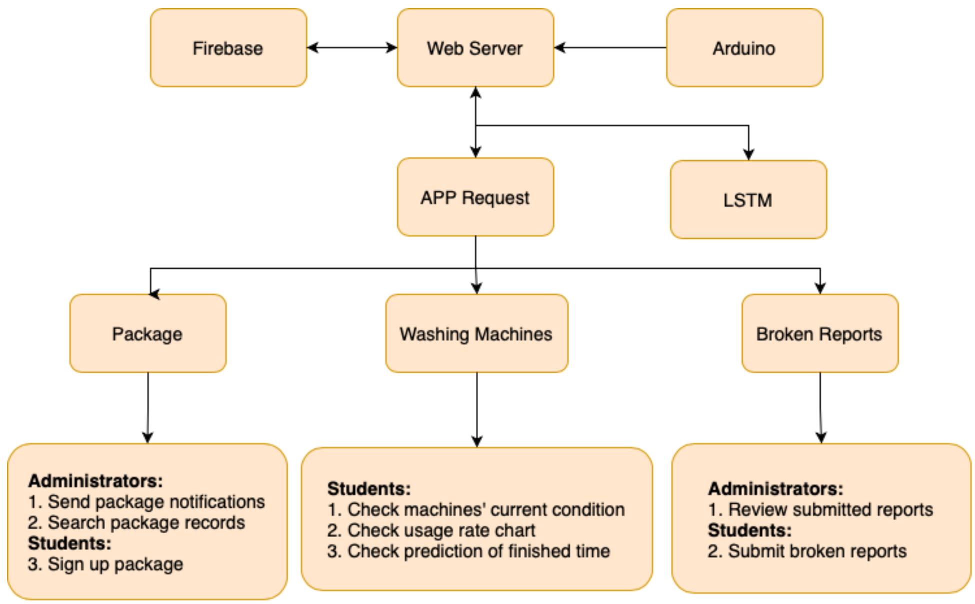

Figure 2.

Package routing: Offer GET and POST methods to get and revise data
Power routing: Offer GET methods for Arduino to update new data of power value.
Repair routing: Offer POST and PUT methods to update data of the reports.

APP Request

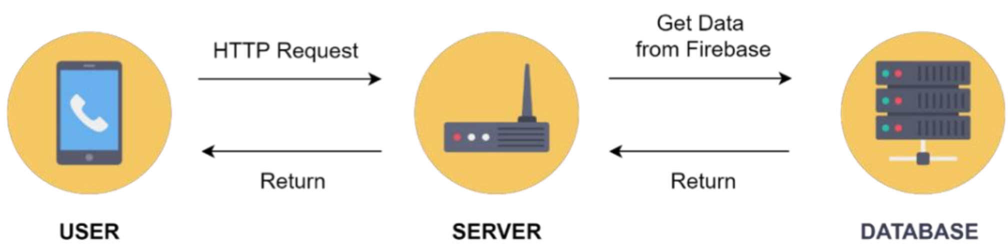

Figure 3.

(2) Diagram of IoT (Internet of Things) Device Structure

Arduino device for detecting the electric current of washing machines.

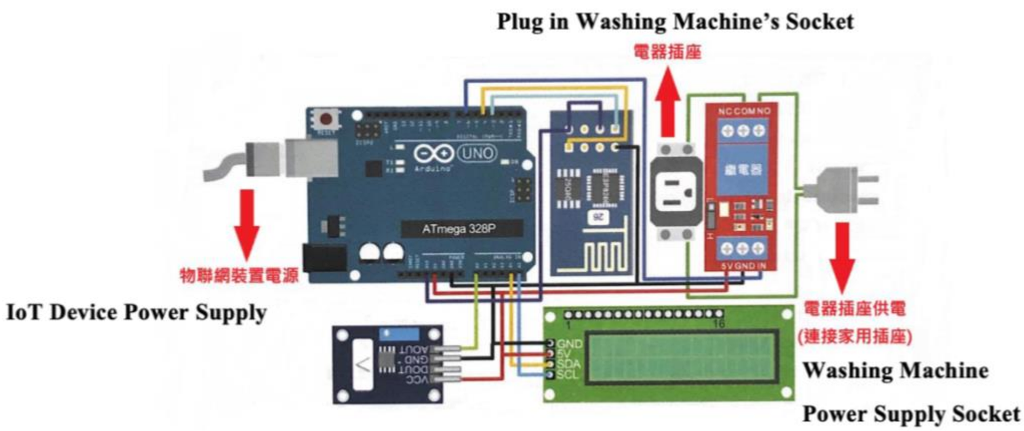

Figure 4.

(3) LSTM

We found that while selecting each washing mode, the required time is a fixed value, and the power values have a correlation with time. Therefore, the LSTM model is used to analyze the power (covert detected electric current to power value) data to estimate the mode currently selected by the user. This method can effectively reduce the time for users waiting for available machine and maximize the usage rate of the washing machine.

The three modes and the corresponding washing time are respectively A (washing times: 2 times) = 26 minutes, B (washing times: 3 times) = 48 minutes, C (washing times: 4 times) = 68 minutes.

The first 30 items of power data will be estimated to a mode. In addition, Android Studio will monitor firebase database in real time. If the estimated time is up, the electric current uploaded by Arduino device is still updating since the machine is still operating. Android Studio further add 5 additional minutes to the estimated time. The status of the machine should change to idle when there is no more electric current detected by Arduino device.

(4) Database Design

Admin Dataset.

As shown in the figure below, the administrator data set stores the list of administrators, and the administrator number is used as the primary key. Admin Dataset used while administrators login their accounts.

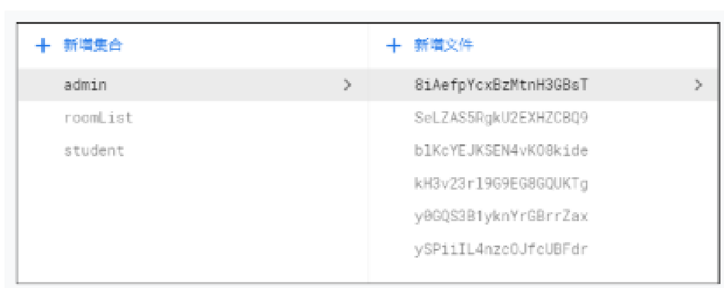

Figure 5.

Bed Number Dataset.

The room number and bed number of the dormitory student are used as the primary key, and a "Reference" column is set to correspond to the Student Dataset, so that the student's package information can be added to their student number, and administrator can search package records through student's bed number since bed number is unique identity for each dormitory student.

Figure 6.

Student Dataset.

Store the dormitory student information, with the student number as the primary key. The internal storage format of the file is:

Student name: STRING

Package array: ARRAY

Login Token: STRING

Package contents: MAP {package number, package status, creation time}

Figure 7.

Machine Dataset.

The Realtime database is in JSON format, and it is a directory of washing machines. The Key is designated as the name of dormitories and the floor of the machine. The value corresponding to the Key is a map array, which stores all the machine's data in certain area. The "con" and "ele" columns should change synchronously as the current status Arduino upload to Server. If the status of current is in use, update "con" to using and "ele" to true. If the power is off, "ele" will be updated to false, "con" is broken or usable. Whether "con" column set to broken is determined by if Server has received repair report. The "id" column is the serial number of each machine, and "rep" is the description attached when the student reported for repairs. While the machine is used, the server sends an update to add the "mode" and "finish" columns on the machine data, "mode" stores the estimated modes predicted by the LSTM model, and "finish" stores the estimated finished time of the machine operation (unit: millisecond).

Machine information: MAP {status, power consumption status, machine number, repair description}

```
"Washer" : {
  "1F" : [ {
    "con" : "usable",
    "ele" : false,
    "id" : 1,
    "rep" : ""
  }, {
    "con" : "using",
    "ele" : true,
    "id" : 2,
    "rep" : ""
  }, {
    "con" : "broken",
    "ele" : false,
    "id" : 3,
    "rep" : "YYDS😎😎"
  }, {
    "con" : "using",
    "ele" : true,
    "id" : 4,
    "rep" : ""
  }, {
    "con" : "usable",
    "ele" : false,
    "id" : 5,
    "rep" : ""
  }, {
    "con" : "usable",
    "ele" : true,
    "id" : 6,
    "rep" : ""
  }, {
```

Figure 8.

(5) Diagram of System Operating Architecture

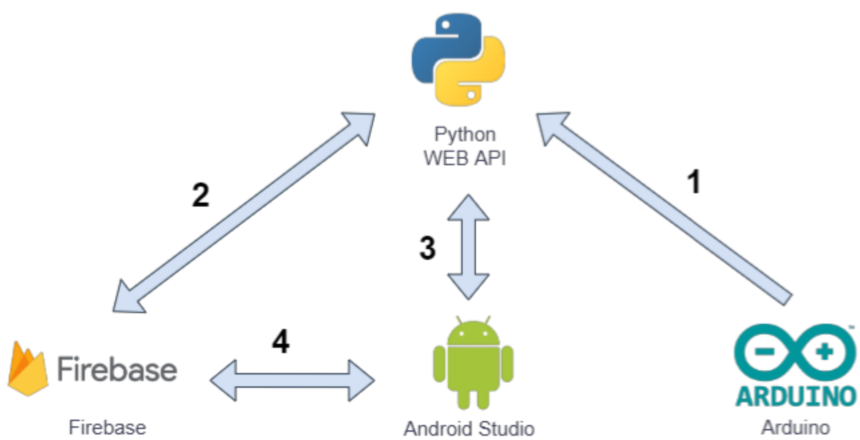

Figure 9.

1 → Arduino uses a specific port to connect to the Server. After Arduino detecting the data of electric current, it will send the machine status to Server. Then, Server will take on to update the machine's current status (In use, Idle Broken). Additionally, the LSTM module in Server will read and analyze the first 60 items of power data when initiating washing machine. Then, LSTM divides the 60 items of data into two pieces of data with a length of 30 items; then throw them into the prediction model respectively. Finally, the outcome is two prediction modes, and the estimated finished time will be average time of the two modes. Therefore, if LSTM module can measure which mode is running through the top 60 items of power value, we can calculate the roughly finished time of washing machine.

2 → Use Python development kit, Flask, to write WEB API methods and connect firebase database to Server. API can use dbkey to update the machine status in the "machine data table" in the database.

3 → After the user selects laundry mode, LSTM model in Server will predict the estimated finished time the process has to take, and display the time on the APP.

4 → In Android Studio, we use HTTP Request to acquire the contents of database, and display contents on the APP. The Firebase Real Time database allows special APPs to be read-only, allowing Android Studio to skip HTTP Request stage and monitor the latest database updates. For example, when Android Studio finds that the machine status in the database variates, it will immediately make corresponding changes. Firebase FCM can also broadcast and send notifications to individual device or entire group of people.

5. Criteria for Study's Success

(1) AI Predicting Precision Up to 80%
Our final predicting precision is up to 97% meaning most of the prediction is consistent with the facts.

(2) Related Enterprise Collaboration
After hearing the development of E-Dorm, dormitory of Tamkang University indicates that they are willing to cooperate with our team and practices the inspiration in large-scale gradually. The collaboration manifests that E-Dorm invention can be commercialized while benefits both school students and dormitory organizations.

(3) Rank in Top Three of Project Competition
Within the campus project practice competition, our team won the second place representing this is a topic worth further spreading and discussing.

Theory

1. Introduction

(1) System Environment
Mobile Operating System (Android 8.1).
Server Operating System (Windows Server 2012).

(2) System Development Tools
Mobile Device: Android Studio.
IoT: Arduino (C++).
Back-end Programming Language: Python, Tensorflow, Keras.
UI Design: Adobe Illustrator 2020Arduino.

2. Arduino Device

(1) Introduction of Using Modules and Electronic Components

Arduino Uno.

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller. It is developed by Arduino.cc. The board is equipped with a set of digital and analog input and output pins that can interface with various expansion boards and other circuits; often utilized on IoT.

ESP8266 WIFI Module.

ESP8266 is a low power consumption UART-WiFi transmission module, which has considerable competitive size and ultra-low power consumption. These advantages make it competitive in the technology industry. It is specially designed for mobile devices and IoT applications. Making user devices connect to the Wi-Fi wireless network and communicate with the Internet or LAN to realize the networking function.

Relay.

Relay is an electronic control device, which has a control system (also known as input circuit) and a controlled system (also known as output circuit). It is usually used in automatic control circuits. In fact, it is an "automatic switch" that uses smaller current to control the larger current. Therefore, it plays the role of automatic adjustment, safety protection, conversion circuit, etc.

WCS1800 Hall Sensor Module.

WCS1800 contains a precision, low temperature drift, linear Hall IC with temperature compensation design and a high temperature fired C-ring current converter. Its internal 9.0mm diameter current channel allows us to monitor the current without destroying or changing the original system mechanism. When the current pass through the internal channel, the current converter of the C-ring will convert this current into a magnetic field proportionally, and the linear Hall IC will convert this magnetic field into an output voltage as the proportion.

(2) Function Description

Using the ESP8266 module to connect to Wi-Fi, the LCD will display the network connection and upload status during the connection process. At the same time, LCD will also display the present power consumption of the socket. After the connection is successful, the Arduino Uno will upload the number of power value to the Server through ESP8266 module, and the local-end can use Excel to record the power value per second.

(3) IoT Device Demonstration

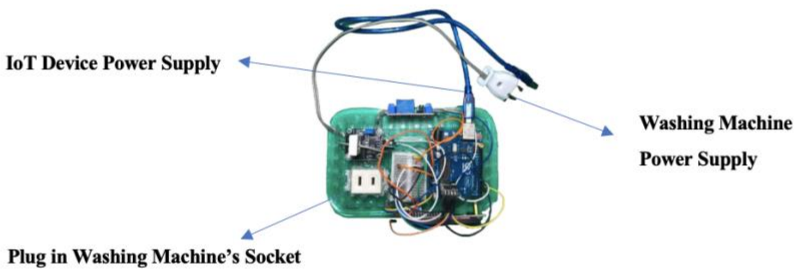

Figure 10.

3. LSTM

(1) Data Collection

Combing Arduino device to the socket of the washing machine; then collect the current used by the washing machine, and upload the power data in real time through WiFi to know whether the washing machine is currently in use. If the detection finds out that the machine is in use, Server employ the collector to get the first 60 items of power value after the machine started to operate. After inputting values into LSTM model in Server, the washing mode and the corresponding finished time is obtained. The reason to make use of the first 60 items of data is because the time for prediction is limited since it is not possible to fulfill prediction exceeding 5 minutes. Accordingly, prediction must start while the machine just operating for shortly period.

(2) Throw in Power Signals Example

Mode A

Index	Power	Status
1.	119	Busy
2.	68	Busy
3.	93	Busy
4.	59	Busy
5.	119	Busy
6.	51	Busy
7.	93	Busy
8.	85	Busy
9.	85	Busy
10.	85	Busy
11.	85	Busy
12.	0	Busy
13.	76	Busy
14.	102	Busy
15.	76	Busy
16.	102	Busy
17.	51	Busy
18.	93	Busy
19.	85	Busy
20.	85	Busy
21.	102	Busy
22.	76	Busy
23.	8	Busy
24.	76	Busy
25.	102	Busy
26.	51	Busy
27.	85	Busy
28.	76	Busy
29.	85	Busy
30.	93	Busy

Mode B

Index	Power	Status
1.	25	Busy
2.	8	Busy
3.	25	Busy
4.	59	Busy
5.	17	Busy
6.	16	Busy
7.	8	Busy
8.	25	Busy
9.	16	Busy
10.	17	Busy
11.	33	Busy
12.	8	Busy
13.	33	Busy
14.	8	Busy
15.	16	Busy
16.	16	Busy
17.	51	Busy
18.	33	Busy
19.	0	Busy
20.	33	Busy
21.	0	Busy
22.	25	Busy
23.	0	Busy
24.	25	Busy
25.	16	Busy
26.	42	Busy
27.	33	Busy
28.	8	Busy
29.	33	Busy
30.	8	Busy

Mode C

Index	Power	Status
1.	68	Busy
2.	16	Busy
3.	17	Busy
4.	34	Busy
5.	16	Busy
6.	67	Busy
7.	34	Busy
8.	8	Busy
9.	25	Busy
10.	25	Busy
11.	8	Busy
12.	16	Busy
13.	25	Busy
14.	34	Busy
15.	8	Busy
16.	50	Busy
17.	25	Busy
18.	8	Busy
19.	16	Busy
20.	25	Busy
21.	68	Busy
22.	16	Busy
23.	17	Busy
24.	34	Busy
25.	0	Busy
26.	229	Busy
27.	25	Busy
28.	0	Busy
29.	8	Busy
30.	25	Busy

Figure 11.

(3) Data Processing

After all the data is labelled by their modes A, B, and C. If the "Status" column shows continuous "Busy", it is determined that the machine is now in use, and the power value of the next 30 items of data will be captured (Figure 12) to fulfill prediction. "Type" column is added to help discern which mode the data belonging to. To increase the amount of training data, 300 more pieces of data are simulated according to the three modes' original feature, and thrown into the training process. Because the first 30 pieces of data are analyzed (n>=30), most statistical examples conform to the normal distribution. To be closer to the actual situations, 95% and 5% of the simulation data will be generated separately. The 95% of the 300-simulation data will produce variables that conform to the normal distribution characteristics of the original data (Figure 13). The remaining 5% of the data is dividing detected power value into segments of 0~5, 5~10, 10~40, and above 40 value. The random variables generated by each segment will be added or subtracted to the original date of power value. Therefore, random part will generate noise in the training for the noise will make training more accord with the reality, and the prediction precision further promoted. Lastly, dividing 80% of all 900 items of data as training data (Figure 14) and throws it into training model after one-hot encoding.

day	time	power	bound	status
2021/8/14	下午 02:32:32	119	Over	Busy
2021/8/14	下午 02:32:33	68	Over	Busy
2021/8/14	下午 02:32:34	93	Over	Busy
2021/8/14	下午 02:32:35	59	Under	Busy
2021/8/14	下午 02:32:36	119	Over	Busy
2021/8/14	下午 02:32:37	51	Under	Busy
2021/8/14	下午 02:32:38	93	Over	Busy
2021/8/14	下午 02:32:39	85	Over	Busy
2021/8/14	下午 02:32:40	85	Over	Busy
2021/8/14	下午 02:32:41	85	Over	Busy
2021/8/14	下午 02:32:42	85	Over	Busy
2021/8/14	下午 02:32:43	0	Under	Busy
2021/8/14	下午 02:32:44	76	Over	Busy
2021/8/14	下午 02:32:45	102	Over	Busy
2021/8/14	下午 02:32:46	76	Over	Busy
2021/8/14	下午 02:32:47	102	Over	Busy
2021/8/14	下午 02:32:48	51	Under	Busy
2021/8/14	下午 02:32:49	93	Over	Busy
2021/8/14	下午 02:32:50	85	Over	Busy
2021/8/14	下午 02:32:51	85	Over	Busy
2021/8/14	下午 02:32:52	102	Over	Busy
2021/8/14	下午 02:32:54	76	Over	Busy
2021/8/14	下午 02:32:56	8	Under	Busy
2021/8/14	下午 02:32:57	76	Over	Busy
2021/8/14	下午 02:32:57	102	Over	Busy
2021/8/14	下午 02:32:58	51	Under	Busy
2021/8/14	下午 02:32:59	85	Over	Busy
2021/8/14	下午 02:33:00	76	Over	Busy
2021/8/14	下午 02:33:01	85	Over	Busy
2021/8/14	下午 02:33:02	93	Over	Busy
2021/8/14	下午 02:33:03	85	Over	Busy
2021/8/14	下午 02:33:04	8	Under	Busy
2021/8/14	下午 02:33:05	76	Over	Busy
2021/8/14	下午 02:33:06	119	Over	Busy
2021/8/14	下午 02:33:07	59	Under	Busy
2021/8/14	下午 02:33:08	85	Over	Busy
2021/8/14	下午 02:33:09	76	Over	Busy
2021/8/14	下午 02:33:10	85	Over	Busy
2021/8/14	下午 02:33:11	85	Over	Busy
2021/8/14	下午 02:33:12	76	Over	Busy
2021/8/14	下午 02:33:13	93	Over	Busy
2021/8/14	下午 02:33:14	76	Over	Busy
2021/8/14	下午 02:33:15	8	Under	Busy
2021/8/14	下午 02:33:16	161	Over	Busy
2021/8/14	下午 02:33:16	161	Over	Busy
2021/8/14	下午 02:33:17	85	Over	Busy
2021/8/14	下午 02:33:18	76	Over	Busy
2021/8/14	下午 02:33:19	85	Over	Busy
2021/8/14	下午 02:33:20	85	Over	Busy
2021/8/14	下午 02:33:21	85	Over	Busy
2021/8/14	下午 02:33:22	85	Over	Busy
2021/8/14	下午 02:33:23	76	Over	Busy
2021/8/14	下午 02:33:24	85	Over	Busy
2021/8/14	下午 02:33:25	34	Under	Busy
2021/8/14	下午 02:33:26	110	Over	Busy
2021/8/14	下午 02:33:27	42	Under	Busy
2021/8/14	下午 02:33:28	85	Over	Busy
2021/8/14	下午 02:33:29	93	Over	Busy
2021/8/14	下午 02:33:30	76	Over	Busy
2021/8/14	下午 02:33:31	85	Over	Busy
2021/8/14	下午 02:33:32	85	Over	Busy
2021/8/14	下午 02:33:34	110	Over	Busy
2021/8/14	下午 02:33:35	76	Over	Busy
2021/8/14	下午 02:33:36	85	Over	Busy
2021/8/14	下午 02:33:37	16	Under	Busy
2021/8/14	下午 02:33:38	76	Over	Busy
2021/8/14	下午 02:33:39	93	Over	Busy
2021/8/14	下午 02:33:40	68	Over	Busy
2021/8/14	下午 02:33:41	102	Over	Busy
2021/8/14	下午 02:33:42	85	Over	Busy
2021/8/14	下午 02:33:43	76	Over	Busy
2021/8/14	下午 02:33:44	85	Over	Busy
2021/8/14	下午 02:33:45	68	Over	Busy
2021/8/14	下午 02:33:46	85	Over	Busy
2021/8/14	下午 02:33:47	76	Over	Busy
2021/8/14	下午 02:33:48	8	Under	Busy
2021/8/14	下午 02:33:49	76	Over	Busy
2021/8/14	下午 02:33:50	0	Under	Busy
2021/8/14	下午 02:33:51	93	Over	Busy
2021/8/14	下午 02:33:52	281	Over	Busy
2021/8/14	下午 02:33:53	323	Over	Busy
2021/8/14	下午 02:33:54	153	Over	Busy
2021/8/14	下午 02:33:55	25	Under	Busy
2021/8/14	下午 02:33:56	76	Over	Busy
2021/8/14	下午 02:33:57	8	Under	Busy
2021/8/14	下午 02:33:58	85	Over	Busy
2021/8/14	下午 02:33:59	8	Under	Busy
2021/8/14	下午 02:34:00	76	Over	Busy
2021/8/14	下午 02:34:01	8	Under	Busy
2021/8/14	下午 02:34:02	76	Over	Busy

Figure 12.

```python
# 原資料當作母體填入平均&標準差產生模擬資料
def generate_data(mean,std,n):
    g=np.round(np.random.normal(mean,std,n))
    return np.abs(g)

# 噪音資料產生
def noise(array):
    res = []
    for i in array:
        if i<10 and i>=5:
            res.append(i+np.random.randint(-3,55))
        elif(i>10 and i<40):
            res.append(i+np.random.randint(-5,60))
        elif(i>=0 and i<5):
            res.append(i+np.random.randint(0,55))
        elif(i>40):
            res.append(i+np.random.randint(-5,60))
    return res
# 產生資料量
total_size=300  #每part總資料量
noise_rate=0.05 #雜訊比率
dataSize = round(total_size*(1-noise_rate))
noise_size=round(total_size*0.05)
```

Calculate original detected data's average value and standard deviation.

Generate noise data.

Figure 13.

```python
# MixData get Simple for x_train
mixData=pd.concat([df3,df2,df1],axis=0)
x_mixData=np.array(mixData.power.values)
y_mixDate=np.array(mixData.type.values)
# 切割資料集
x_train , x_test , y__train , y_test =train_test_split(x_mixData,y_mixDate,test_size=0.2,random_state=10)
x_train=np.array(list(x_train))
x_train.shape=(-1,30,1)
```

Figure 14.

(4) Build Model

The tools used for machine learning is Keras and Pandas, and the programming language is Python. The input layer of LSTM is related to time series. To elaborate, it must be three-dimensional data (batch_size, time series, power value) and connected to 3 output neurons. The number of epochs iterations is 50, the fit parameter batch is set to 5, and 25% of the total 720 training data are used as the verification set. The Activation Function used is "Softmax", which can convert the Y value into a probability. The loss function is set to "Logarithmic loss", which is suitable for multi-classification problems. After countless experiments on parameter setting, we finally obtain the optimal model.

```
model = Sequential()
model.add(LSTM(10,input_length=30,input_dim=1))
model.add(Dense(units=3,activation='softmax'))
model.compile(loss=losses.BinaryCrossentropy(),optimizer=optimizers.Adam(), metrics=[ 'accuracy' ])
model.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_4 (LSTM)                (None, 10)                480
_________________________________________________________________
dense_4 (Dense)              (None, 3)                 33
=================================================================
Total params: 513
Trainable params: 513
Non-trainable params: 0
_________________________________________________________________
```

Figure 15.

Results

1. AI Model Evaluation

Separate the test set with Test = 0.2, the accuracy of the test set is approximately 95% ~ 97%. When machine models A, B, C are put in the model (excluding those data that have been trained beforehand), the model can correctly measure in most of the time.

acc: 0.9666666666666667

	predict_result	label_result
A	56	56
B	56	59
C	68	65

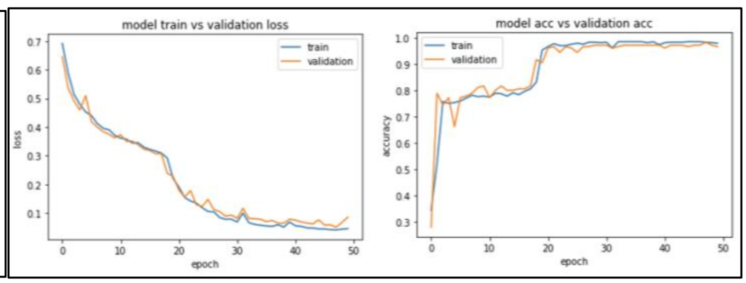

Figure 16.

Discussion

1. The Meaning and Significance of The Results

Finding that the power of the washing machine has a linear relationship with the washing mode, in order not to violate privacy of users and disassemble the washing machines, we decided to use LSTM for AI prediction. Finally, our team successfully detect washing machine's electric current, convert the current into power, upload power value to Server, use the LSTM model to analyze the laundry mode and predict the finished time of the laundry. The prediction accuracy rate is as high as 97%. We eventually improve difficulties of dormitory lives without insulting user privacy and make it easier for schools and communities to imitate the implement.

2. Faced Difficulties and Final Solutions

(1) Methods to Estimate The Finished Time of Washing Machines

Difficulties.

At the beginning, we first thought of analyzing the remaining time showed on the screen but found that this action required dismantling the machine to receive the information on the screen. In this way, it will be inconvenient for other organizations to imitate the implement. Later, we thought of using monitor to record the image in the video and estimate the remaining time showed in graphics. Still, this method involves the privacy right of the users. For instance, the monitor will record the user's face and the details of users' clothes. Moreover, the remaining time on the screen may be blocked by obstacles or students. Therefore, the method will require multiple cameras to solve the problem of blind spots, and multiple high resolution cameras cost large amount of expense.

Solution.

After further research, we found that there is a corresponding time for each mode, and the power used in each mode is also unique. Then, we decide to use the LSTM model to analyze the power, acquire the mode selected by the user, and display when the washing machine will finish the washing process on the APP. The most important thing is that LSTM estimation solves the difficulties of dismantling washing machines, privacy issues, and purchasing multiple cameras.

(2) Low Prediction Accuracy

Difficulties.

When the LSTM model is initially used to predict, the results are always different from the actual facts.

Solution.

To increase the accuracy rate, we further study the power data of each mode, obtain the average value and standard deviation. Next, we use the normal distribution to generate more samples for training. After the number of samples increases significantly, the accuracy rate also increases. However, the prediction results begin to appear overfitting causing the result to be wrong. Therefore, we create some noise information to fit in the actual situation by putting random variables into the detected power data in intervals. After adjusting the parameter values, the highest accuracy rate is up to 97%.

(3) Mechanism to Prevent Prediction Errors

Difficulties.

To make up for the inaccurate predictions in a few cases, which lead to the predicted end time is far from the actual finished time.

Solution.

While solving this problem, we originally used the first 30 items of detected currents when the machine was started, and then converted the current into a power value. Instead, we used the 60 items data after the machine was started, so two modes can be obtained. If the two predicted modes are different, the average time of the two modes will be displayed on the APP. As a result, the difference between error and actual result can be reduced.

3. Unexpected Results with Theoretical Expectations

(1) Association with Washing Machine Power and Mode

At the first, we didn't think that the power of the washing machine was related to each mode. We originally wanted to use Fourier Transformation to convert the time signals into frequency. Later, after several statistics, we found the data regularity of each mode. Finally, we use method as the above mentioned to adjust the accuracy rate of training process. To sum up, I learned the concept and application of Fourier's which was a unexpected gain, and the importance of analyze statistics of data.

(2) Arduino Sensor May Burn Out

In the first phase, while detecting the current of the washing machine, we did not consider that the Arduino sensor might burn out due to a short circuit or overload current. Therefore, the current values measured in previous tests were not consist with common circumstance. After updating the sensor element, the detected current value became correctly range in reasonable value. As detecting other machines in the future, we need to pay attention to the power consumption of the machine avoiding the sensor from burning out so the end time cannot be predicted. Furthermore, we need to replace the sensor regularly to reduce the possibility of sensing errors.

Conclusion

After developing the E-Dorm APP, we hope to be continuing improve the project in the future to make more people familiar with and interested in this application. At present, we are still working hard to expand involved aspects, such as the following prospects.

1. Detecting Other Dormitory Equipment

   For now, the hardware equipment we use can correctly detect the current of the washing machine. As for other machines in the dormitory, such as dryers and hydro extractor have not been examined. Therefore, there is still room for machines type expansion. If the hardware can detect more diverse devices, the APP can detect regularity of different machines by adjusting the program and adding LSTM training models. Making all the machines in the dormitory can be planned more comprehensively.

2. Adjunction of Appointment System

   If an appointment system can be introduced in the future, the overall operation of E-Dorm will be more efficient. Although it appears that there is no clue whether the former user has taken out his or her clothes, and there are various potential factors such as someone jumping in the queue. Nevertheless, if a reservation system can be established integrating online payment system, the dormitory student can make an appointment for the washing machine in advance and directly determine when he or she can use it. It will undoubtedly enhance convenience to another supreme level.

3. Partnership with Self-Service Laundry Stores

   In addition to collaborate to dormitories, we also reckon that the hardware of this system can be applied to the management of self-service laundry stores. In this way, customers can know whether there are machines available via the APP. Besides, cooperating with self-service laundry stores, we can generate additional income to develop more upgrade functions on the APP.

References

[1] Yang, M. (2018). Best Arduino IoT Introduction and Applications: Build a smart house with easy ways, Chi Feng Publisher, Taipei City.

[2] Brandon Rohrer 2017, How Recurrent Neural Networks and Long Short-Term Memory Work, Brandon Rohrer published, viewed 1 August 2021, <https://e2eml.school/how_rnns_lstm_work.html>.

[3] Teng-Yuan Chang 2019, Brief Talk about Recurrent Neural Network (RNN) and Long Short-term Memory Model (LSTM), Teng-Yuan Chang published, viewed 1 August 2021, <https://brohrer.mcknote.com/zh-Hant/how_machine_learning_works/how_rnns_lstm_work.html>.

[4] JT 2018, LSTM Deep Learning: Stock Price Prediction Practice, JT published, viewed 1 August 2021, <https://medium.com/data-scientists-playground/lstm-深度學習-股價預測-cd72af64413a>.

[5] DataSCi Ocean 2021, Understanding the input and output of LSTM in Keras, DataSCi Ocean published, viewed 31 August 2021, <https://www.potatomedia.co/post/355196c7-4c37-4ee4-a9cb-8c3ec965313d>.

[6] Kyle Mo 2020, Preliminary understanding of distributed database and NoSQL CAP theory, Kyle Mo published, viewed July 2021, <https://oldmo860617.medium.com/初步認識分散式資料庫與-nosql-cap-理論-a02d377938d1>.

[7] Wei Chieh 2019, Create a cloud Firestore database in ten minutes, W3HexSchool published, viewed July 2021, <https://w3c.hexschool.com/blog/8a8d783c>.

[8] Firebase Document, Firebase Teaching Document, Firebase published, viewed July 2021, <https://firebase.google.com/docs?authuser=0>.