

Design and review on applying the neural network
to the abandoned bike report system

Keyword: Neural network, post review, MLP

One. Introduction

I. Motivation

My friends and I held the charity event “Abandoned Bike Revival” for the past three years to lead students to find abandoned bikes on the street. We found over 1,000 bikes and reported to the cleaning squadron. Over 700 of them have been recycled. We set up a Facebook fanpage, wrote a reporting app and got a great outcome. Our fanpage and app helped the cleaning squadron with bike recycling. However, we found these issues during the event:

1. The government has a great bike recycling service but each county and city have its own online reporting website, which is inconvenient. Most of the people don’t know how to report the bike.
2. We used App Inventor to design this app. It can only be used on Android systems but not IOS systems. Nowadays people don’t want to use unfamiliar apps.
3. The reported cases must be checked manually for qualification before being sent to the cleaning squadron. It takes extra labor. The app is inefficient because the case must be checked manually. The informer cannot receive the feedback immediately. People would be reluctant to report the bike.

To solve these issues, I came up with a solution by replacing the Facebook chatbot with the app. It is easier to use and users of all kinds of mobile systems can report the bike. In school, our instructor of Information Technology Class introduced AI. I was attracted by this technology and wanted to learn more. I want to write an AI chatbot to simplify the process and deliver a better user experience. I hope that people can report abandoned bikes on the street easily and help the cleaning squadron recycle them, to facilitate bike recycling.

There are lots of AI neural network models. I want to find out which model can be used for this study? How to set the parameter? How to use AI for identification of abandoned bikes, house number identification and case rating? This research will explore and explain the process, method, theory and result of the research profoundly.

II. Purpose

When I came up with the idea of using AI to determine whether the reported case can be sent directly, I realized that a reported case is like a post. It is like posting on the online forum, discussion forum, auction platform, social platform and public media. The reported case and post have the following in common:

1. There might be lots of posts. If each post must be checked manually before being posted, it costs a lot and it is inefficient.
2. Picture is an important factor of a post. However, copyright might be violated and the picture might be considered an advertisement.
3. Some posts are the first posts of the new users. There is no previous rating for reference.
4. Users with high ratings are more trustworthy. However, they might write a low-quality post, and vice versa. However, the post cannot only be judged by the previous rating.

In this study, the AI rating system designed for reporting cases is an “automatic post rating system” and can be used for all kinds of forums. It decides whether a post can be posted directly or needs further review or needs to be blocked. The government can check the validity of the reported or claimed case automatically. This system can be used in various fields. Therefore, when studying this AI solution, I need to consider how to use this solution for other types of posting. I hope that our research outcome can be used in more platforms and be more valuable.

Besides, I use AI to check if there is at least an abandoned bike in the uploaded photo. I want to make it as precise as possible. I also want to use AI for house number identification, which has never been achieved before. When the user reports the abandoned bike, he does not need to type any text. He can take a photo of the doorplate, upload and convert it into an address in the cloud, and provide it to the cleaning squadron.

Based on these study objectives, I want to achieve the following in this study:

1. Make a Facebook Messenger chatbot for people to report the bike conveniently anytime.
2. Create an AI system to rate the reported case accurately. The cleaning squadron can recycle the bike based on the rating.
3. Use AI to identify if there is an abandoned bike in the photo.
4. Use AI to analyze and identify an accurate and complete address and house number based on the doorplate in Taiwan.

5. In the study, explore the way to determine the number of neurons in the hidden layer of MLP. Try to find a solution for unsolved issues in AI deep learning.

Two. Research method and process

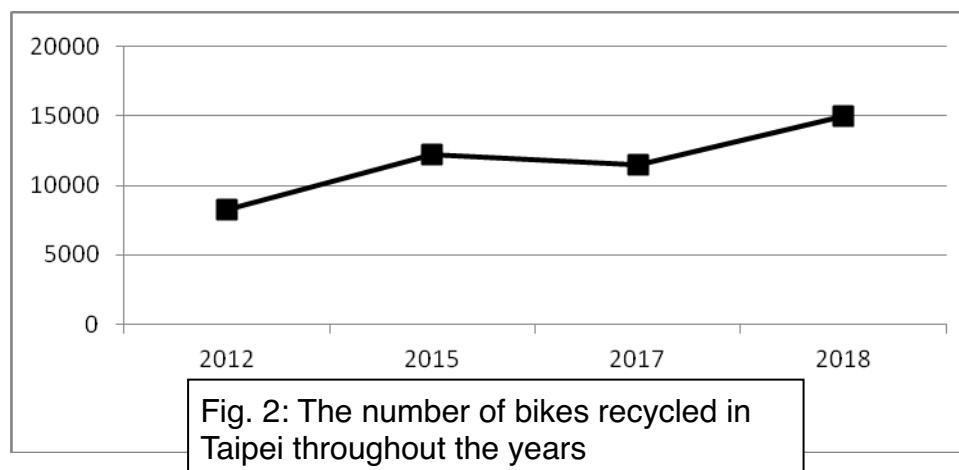
I. Research equipment and device

- Two mobile phones: Test the chatbot.
- Two computers: They are used for programming, writing reports and doing experiments.
- Jupiter Notebook: It is an environment for developing the neural network learning script.
- Keras + Tensorflow: It is a neural network platform. Even beginners can use AI.
- Facebook Developer: Set a fanpage and the chatbot link.
- AWS: It is used to store all the programs and databases created in this study and it provides a 24-hour reporting service.
- Google Cloud Vision API: It is a powerful image identification engine. The user can mark specific objects in the photo. It can identify texts. This research is used to analyze images and acquire its output for AI application.
- Google Maps API: It is a geographical information platform. In this research, it is used to find a complete address based on a partial address.

II. Background and relevant theory

(I) Process and bottleneck for handling the abandoned bike

Biking has been popular in Taiwan for the past two decades. Throughout the time, there have been more abandoned bikes on the street. In Taipei, the cleaning squadron recycled 6,000 bikes in 2003, 8,264 bikes in 2012, 12,173 bikes in 2015, 11,500 bikes in 2017, and over 15,000 bikes in 2018 (Fig. 2) [1][2].



The cleaning squadron works hard, but many abandoned bikes still occupy the rack and block the sidewalk. The city street looks messy. They hinder pedestrian safety because they are not recycled properly (Figs. 3 and 4).



Fig. 3: The abandoned bike hinders the safety of pedestrians



Fig. 4: The abandoned bike occupies the rack

Recycling abandoned bikes mostly relies on patrol by the cleaner and reports by people. However, it's not the main job of the cleaning squadron and there are not enough people for



Fig 5: The recycling standard for abandoned bike

recycling. Cleaners can only check if there's abandoned bike while collecting the trash. Therefore, there are lots of blind spots.

After someone reports the bike, the cleaner will go check the spot in a few days. If the abandoned bike meets the requirement (Fig. 5), the cleaner will attach an announcement for recycling on the seat (Fig. 6). The cleaning squadron will come back seven days later. If the owner did not remove the post, the cleaning squadron will take it to the disposal area and post it on the website for a month. The owner can collect it within a month. If no one collects the bike after a month, the bike will be handled at the disposal area (Fig. 7). If the bike can be repaired, an auction

will be held. If it cannot be repaired, the usable parts will be removed to repair other bikes. The rest of the parts will be recycled by the material.



Fig 6: The cleaning squadron posts an announcement on the bike



Fig 7: Repair or recycle the bike. Give this bike a new life

The problem now is that not many people would report the bike. The main reason is that the reporting procedure is inconvenient. Many people don't know that they can report the bike. That is why we use AI to create a Facebook chatbot as the bike reporting system.

(II) AI theory

Wikipedia states that Artificial Intelligence (AI) is also known as machine intelligence. It means the intelligence of the machine created by humans [3]. The most important part of intelligence is learning. The machine is able to learn with the help of AI. We call it machine learning. To achieve artificial intelligence, the popular method nowadays is neural networks. It is a computing model simulating human's cranial nerve structure. It aims to help the computer simulate how human thinks to solve abstract problems [4][5]. When using the neural network for machine learning, people developed numerous network models. Nowadays, the mainstream consensus is that there should be multiple layers in these networks, which means there should be at least one hidden layer between the input and output layers. It is called deep learning. In 2006, Geoffery Hinton proposed deep learning (multilayer neural network). This idea performed well in 2012 ImageNet [5].

Now there are three basic models for deep learning, Multi-layer Perceptron (MLP), Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) [6] [7]. These are explained briefly below:

(1) Multi-layer Perceptron (MLP)

MLP is a classic neural network. It consists of one or multiple neurons. The data is sent to the input layer and stored in one or multiple hidden layers. A prediction is made on the output layer. MLP is very flexible and can be especially used for classifying and predicting problems [8].

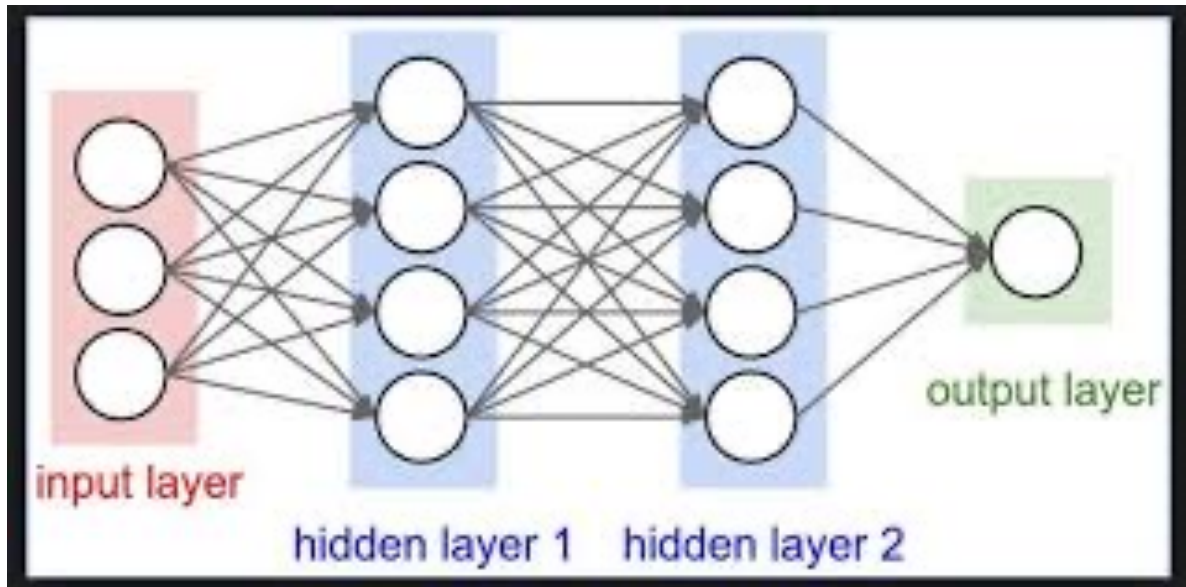


Fig. 8: The diagram for MLP neural network model

(2) Convolutional Neural Network (CNN)

CNN is different from MLP. It divides numerous input neurons into several sections. Each section handles a small part of the information. These sections integrate the learning results and enter the next layer for learning. This effectively reduces complexity and does not omit important information. Many famous AI systems, e.g. Google Cloud Vision, Alpha Go, have made breakthroughs because they are based on the CNN model.

Through partial connection and shared weights and biases, CNN significantly reduces the number of parameters. Now CNN has achieved exciting achievements in numerous computer vision tasks, such as image identification, target detection and facial recognition [8].

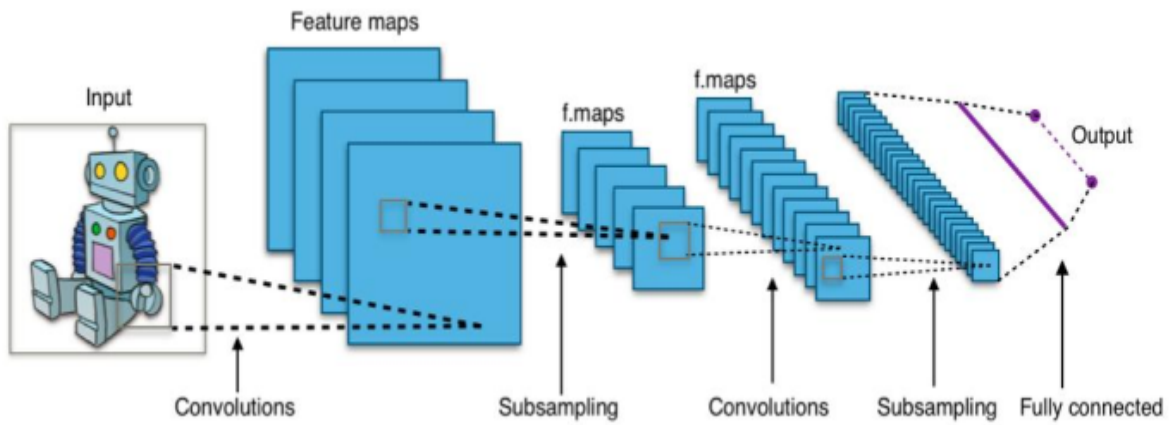


Fig. 9: The diagram for CNN neural network model

(3) Recurrent Neural Network (RNN)

Compared to MLP and CNN, RNN uses shared weights and biases to handle the extended sequence. (The fully connected layer needs the input from the fixed dimension, CNN can only accept input from the fixed dimension.) RNN introduced a ring structure. The input at a certain time is associated with the concurrent input and the status at the previous time. Through shared weights and biases, RNN is especially suitable for learning signals with timelines, such as translation, text mining, chatbot, voice recognition...[5], [8].

The number of hidden nodes is a critical parameter of MLP. The neural network with too many nodes might overfit data. Its generality for the data not used for the training is poor. The number of hidden units configured under the model is low. The neural network is not quite accurate [9], [10]. This issue deeply troubles scientists. Therefore, they brought up aspects from different perspectives [11][12][13]. Jason Brownlee suggested using the following five methods to find the solution, experiment, intuition, prioritized depth, literature review and case search [14]. Foram thought that no specific method could predict the number of layers or neurons required. He organized several common suggestions, such as that the number of neurons on the hidden layer should be on the same quantity level as the output and input layers [15].

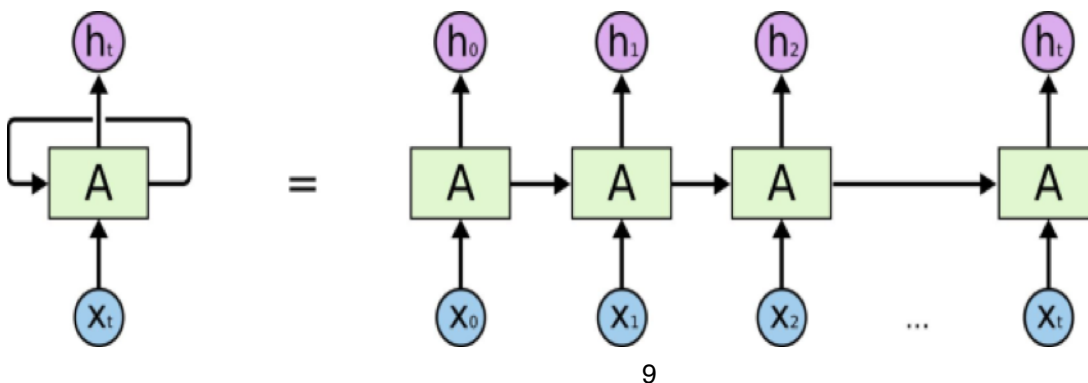
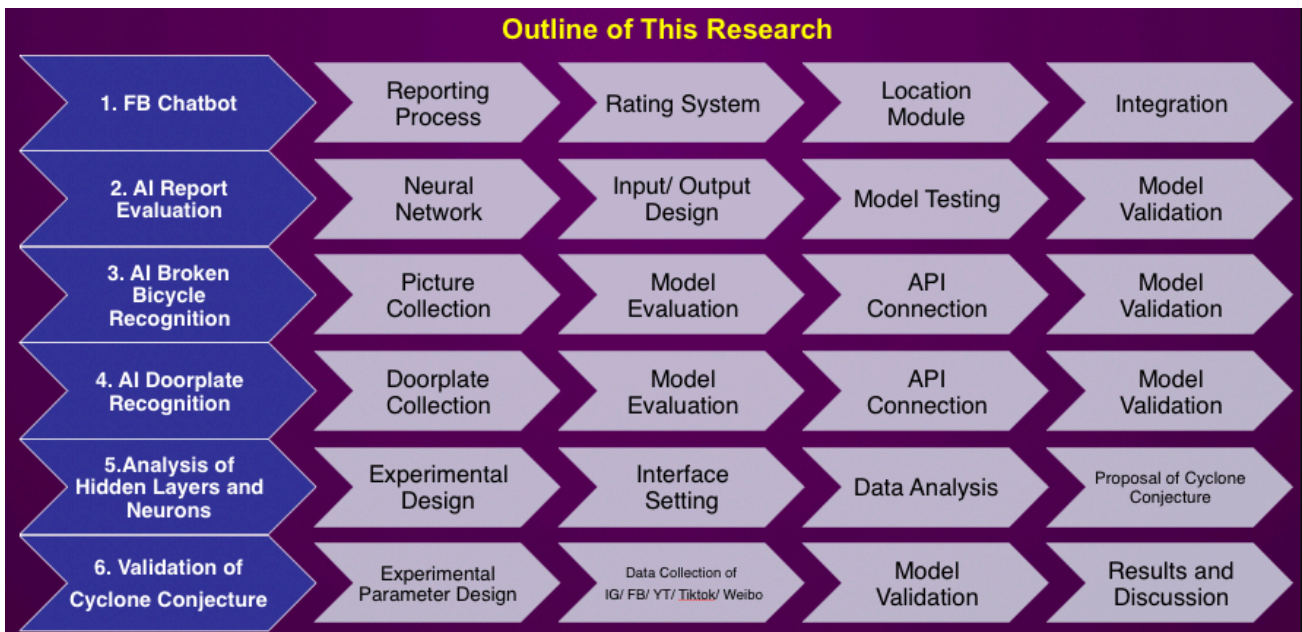


Fig. 10: The diagram for RNN neural network model

Based on other literature reviews ([16], [17], [18]) and some of our small experiments, there is no solution for the number of hidden layers and the number of neurons on each layer in the MLP model. The relationship between the output and input layers is changeable. To be extreme, if the data provided for the input layer and the expected answer expected by the output layer are random and unrelated, learning is impossible regardless of the number of layers or neurons.

(III) Method and process

According to the study purpose, this study can be divided into five parts:



- 1. Facebook chatbot**
- 2. AI post rating**
- 3. Bike image recognition**
- 4. Address analysis based on house number**
- 5. Study on the number of hidden neurons**

Based on the five parts above, the following diagram shows the study method and process:

Fig. 11 Research method and process

Three. Research result and discussion

I. Facebook Messenger chatbot

(I) Reporting process design

This study uses Facebook Messenger as the interface for reporting the abandoned bikes on the street. Its advantage is that it is easy to use, widely acceptable, popular, and stability. The user process designed in this study is as shown below (Fig. 12):

(II) Achievement system design

According to our experience in the past three years, each person reports 5 cases on average. (About 200 people reported 1,000 cases.) However, the top 40 users reported 700 cases. Each of them reported 17.5 cases on average. Inactive users report less than 2 cases on average. The number of cases reported by active users is nine times the inactive users.



Fig. 12: Chatbot script

If every user can become active, this charity event will be more successful. According to our experience with the mobile game, the achievement system is a key to keeping gamers active and keep playing the game. The target group of this event is elementary school and middle school

students. Based on the education system in Taiwan, the achievement level is divided by grade to rate user achievement. There are 21 levels as follows:

"New student", "Junior level in elementary school", "Middle level in elementary school", "Senior level in elementary school", "Seventh grade", "Eighth grade", "Ninth grade", "Tenth grade", "Eleventh grade", "Twelfth grade", "Freshman", "Sophomore", "Junior", "Senior", "Graduate", "PhD", "Post-Doctoral", "Assistant professor", "Professor" and "Principal."

The formula for the level is as follows:

$$lvl = \text{round}(\text{float}(\text{item}['\text{credit}'])) + \text{min}(\text{int}(\text{item}['\text{totalz}']), 50) // 5$$

Credit is the average point a user gets for all the reported cases for now (10 points the highest) total is the total number of reported cases. (If there are over 50 cases, it is considered 50 cases.) The design of the formula above is to give more points and assign a higher level to the user the more cases the user reports.

(III) Locating address module

The location must be provided for reporting the bike. Facebook Messenger allows users to send the location. The diagram is as follows (Fig. 13):

(IV) Integration and making

Based on the designed process, we used Python to write a program and upload it to AWS (Amazon Web Services).

We used the [New life for abandoned bike] fanpage set up two years ago and connected the chatbot to the Messenger step by step based on the Facebook instructions. This is the screenshot of the main screen after the chatbot is connected successfully:

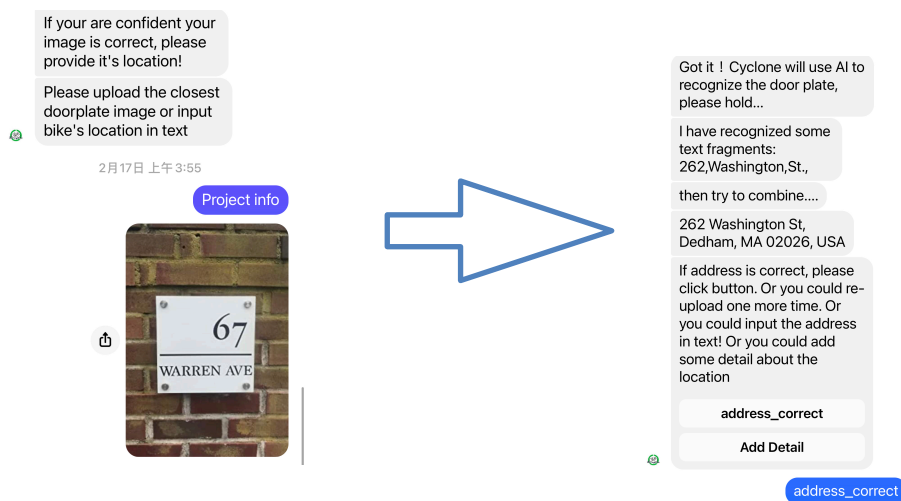


Fig. 13: The screenshot for sending the location via Facebook

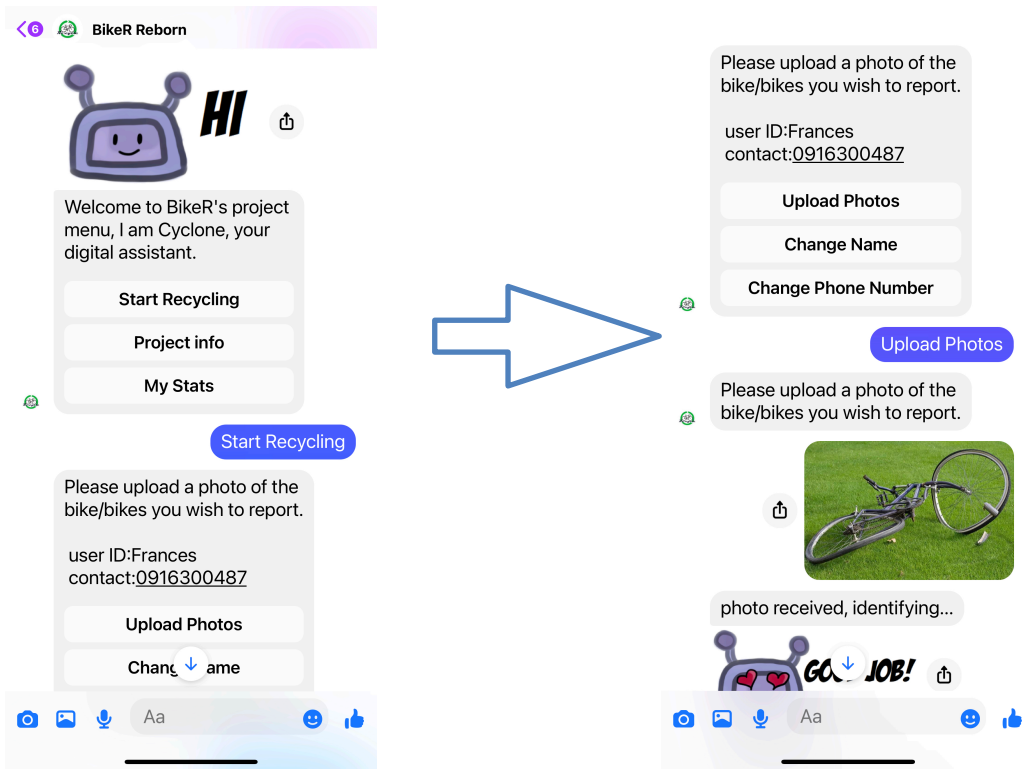


Fig. 14: The screenshot for the dialogue with Chatbot

(II) AI post rating network

Based on the design, the chatbot shall rate the reported cases (0-10). Only the case rated 7 or above (more reliable) can be sent to the cleaning squadron by email. Otherwise, the user has to confirm to report again and email the volunteer to deal with the bike (confirmation over the phone may be required) before sending an email to the cleaning squadron. This part cannot be done by traditional programming because the reported case is varied, just like the posts online. It would be difficult to use this program and it would be hard to maintain the program if using “if... then....else.” Therefore, I tried to use the AI neural network to solve this problem.

The chatbot made in Research 1 collects the reported cases like posts, as shown below:



Fig. 15: The classic surface parameter for reporting the case

Besides, the database can also check some of the cases reported by this user and the number of times he reported the case. The output of the image recognition module and locating address module are also contributed to this reporting process. The following is the result: (The red part is the description provided by the system.)



Fig. 16: The complete parameter for the classic reported case

Research 2 aims to realize an AI neural network. By using the input above, we created a rating (0-10) for the chatbot. The research design is described below.

1. Neural network model design

Based on previous studies, in this study, the MLP neural network model is used. The tools are Google Tensorflow and Kares, which are the most popular. The development environment is Jupiter Notebook, which is also most popular.

2. Input/output layer design

It is defined that there are seven neurons in the input layer. They correspond to the information below:

- Is a photo uploaded? (0/1)
- Is there an abandoned bike in the photo? (0-10)
- Is this a new user today? (0/1)
- The average rating for the reported cases (0-10)
- The total number of reporting times (0-100)
- The number of reporting times in the last ten minutes (0-100)
- Is the address valid? (0/1)

It is defined that there are 11 neurons in the output layer, each representing the rating (0-10). 0 is most disqualified and 10 is most qualified. Whichever the neuron with the highest possibility, the system uses that score as the final rating.

3. Training and test sample database

Use step1-Build_questionary_588_0208.py to generate a list of 588 scenarios that require an interview with the expert (“step1-Build_questionary_588_0208.csv”).

Use “step1-Build_questionary_588_0208.csv” for the interview with the expert. Ask the expert to grade each scenario.

Use step2-nomalize_questionary_5500_0208.py to expand the number of entries for each score to the closest number of entries by using the random number (400-800 entries). The total training samples are 5,500 samples. This ensures that the weight of each score is consistent. The 11 neurons on the output layer have an equal chance to develop. If 357 training samples out of 588 are graded 0, the neurons graded 0 would be strengthened excessively. To provide 550 testing samples and 500 evaluation samples, these two parts are sampled randomly. The file generated finally is Use “step2-Build_questionary_5500_0208.csv”.

4. Training and test verification

Use the script Keras_recycle0209 in Keras to start training the MLP model. Based on the literature review and initial test, we determine the common parameters of these experiments as below:

- Batch_Size is fixed to 40 (about 1% of the total number of training samples).
- 500-2000 epochs are done for each experiment.
- 80% of the samples in the sample database are used as the training samples and 20% of them are used as the verification samples.

After repeated training and verification, the trained neural networks are packaged into a PostRank.h5 file. This .h5 file can be considered as an usable function but the source code cannot be seen. It can receive the input and return the output value after computing. I only need to put this function into the chatbot created in the previous study!

However, during the training, I found out that the biggest problem with the number of neurons in the hidden layer is that, according to other works of literature and suggestions from other research reports, using a hidden layer structure with 10 neurons will get the following training result:

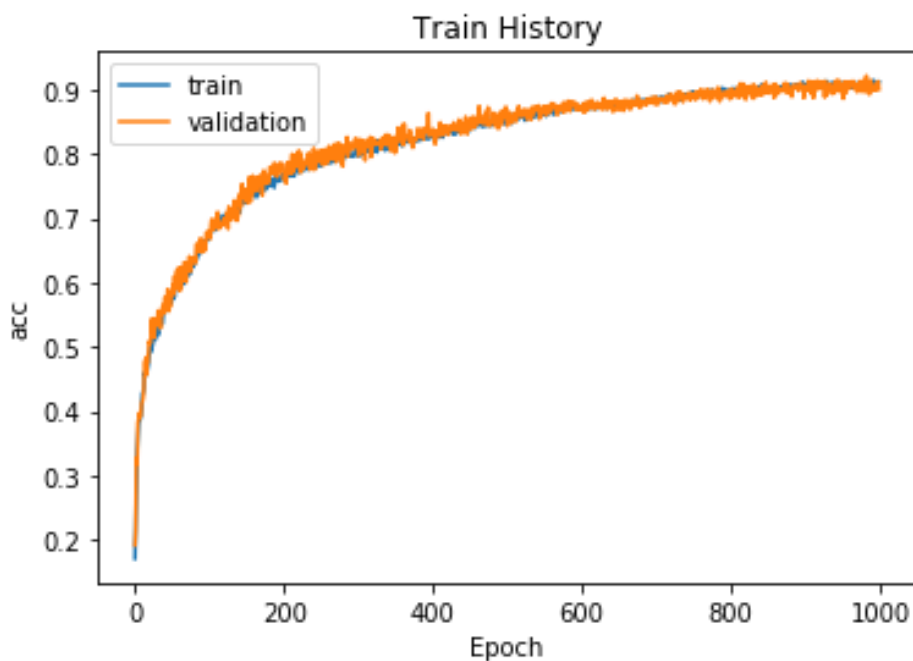


Fig. 17: The result of AI training

After 1,000 times of training, accuracy reached 91%. However, I am curious if this is the best we can get from the neural network.

Therefore, I decided to conduct systematic research to find the best configuration.

(III) Bike image recognition

It would be best to attach a photo when reporting the bike. It would be convenient for the cleaner to compare the photo to the spot where the bike is abandoned. The photo can be used for filtering the valid cases, which can be prioritized. It is easy to take a photo by phone. The user can take a picture of the abandoned bike and upload the photo easily.

Therefore, a photo is an important way for AI to determine if this reported case is qualified. It would be best if the system could determine if there is a bike in the photo or even if the bike is qualified for recycling. The following is our research process:

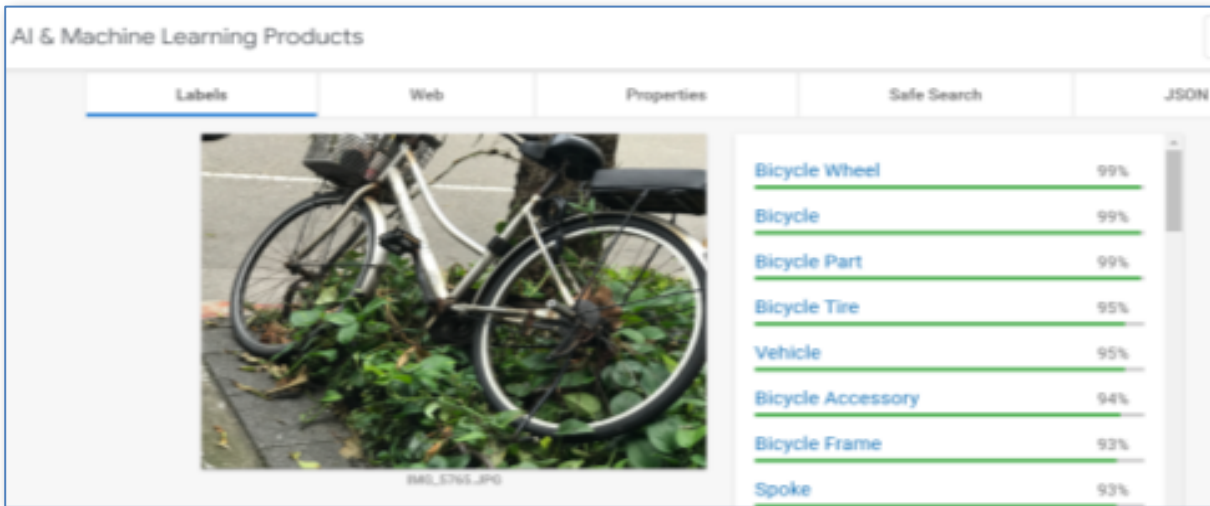


Fig. 18: The example for using Google Cloud Vision API to identify the bike

1. Collect the sample photos

This event has been held for three years. We have about 500 photos of bikes in all kinds of conditions. About 300 of them are photos of abandoned bikes and 200 of them are the ones for bikes in normal condition. In this study, to train the AI to identify abandoned bikes, I decided to collect more photos for the database. Students take pictures on the street during the holiday and get 500 photos of abandoned bikes and bikes in normal condition. They get about 1,000 photos of cars and scooters taken in daily life. There are 2,000 photos in total.

2. Evaluate the neural network model

After search and testing, I chose Google Cloud Vision API as the development tool. It is an API for photo analysis based on CNN neural network. (The following screenshot shows the testing example.) It is a trained AI that can identify lots of objects in the photo (including bike, car and scooter). Its output tag is used as the input layer of AI module “Determining the abandoned bike.”

The AI module “Possibility as an abandoned bike] is used as the output layer. There are several hidden layers used in the middle.

The advantage of Google Cloud Vision API is that if this post rating system is used in other applications, such as auction posting, this AI can also be responsible for judging whether the uploaded photo matches the topic. When auctioning on oven, there should be an oven in the photo. The AI should even compare the brand of the oven in the photo.

3. Connect to the external API

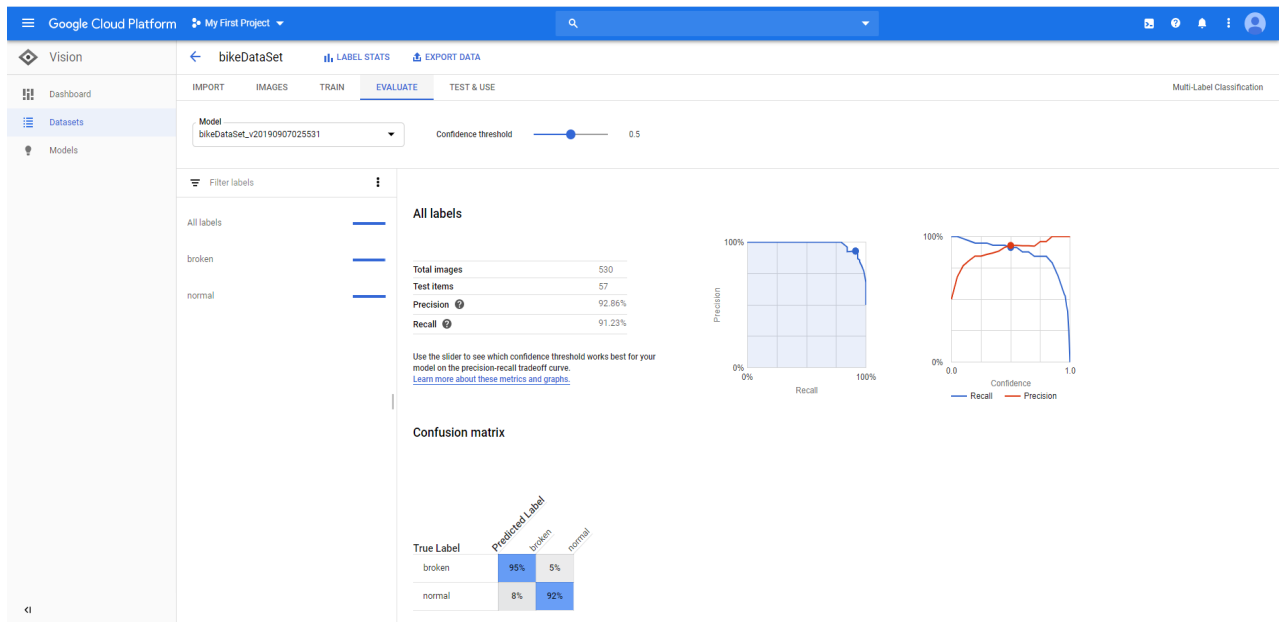
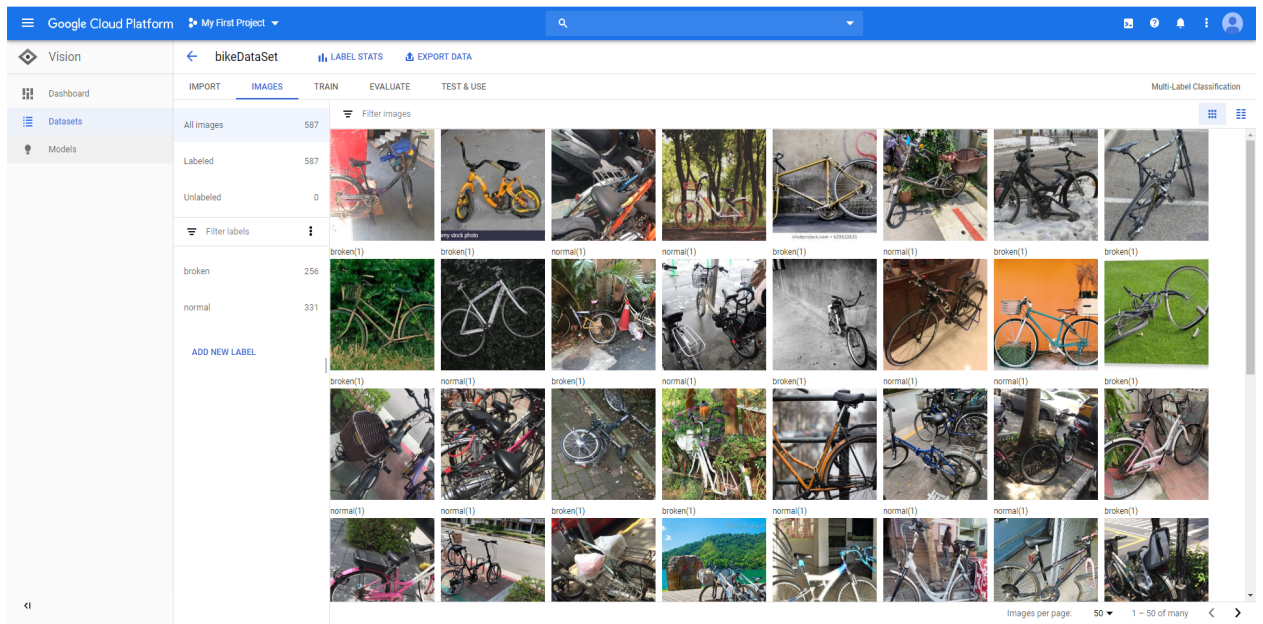
By referring to the Python example program provided by other authors on GitHub, I successfully connected Google Cloud Vision API to our MLP AI model.

4. Training and test verification

I used the organized 2,000 photos to train the AI. First of all, let the expert judge the rating (Is this an abandoned bike) for these photos. 0 means it is least likely an abandoned bike and 10 means it is most likely an abandoned bike. Next, use a 10x40x40x11 MLP neural network for training to complete the second AI application in this study. The main problem is that using AI to judge if there is a bike in the photo can achieve 95% accuracy. However, the accuracy is low if using AI to determine if a bike is abandoned. The main problem is the standard for judging if a bike is abandoned. I need to improve the program for this part.

5. Using AutoML to improve recognition rate

After the initial selection, I tried to use the previous model to improve the recognition rate but I don't see much improvement. So, I decided to recognize the bike in other ways. I used AutoML Vision of Google Cloud to train the new recognition tools. I used the bike parts to judge whether a bike was in good condition. Now I take a different approach. I use AutoML to check the overall condition of the bike. I use about 600 photos to train a new model. I can achieve 93% accuracy by using AutoML Vision.



(IV) Address analysis based on house number

The user mostly likely has a problem with entering the address when reporting. The user can use the locating function of Facebook to provide the address. However, some users cannot locate the location precisely by phone. The detected location is a bit far from the actual location. It would be very convenient if the user could take a picture of the doorplate when reporting and uploading the photo to the chatbot to automatically analyze the address. This can further confirm the validity of the case. It is the best of both worlds.

However, I searched the entire network and confirmed that no one had created the function to [analyze the address based on the photo of the doorplate]. Google claimed that it's working on the

development of doorplate recognition, and it can be used in the U.S. It can only analyze the house number instead of the complete address. I tried to use AI to analyze the address based on the doorplate. The result is as follows:

1. Collect the photos of the doorplates in all counties and cities

The first step of developing AI application is to collect training samples. It is not difficult to analyze the doorplate photo. I collected and took photos everywhere and collected over 300 photos of all kinds of doorplates in a month. There are certain formats of the doorplate. 300 photos are quite enough.

2. Evaluation model

After search and testing, Google Cloud Vision API is also used for the pre-processing image analysis tool of AI. I used this tool to convert the doorplate photo into numerous text sections and contents as follows:



Fig. 20: Use Google Cloud Vision API to test doorplate analysis.

Google Cloud Vision API cannot analyze the doorplate. It can only try to analyze the texts. However, there are lots of errors. Some texts cannot be recognized. If treating the recognized texts as the address, it is almost impossible for the cleaner to use it. For incomplete address, I only need to find the correct text sections and the correct order to use the keyword search function of Google Map API to provide a partial address. This tool then can look up a complete address including the county, city and district.

However, I need to find out the correct text sections and their order. I designed the third MLP model application in this study:

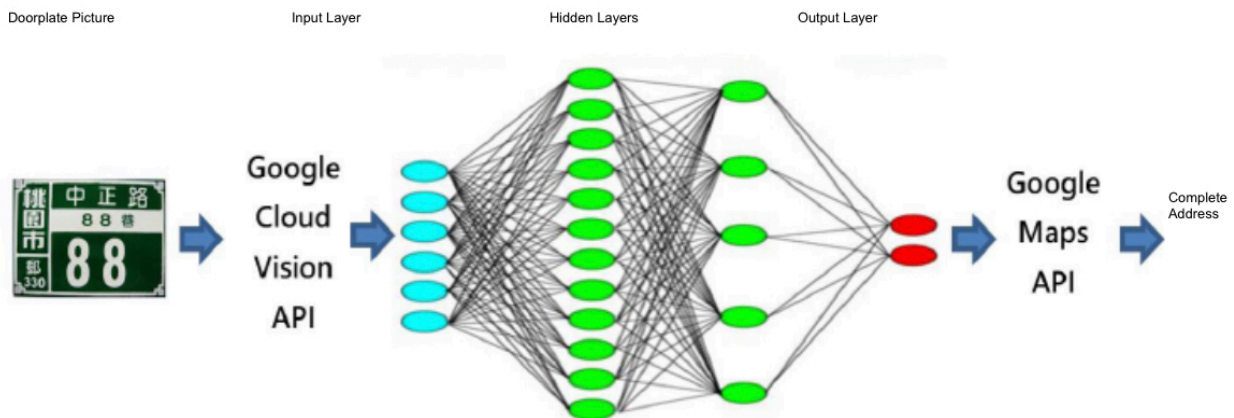


Fig. 21: The diagram for the AI module analyzing the doorplate in Taiwan

3. Connect API

This study also refers to the Python example program provided by other authors on GitHub to connect Google Cloud Vision API and Google Maps API. With the first one on the front and the latter one on the back, we connect them to our MLP AI model.

4. Training and verification

By using the 300 photos of the doorplate we collected, I obtained the text sections and contents analyzed by Google Cloud Vision API and used them as the input layer of MLP model. I used the AI to determine the useful text section and the correct order of the text and used them as the output layer of MLP model.

Next, I used a 10x40x40x11 MLP neural network for training to complete the third AI application in this research. Due to inconsistency of the doorplate format in different counties and cities, the experiment result only shows about 50% of the accuracy. I am still trying to increase accuracy.

(V) The research for the number of neurons in the hidden layer in the MLP post rating network

In this study, I observed the training result provided by Tensorflow. Besides accuracy, I can also find out the entire training curve, discrepancy of rating (difference between AI rating and

expert rating), average error score...etc. I can observe the pros and cons of different network structures. I can find a pattern by using different numbers of hidden layers and hidden neurons for the experiment repeatedly.

1. Experiment design

According to Research 2, we slightly adjusted the parameter settings as below:

- Batch_Size is fixed to 40 (about 1% of the total number of training samples).
- 500-2000 epochs are done for each experiment (number of times of training). I can stop when the curve becomes stable.
- 80% of the samples in the sample database are used as the training samples and 20% of them are used as the verification samples.

I test 0 hidden layers to 5 hidden layers. In each hidden layer, I start from 10 neurons. I add the number of neurons at an equal ratio (10, 20, 40, 80....) until the number becomes stable.

2. Experiment environment

I used Jupiter Notebook to execute Keras_recycle0209.py to use the computer to simulate neural network computing. Based on the parameter setting, I read the sample in the sample database step2-Build_questionary_5500_0208.csv to train the MLP neural network. I used the expert rating of the sample as the feedback of the neural network to adjust the weight parameters between neurons in all layers. It is like how a human brain learns new skills. By trial and error, I strengthened and weakened certain connections among neurons. Eventually, I use the strong and weak connections among neurons to present the logic of this skill.

This program automatically organizes the process and result of the training and it would be convenient to organize an experiment report.

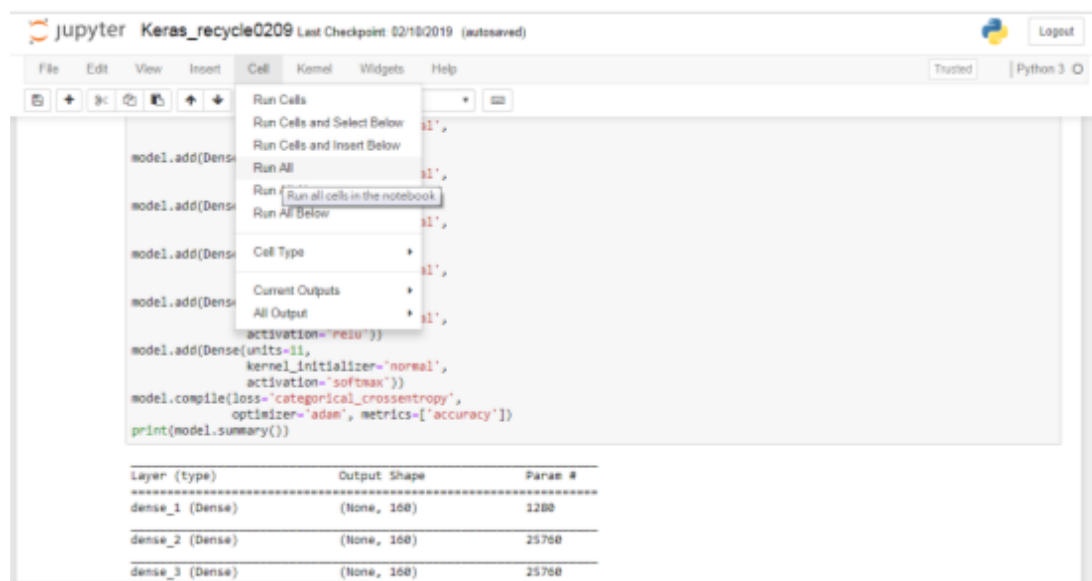


Fig. 22: The diagram for using the AI training program

3. Data analysis

I calculated the “weights parameter number” of different layers and number of neurons and used them as the computing for each neural network. I extracted the important parameters of training results, including accuracy, error and the number of samples with 2 points of difference or above. This result is used as the computing effectiveness of each neural network. With the cost and effectiveness, I can compare and analyze the quantified efficiency for each parameter setting. Therefore, we would be able to find out what kind of neural network structure can achieve the best effect with the minimum computation.

4. Discussion on the result

I performed experiments for 27 types of neuron structures and quantities. I selected 0 to 5 hidden layers and there are 0 to 3200 neurons on each layer. I can calculate the total number and weight of neurons (computation). I generated complete experimental data from each experiment and computed accuracy and average error. I selected 12 iconic experiment results, explained them, and discussed them in detail. For the ones we chose, the first experiment result of each hidden layer (minimum total number of neurons) and the first section in the hidden layer achieving 99% and above (99% accuracy and above achieved for the minimum total number of neurons). I also

analyzed the section achieving 100% accuracy. I use a table to organize and present the 27 experiment parameters and results. The 12 iconic experiment results are as follows:

(I) Experiment 1: There is no hidden layer in the MLP neural network. The neural network structure is 7x11. There are 18 neurons. The adjustable weight parameters are 88 (7+1) x11=88).

1. The result of the last batch of training:

Epoch 500/500

- 0s - loss: 1.4426 - acc: 0.4539 - val_loss: 1.4655 - val_acc: 0.4627

2. After the training, the overall evaluation accuracy is 45.9% and the average error is 1.45.

3. Accuracy and error curve:

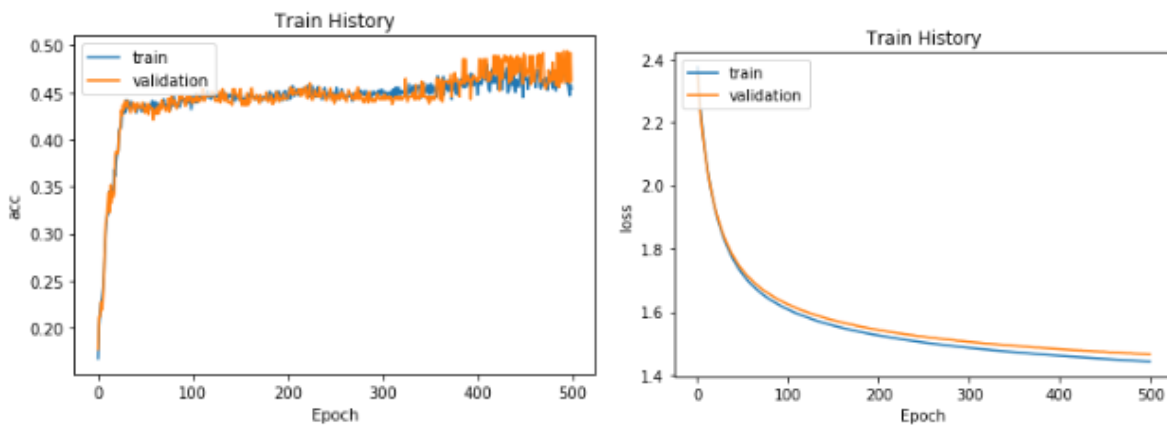


Fig. 23: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 2,978 samples. 1,716 samples have over 1 point of difference and 787 of them have over 2 points of difference.

5. After eliminating repeated samples, there are discrepancies in the predicted value of 180 samples. The predicted value of 72 of them has over 2 points of difference. They are listed below (the first five samples are summarized here):

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	0	-1	0	5.8	16	2	1	5	8	3.0
1	0	-1	1	5.9	16	0	1	5	8	3.0
2	1	1	0	3.2	2	0	1	9	5	4.0
3	1	0	1	5.9	1	0	1	4	7	3.0

4	0	-1	1	0.0	0	0	1	8	1	7.0
---	---	----	---	-----	---	---	---	---	---	-----

6. Result analysis

With a sufficient number of rounds, the accuracy of AI rating is 45.9% and the discrepancy is excessive. The observation learning curve fluctuates continuously but it cannot be improved further. This shows that the MLP network cannot learn this sample effectively. It could be that the cranial capacity is too small. The AI cannot learn.

(II) Experiment 2: There is a hidden layer in the MLP neural network (10 neurons available). The neural network structure is 7x10x11. There are 28 neurons. There are 201 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 1000/1000

- 1s - loss: 0.4755 - acc: 0.8557 - val_loss: 0.5067 - val_acc: 0.8491

2. After the training, the overall evaluation accuracy is 85.3% and the average error is 0.48.

3. Accuracy and error curve:

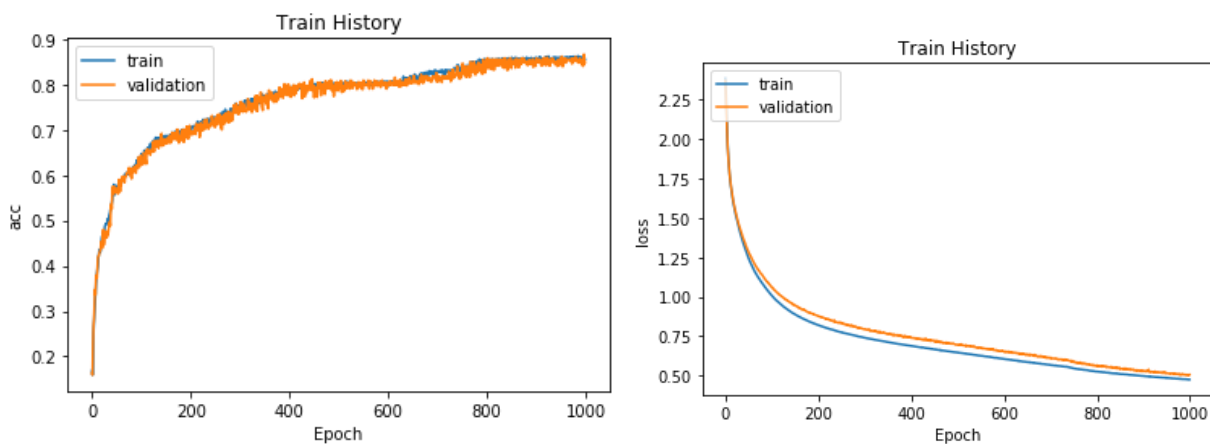


Fig. 24: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 808 samples. 330 samples have over 1 point of difference and 226 of them have over 2 points of difference.

5. After eliminating repeated samples, there are discrepancies in the predicted value of 60 samples. 24 of them have over 2 points of difference. They are listed below (the first five samples are summarized here):

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	1	1	0	3.2	2	0	1	9	6	3.0
1	0	-1	1	0.0	0	0	1	8	1	7.0
2	0	-1	1	3.1	16	0	1	0	3	3.0
3	1	1	0	0.3	2	0	1	5	2	3.0
4	0	-1	1	2.6	8	1	1	0	4	4.0

6. Result analysis

With a sufficient number of rounds, the accuracy of AI rating is increased to 85% but the discrepancy is excessive. There are too many deviations with large gaps. This shows that the MLP network still cannot learn this sample effectively.

(III) Experiment 5: There is a hidden layer in the MLP neural network (200 neurons available). The neural network structure is 7x200x11. There are 218 neurons. There are 3811 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 500/500

- 0s - loss: 0.0459 - acc: 0.9873 - val_loss: 0.0532 - val_acc: 0.9918

2. After the training, the overall evaluation accuracy is 99.4% and the average error is 0.05.

3. Accuracy and error curve:

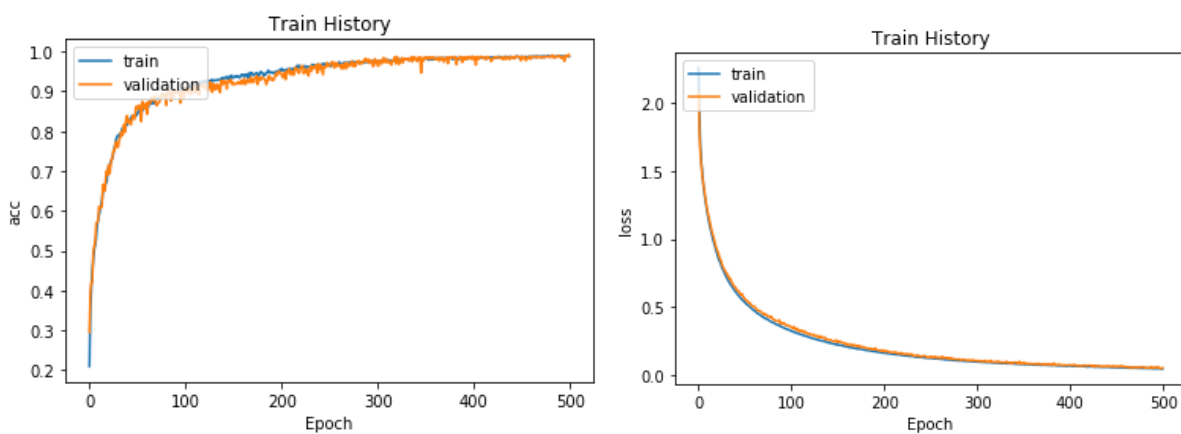


Fig. 25: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 35 samples. 8 samples have over 1 point of difference, and 7 have over 2 points of difference.

5. After eliminating repeated samples, there are discrepancies in the predicted value of 4 samples. They are listed below:

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	0	-1	0	9.8	2	0	1	10	9	1.0
1	0	-1	1	7.8	2	0	1	8	7	1.0
3	1	1	1	0.0	0	0	1	10	7	3.0
30	1	1	0	0.0	0	0	1	10	8	2.0

(IV) Experiment 9: There are 2 hidden layers in the MLP neural network (5 neurons available for each). The neural network structure is 7x5x5x11. There are 28 neurons. There are 136 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 500/500

- 0s - loss: 0.7571 - acc: 0.7055 - val_loss: 0.7782 - val_acc: 0.7209

2. After the training, the overall evaluation accuracy is 70.7% and the average error is 0.76.

3. Accuracy and error curve:

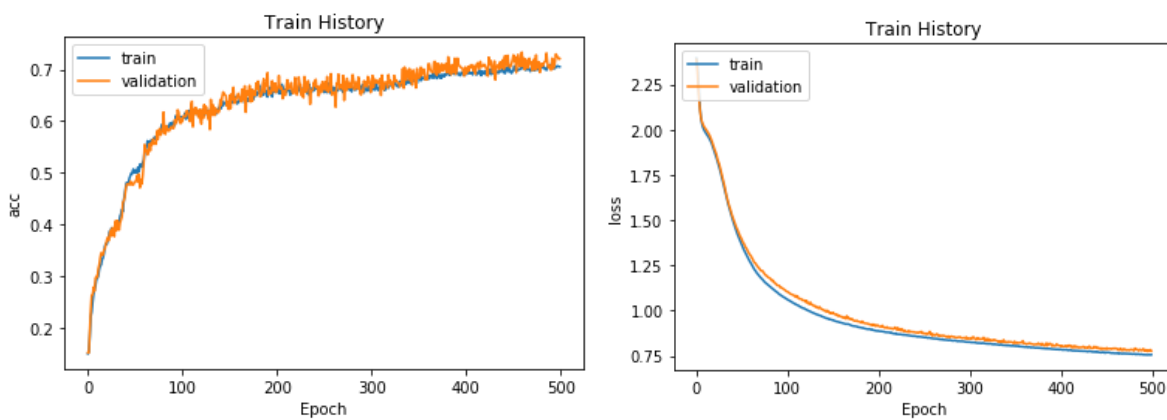


Fig. 26: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 1,608 samples. 549 samples have over 1 point of difference and 256 of them have over 2 points of difference.

5. After eliminating repeated samples, there are discrepancies in the predicted value of 85 samples. 14 of them have over 2 points of difference. They are listed below (the first five samples are summarized here):

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	1	1	0	3.2	2	0	1	9	6	3.0
1	0	-1	1	0.0	0	0	1	8	1	7.0
3	1	1	0	0.3	2	0	1	5	2	3.0
5	0	-1	0	0.0	0	0	1	8	1	7.0
6	1	1	1	0.0	0	0	1	10	1	9.0

(V) **Experiment 11:** There are 2 hidden layers in the MLP neural network (20 neurons available for each). The neural network structure is 7x20x20x11. There are 58 neurons. There are 811 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 2000/2000

- 0s - loss: 0.0020 - acc: 1.0000 - val_loss: 0.0106 - val_acc: 0.9991

2. After the training, the overall evaluation accuracy is 99.99% and the average error is 0.004.

3. Accuracy and error curve:

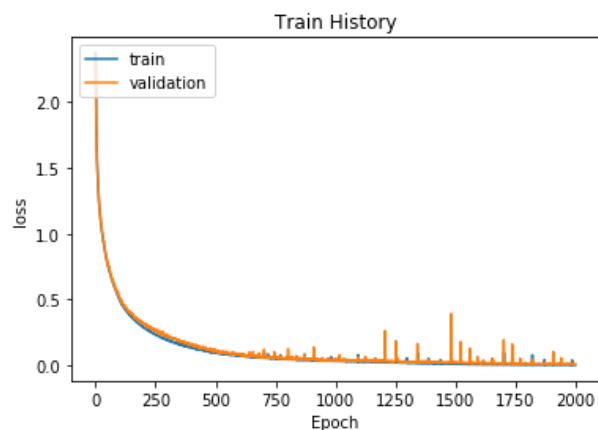
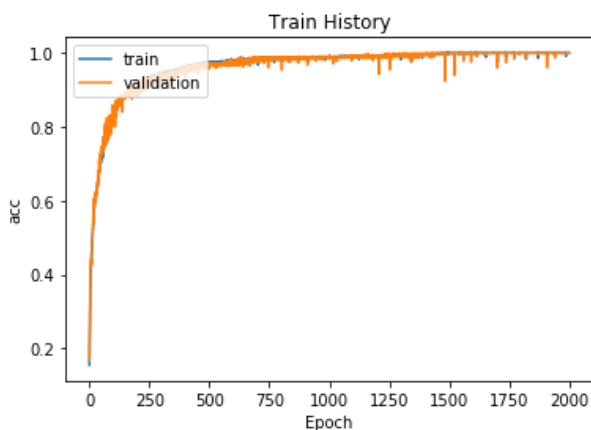


Fig. 27: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 1 sample:

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	1	1	0	0.0	0	0	1	10	4	6.0

(VI) Experiment 16: There are 3 hidden layers in the MLP neural network (20 neurons available for each). The neural network structure is 7x20x20x20x11. There are 78 neurons. There are 1,231 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 2000/2000

- 0s - loss: 0.0278 - acc: 0.9923 - val_loss: 0.0778 - val_acc: 0.9873

2. After the training, the overall evaluation accuracy is 99.1% and the average error is 0.04.

3. Accuracy and error curve:

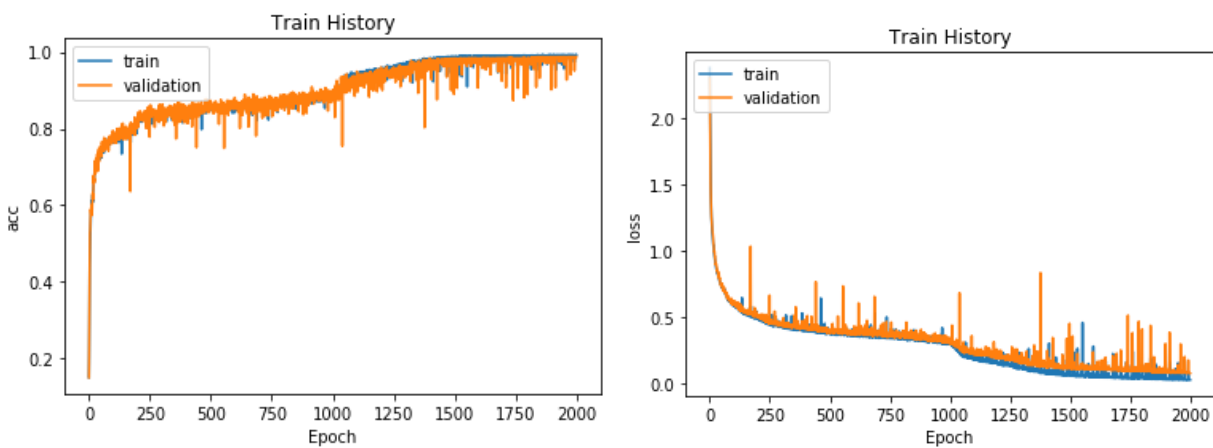


Fig. 28: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 47 samples. 38 samples have over 1 point of difference and 37 of them have over 2 points of difference.

5. After eliminating repeated samples, there are discrepancies in the predicted value of 13 samples. 10 of them have over 2 points of difference. They are listed below:

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	0	-1	0	0.0	0	0	1	8	1	7.0
1	1	0	1	0.0	1	0	1	0	7	7.0
4	1	0	1	0.3	2	0	1	0	7	7.0
5	1	0	0	0.0	1	0	1	0	7	7.0
7	0	-1	1	3.2	4	0	1	0	3	3.0
10	0	-1	0	0.0	2	0	1	0	8	8.0
14	0	-1	1	0.0	2	0	1	0	8	8.0
30	1	1	0	0.0	0	0	1	10	5	5.0
31	0	-1	1	2.8	100	1	1	0	3	3.0
34	0	-1	0	2.9	100	4	1	0	3	3.0

(VII) Experiment 19: There are 3 hidden layers in the MLP neural network (80 neurons available for each). The neural network structure is 7x80x80x80x11. There are 258 neurons. There are 14,491 adjustable weight parameters.

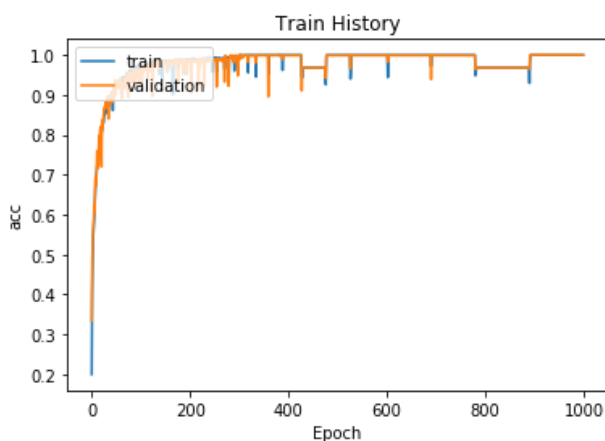
1. The result of the last batch of training:

Epoch 1000/1000

- 0s - loss: 2.8901e-06 - acc: 1.0000 - val_loss: 5.3457e-04 - val_acc: 1.0000

2. After the training, the overall evaluation accuracy is 100.0% and the average error is 0.0001.

3. Accuracy and error curve:



30

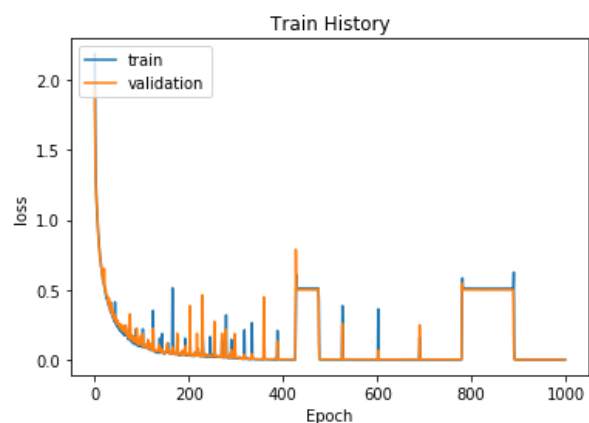


Fig. 29: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there is no discrepancy.

5. Result analysis:

This neural network has no discrepancy. This means that it can fully understand the knowledge in the sample database at this scale and it can judge as accurately as an expert.

(VIII) Experiment 20: There are 4 hidden layers in the MLP neural network (40 neurons available for each). The neural network structure is 7x40x40x40x11. There are 178 neurons. There are 5,691 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 2000/2000

- 0s - loss: 5.4760e-05 - acc: 1.0000 - val_loss: 0.0094 - val_acc: 0.9991

2. After the training, the overall evaluation accuracy is 99.98% and the average error is 0.002.

3. Accuracy and error curve:

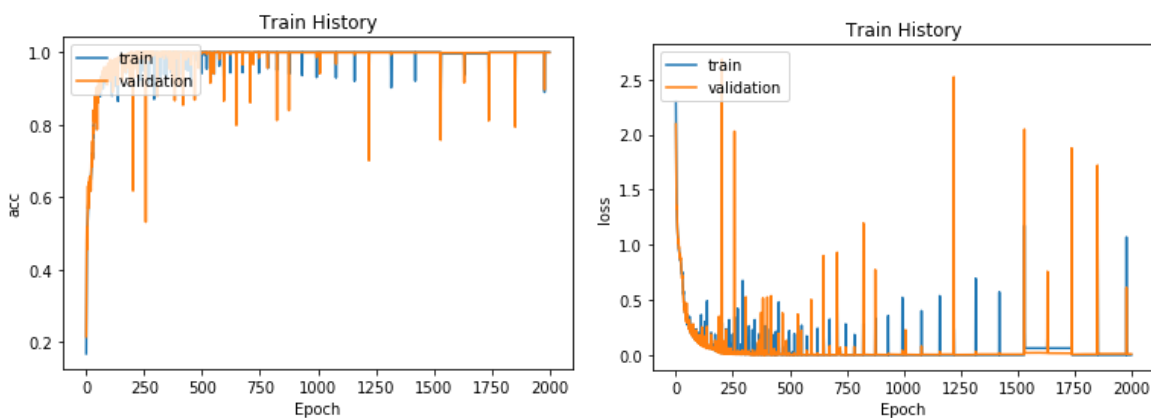


Fig. 30: Number of rounds-Accuracy and number of rounds-error curve

1. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 1 sample:

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	1	1	0	0.0	0	0	1	10	8	2.0

(IX) Experiment 22: There are 4 hidden layers in the MLP neural network (80 neurons available for each). The neural network structure is 7x80x80x80x11. There are 338 neurons. There are 20,971 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 2000/2000

- 0s - loss: 0.0110 - acc: 0.9993 - val_loss: 0.0561 - val_acc: 0.9964

2. After the training, the overall evaluation accuracy is 99.9% and the average error is 0.02.

3. Accuracy and error curve:

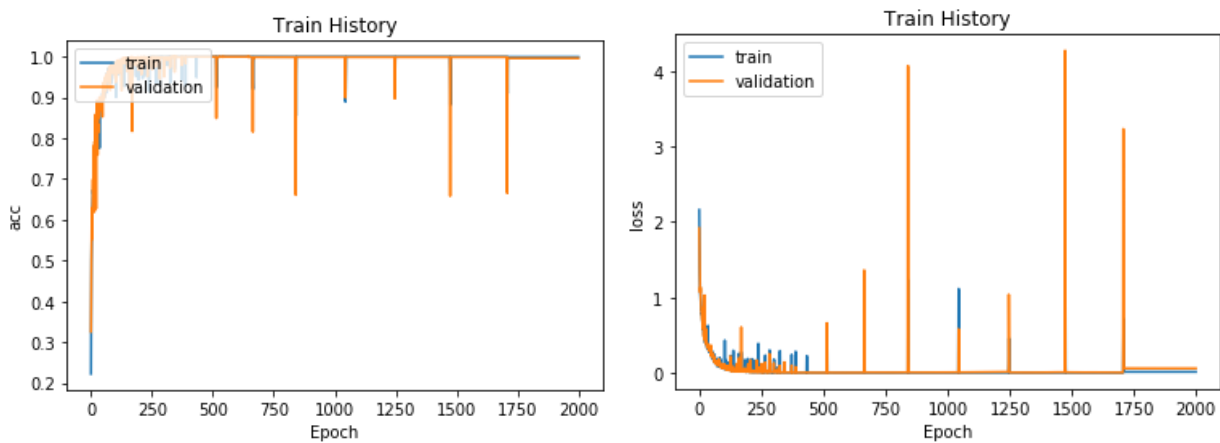


Fig. 31: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 7 samples. After eliminating repeated samples, there are discrepancies in the predicted value of 2 samples. They are listed below:

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	1	0	1	8.0	16	0	1	10	9	1.0

5	1	1	0	0.0	0	0	1	10	8	2.0
---	---	---	---	-----	---	---	---	----	---	-----

(X) Experiment 23: There are 5 hidden layers in the MLP neural network (10 neurons available for each). The neural network structure is 7x10x10x10x10x11. There are 68 neurons. There are 641 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 2000/2000

- 0s - loss: 0.0469 - acc: 0.9877 - val_loss: 0.1012 - val_acc: 0.9773

2. After the training, the overall evaluation accuracy is 98.6727% and the average error is 0.057485.

3. Accuracy and error curve:

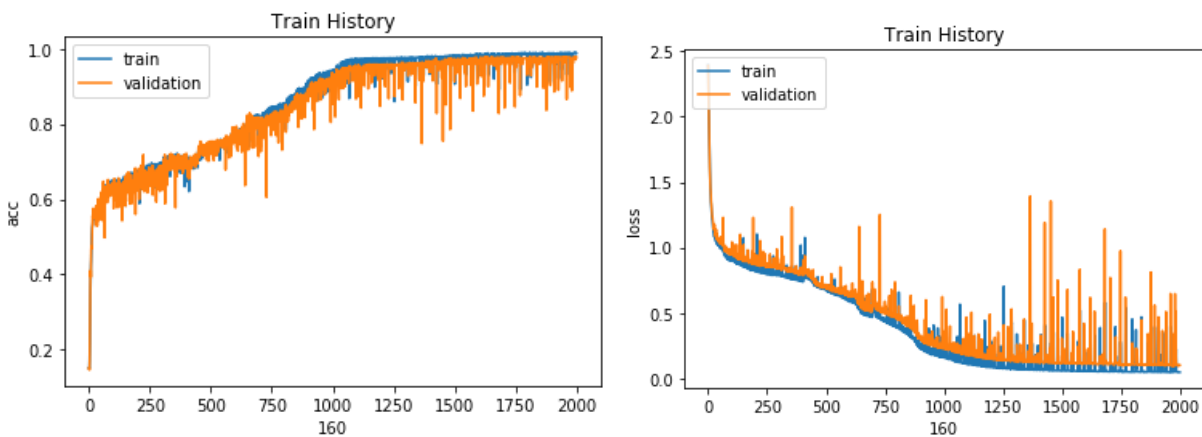


Fig. 32: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 73 samples. 58 samples have over 1 point of difference and 57 of them have over 2 points of difference.

5. After eliminating repeated samples, 7 samples have over 2 points of difference. They are listed below:

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	0	-1	1	0.0	0	0	1	8	1	7.0
2	0	-1	0	0.0	0	0	1	8	1	7.0
3	1	1	1	0.0	0	0	1	10	7	3.0
20	0	-1	0	2.8	8	0	1	0	7	7.0
39	0	-1	1	0.0	8	1	1	0	7	7.0
41	0	-1	1	0.0	16	0	1	0	7	7.0
59	1	1	0	0.0	0	0	1	10	7	3.0

(XI) Experiment 26: There are 5 hidden layers in the MLP neural network (80 neurons available for each). The neural network structure is 7x80x80x80x80x80x11. There are 418 neurons. There are 27,451 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 2000/2000

- 0s - loss: 1.1921e-07 - acc: 1.0000 - val_loss: 5.6770e-05 - val_acc: 1.0000

2. After the training, the overall evaluation accuracy is 100.0% and the average error is 1.1e-05.

3. Accuracy and error curve:

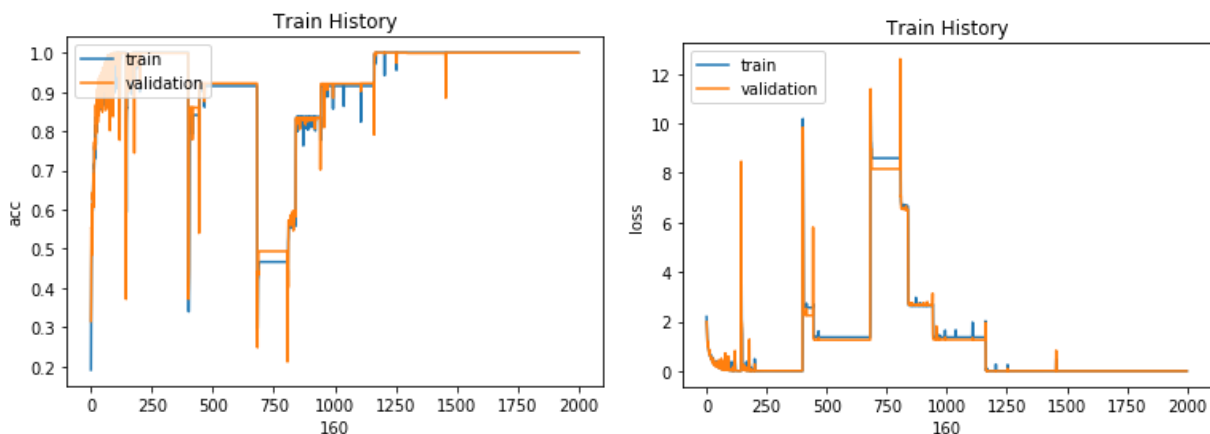


Fig. 33: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 0 samples. 0 samples have over 1 point of difference and 0 of them have over 2 points of difference.

5. Result analysis:

This neural network has no discrepancy. This means that it can fully understand the knowledge in the sample database at this scale and it can judge as accurately as an expert.

(XII) Experiment 27: There are 5 hidden layers in the MLP neural network (160 neurons available for each). The neural network structure is 7x160x160x160x160x160x11. There are 818 neurons. There are 106,091 adjustable weight parameters.

1. The result of the last batch of training:

Epoch 2000/2000

- 1s - loss: 1.1921e-07 - acc: 1.0000 - val_loss: 0.0025 - val_acc: 0.9991

2. After the training, the overall evaluation accuracy is 99.98179999999999% and the average error is 0.000496.

3. Accuracy and error curve:

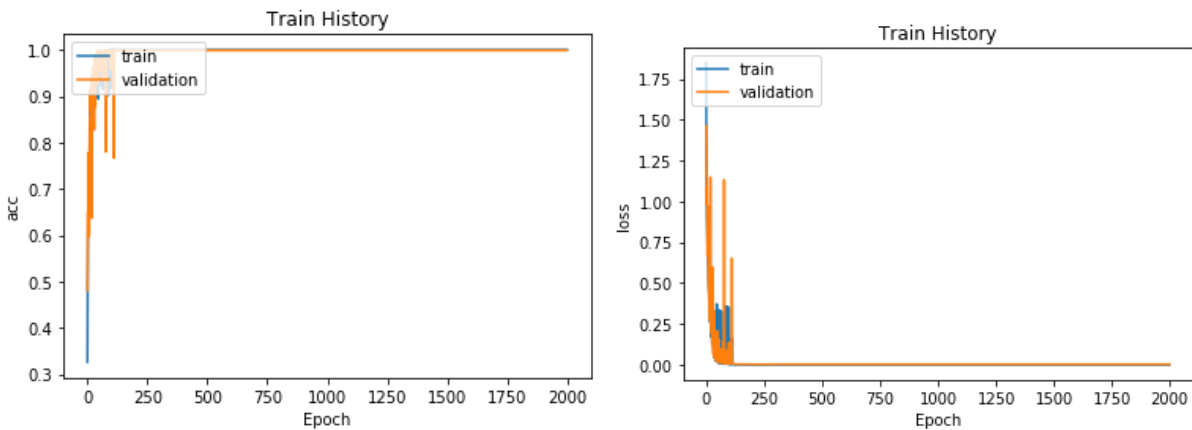


Fig. 34: Number of rounds-Accuracy and number of rounds-error curve

4. I reviewed the discrepancy of the expert and AI rating of all the samples. I found out that there are discrepancies for 1 sample.

Serial No.	A photo is available	The photo is accurate	New user	Credit	Total number of times	Number of repeated times	Location	Rating	AI	Discrepancy
0	1	1	0	0.0	0	0	1	10	8	2.0

(XIII) Summary of experimental data

The following is the parameter and data of the 27 experiments, including the 12 iconic ones. The results of the 12 experiments above are marked in red.

Experiment No.	Number of hidden layers	Number of neurons on each layer	Total number of neurons	Weight (computation)	Accuracy	Average error
1	0	0	18	88	46.27%	1.4655
2	1	10	28	201	84.91%	0.5067
3	1	50	68	961	95%	0.2196
4	1	100	118	1911	97.45%	0.1268
5	1	200	218	3811	99.18%	0.0532
6	1	400	418	7611	99.27%	0.0292
7	1	800	818	15211	99.73%	0.0212
8	1	3200	3218	60811	99.82%	0.0195
9	2	5	28	136	72.09%	0.7782
10	2	10	38	311	88.27%	0.3839
11	2	20	58	811	99.91%	0.0106
12	2	40	98	2411	99.91%	0.0091
13	2	80	178	8011	99.91%	0.0146
14	2	160	338	28811	99.91%	0.0113
15	2	320	658	108811	99.91%	0.0147
16	3	10	48	421	99.36%	0.0268
17	3	20	78	1231	99.73%	0.0117
18	3	40	138	4051	99.91%	0.0042
19	3	80	258	14491	100%	0.0005
20	4	20	98	1651	99%	0.02767
21	4	40	178	5691	99.91%	0.0094
22	4	80	338	20971	99.64%	0.0561
23	5	10	68	641	97.73%	0.1012

24	5	20	118	2071	96.09%	0.1967
25	5	40	218	7331	98.36%	0.0713
26	5	80	418	27451	100%	0.00006
27	5	160	818	106091	99.91%	0.0025

Table 1: The important parameter and data summary of all the 27 experiments

Table 1 shows that in the model designed for different layers, I can achieve the same ideal training result (99.9% and above accuracy). The weight (computation) needed can be high and low. Table 2 is the summary of Table 1:

	The minimum weight (computation) with accuracy reaching 99.9% and above
0 hidden layer	Cannot be achieved
1 hidden layer	Cannot be achieved for 60,811
2 hidden layers	881
3 hidden layers	4051
4 hidden layers	5691
5 hidden layers	27451

Table 2: Comparison between the number of different hidden layers

Table 2 shows that two hidden layers would be the best. It only takes 20 neurons on each layer to achieve 99.91% accuracy. However, in 3 hidden layers, 100% accuracy can be achieved with 80 neurons for each layer. The 2 hidden layers cannot achieve such accuracy. The 4 hidden layers perform stably. However, the training process and result for 5 hidden layers show repetitive fluctuation.

Therefore, the conclusion is that to achieve 99.9% accuracy, **it would be best to have 2 to 4 hidden layers. It would be best to have 1 to 10 times the neurons on each layer (the average**

number of neurons on input and output layers). We call this assumed model the “Cyclone Conjecture.” We will apply this model in other data analyses to verify the feasibility of this model.

Before I used the expert rating samples for the MLP neural network to learn from them, I tried to fill out the rating by random. No matter how many layers and neurons we used, the learning effect is poor and the accuracy stays below 30% or even below 20%. If there are many neurons, the network can memorize the training samples but cannot react to the test samples (which means overfitting). However, when I used the actual expert rating samples, I found out that the MLP neural network can learn way more effectively than expected. I worried that expert rating is subjective and there is no rule for this and the neural network might not be able to learn well, but it turned out that it can achieve 99% accuracy for the test sample. This means that the accuracy of the rating sample is also very important.

(VI) The research for using the “Cyclone Conjecture” model for other different data

To verify the feasibility of this model, I chose the five most popular social media and apps, Facebook, Instagram, Youtube, TikTok and Weibo. We collected user data and collected 1,000 entries from each website. Then, I defined input and output parameters. Eventually, I used Tensorflow to experiment with different hidden layers and numbers of neurons and observed and analyzed the result. In this study, I observed the training result provided by Tensorflow. Besides accuracy, I can also find out the entire training curve, discrepancy of rating (difference between AI rating and expert rating), average error score...etc. I can observe the pros and cons of different network structures. I can find a pattern by using different numbers of hidden layers and hidden neurons for the experiment repeatedly.

1. Experiment design

To verify the feasibility of this model, I chose the five most popular social media and apps. The experiment is designed as follows:

- The social media and apps are Facebook, Instagram, Youtube, TikTok and Weibo.
- I collected 1,000 entries from each website. Then, I defined input and output parameters.
- 500-2000 epochs are done for each experiment (number of times of training). We can stop when the curve becomes stable.

- 80% of the samples in the sample database are used as the training samples and 20% of them are used as the verification samples.

2. Input and output layers

I defined the input and output layers of these five websites in the table below. Instagram has 6 input layers and 8 output layers. The rest of them have 7 input and output layers, respectively. Therefore, I can define that the average number of neurons for input and output layers would be 8, respectively. I test 1 hidden layer to 5 hidden layers. In each hidden layer, we start from 3 neurons. We add the number of neurons at an equal ratio (3, 7, 15, 40, 70 and 120...) until the number becomes stable.

Category	Input							Output	
Instagram	# of Followers	Total Posts	Avg Likes in Last 5 Posts	Face(0-1)	# Video (0-10)	# Posts in Past 7 Days		# of Likes	8 Neurons
Facebook	Photo (0-1)	Face(0-1)	Video (0-1)	Location (0-1)	Link (0-1)	Avg Likes in Past 5 Posts	# Posts in Past 7 Days	# of Likes	7 Neurons
Youtube	Subscribers	View	View-1	View-2	View-3	View-4	View-5	# of Views	7 Neurons
Tik Tok	Followers	View	View-1	View-2	View-3	View-4	View-5	# of Views	7 Neurons
Weibo	Followers	# of Posts	Photos (0-9)	Video (0-1)	Face(0-1)	Likes -1	Likes -2	# of Likes	7 Neurons

Table 3: The input and output layers of these five websites

3. Experiment result

The five tables below show the experiment data of 1,000 entries in the five groups. They show the programming result for the data of these five websites. I calculated the “weights parameter number” of different layers and number of neurons and used them as the computing for each neural network. I extracted the important parameters of training result, including accuracy, error and computation. This result is used as the computing effectiveness of each neural network. With the cost and effectiveness, I can compare and analyze the quantified efficiency for each parameter setting. Therefore, I would be able to find out what kind of neural network structure can achieve the best effect with the minimum computation. The part marked in red means the experiment result at 2 to 4 hidden layers and the multiple (1-10) of the average of the number of neurons on the input and output layers. This experiment result shows that in this interval, I can achieve the best accuracy with minimum computation.

FB	Neurons Per Layer	3	7	15	40	70	120
1 Hidden Layer	# of Weights	52	112	232	607	1,057	1,807
	Accuracy	71.00%	73.70%	76.40%	78.20%	81.20%	81.00%
2 Hidden Layer	# of Weights	64	168	472	2,247	6,027	16,327
	Accuracy	74.20%	76.00%	79.40%	81.70%	81.50%	82.00%
3 Hidden Layer	# of Weights	76	224	712	3,887	10,997	30,847
	Accuracy	75.10%	80.10%	79.70%	80.00%	80.80%	83.00%
4 Hidden Layer	# of Weights	88	280	952	5,527	15,967	45,367
	Accuracy	19.50%	71.70%	74.90%	81.40%	83.10%	84.60%
5 Hidden Layer	# of Weights	100	336	1,192	7,167	20,937	59,887
	Accuracy	19.50%	73.00%	79.70%	82.10%	82.30%	84.50%

IG	Neurons Per Layer	3	7	15	40	70	120
1 Hidden Layer	# of Weights	53	113	233	608	1,058	1,808
	Accuracy	54.90%	59.00%	64.70%	71.00%	74.10%	75.30%
2 Hidden Layer	# of Weights	65	169	473	2,248	6,028	16,328
	Accuracy	54.10%	72.80%	76.50%	83.30%	82.10%	84.60%
3 Hidden Layer	# of Weights	77	225	713	3,888	10,998	30,848
	Accuracy	12.50%	54.30%	76.20%	81.40%	85.20%	83.10%
4 Hidden Layer	# of Weights	89	281	953	5,528	15,968	45,368
	Accuracy	44.10%	54.10%	73.70%	79.40%	86.30%	86.50%
5 Hidden Layer	# of Weights	101	337	1,193	7,168	20,938	59,888
	Accuracy	12.50%	69.10%	72.80%	81.60%	85.90%	85.30%

TikTok	Neurons Per Layer	3	7	15	40	70	120
1 Hidden Layer	# of Weights	52	112	232	607	1,057	1,807
	Accuracy	34.60%	40.80%	42.60%	46.70%	49.30%	53.30%
2 Hidden Layer	# of Weights	64	168	472	2,247	6,027	16,327
	Accuracy	44.60%	49.70%	68.30%	77.10%	81.74%	79.50%
3 Hidden Layer	# of Weights	76	224	712	3,887	10,997	30,847
	Accuracy	14.30%	68.20%	83.60%	82.98%	80.87%	85.30%
4 Hidden Layer	# of Weights	88	280	952	5,527	15,967	45,367
	Accuracy	14.30%	14.30%	82.00%	84.35%	84.97%	86.50%
5 Hidden Layer	# of Weights	100	336	1,192	7,167	20,937	59,887
	Accuracy	14.30%	14.30%	14.30%	85.90%	80.30%	76.40%

YT	Neurons Per Layer	3	7	15	40	70	120
1 Hidden Layer	# of Weights	52	112	232	607	1,057	1,807
	Accuracy	34.80%	41.60%	41.10%	44.30%	42.80%	47.70%
2 Hidden Layer	# of Weights	64	168	472	2,247	6,027	16,327
	Accuracy	47.20%	48.60%	53.40%	54.40%	56.20%	56.90%
3 Hidden Layer	# of Weights	76	224	712	3,887	10,997	30,847
	Accuracy	16.70%	56.60%	52.90%	53.70%	55.40%	57.10%
4 Hidden Layer	# of Weights	88	280	952	5,527	15,967	45,367
	Accuracy	46.20%	16.70%	55.50%	49.80%	56.90%	56.50%
5 Hidden Layer	# of Weights	100	336	1,192	7,167	20,937	59,887
	Accuracy	16.70%	49.00%	48.90%	48.60%	48.90%	50.80%

Weibo	Neurons Per Layer	3	7	15	40	70	120
1 Hidden Layer	# of Weights	52	112	232	607	1,057	1,807
	Accuracy	51.50%	57.80%	60.30%	63.10%	69.50%	71.40%
2 Hidden Layer	# of Weights	64	168	472	2,247	6,027	16,327
	Accuracy	60.90%	64.90%	76.30%	75.00%	75.70%	77.10%
3 Hidden Layer	# of Weights	76	224	712	3,887	10,997	30,847
	Accuracy	64.30%	68.10%	76.30%	78.50%	76.20%	78.90%
4 Hidden Layer	# of Weights	88	280	952	5,527	15,967	45,367
	Accuracy	16.70%	61.90%	73.20%	77.40%	76.20%	77.50%
5 Hidden Layer	# of Weights	100	336	1,192	7,167	20,937	59,887
	Accuracy	16.70%	68.40%	77.60%	77.40%	76.70%	78.70%

Table 4: All the testing results of the five websites

4. Discussion on the result

I performed experiments for 5 types of data and 125 groups with different neuron structures and the number of neurons. I selected 1 to 5 hidden layers and there are 3 to 140 neurons on each layer. I can calculate the total number and weight of neurons (computation). I generated complete experimental data from each experiment and computed accuracy and average error. I selected 15 iconic experiment results and they are listed below.

Experiment Type	# of Hidden Layer	Neurons per Layer	Total Neurons	Computational Weight	Accuracy	Average Score Error
FB	2	15	44	472	79.4	0.55
FB	3	7	35	224	80.1	0.55
FB	4	15	74	952	74.9	0.51
IG	2	40	94	2248	83.3	0.48
IG	3	40	134	3888	81.4	0.51
IG	4	40	174	5528	79.4	0.49
YT	2	15	44	472	53.4	1.14
YT	3	7	35	224	56.6	1.09
YT	4	15	74	952	55.5	1.12
Tiktok	2	40	94	2247	77.1	0.52
Tiktok	3	15	59	712	83.6	0.53
Tiktok	4	15	74	952	82.0	0.41
Weibo	2	15	44	472	76.3	0.62
Weibo	3	15	59	712	76.3	0.63
Weibo	4	40	174	952	73.2	0.61

Table 5: The best testing results of the five websites

I chose the best experiment result from each group and made a graph for each. From the first row from left to right: Facebook, Instagram, Youtube. From the first row from left to right: Tiktok, Weibo. The experiment result shows that besides YouTube, the accuracy for the rest of them can achieve 75% and above. This shows that this model will be useful and can be used in the future.

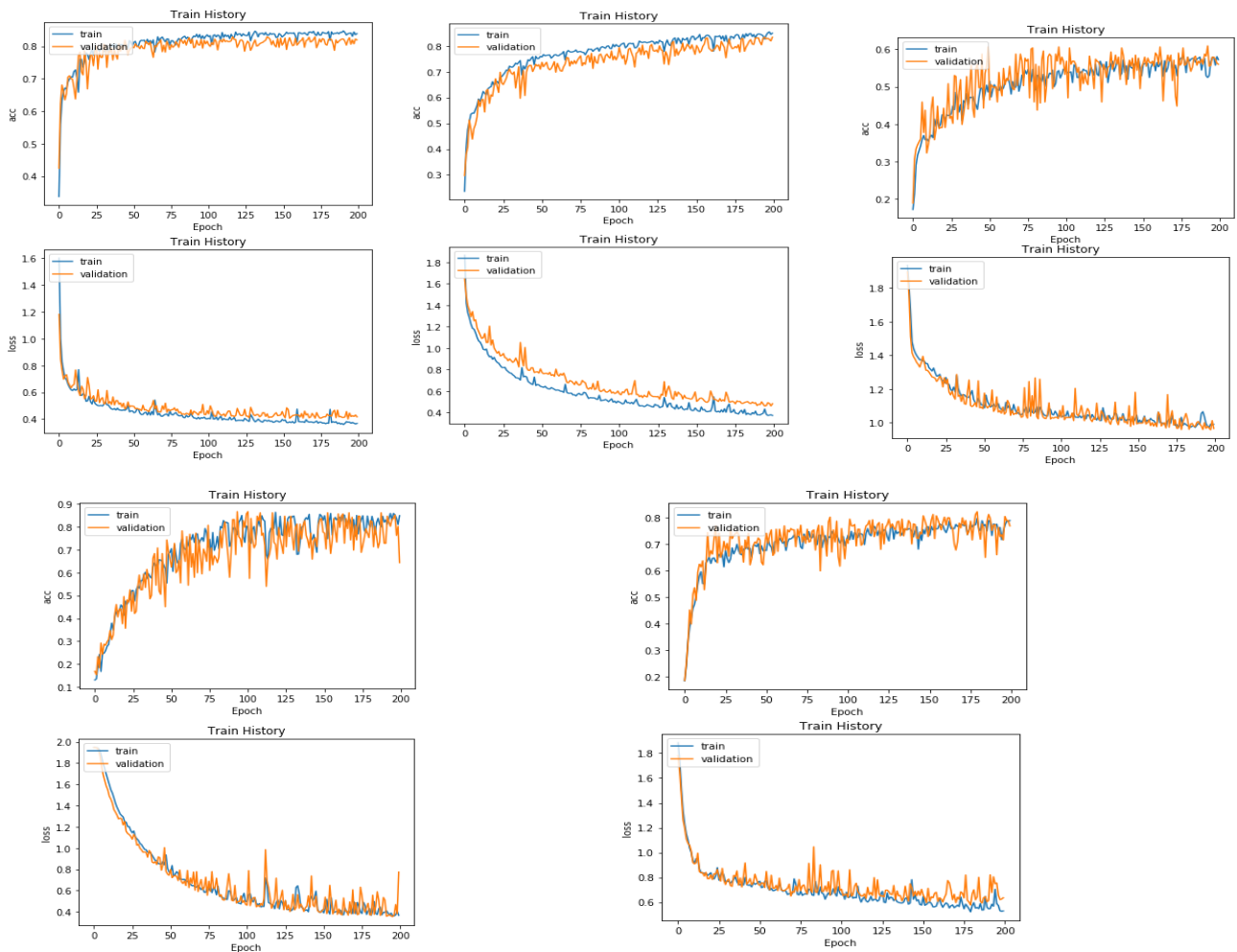


Fig. 35: The best graph for the five websites (accuracy diagram and error diagram)

I can also analyze the five graphs above. In the five graphs, the training curve (blue) and test curve (orange) show similar growth rates. From this observation, I can determine that overfitting did not occur. This shows that the neural network generated by Cyclone Conjecture is very useful.

Four. Conclusion and application

(I) I designed and made a bike recycling system integrating cloud application, AI and Facebook chatbot applications. This system can be used for online reporting and can be linked to the cleaning squadron. Starting from May 2019, this system can be used to report bikes around Taiwan.

(II) I designed the first AI for analyzing and completed the first AI for [judging the abandoned bike]. The AI judges if there is a bike in the photo and the accuracy reaches 95%. The main problem is the standard for judging whether the bike is abandoned. The main problem with the doorplate recognition system is inconsistency among the doorplate format in Taiwan; hence the recognition is affected. The accuracy for now is about 50% and yet to be increased.

(III) By conducting experiments via different parameters and analyzing the output data, I successfully found the rule for the number of hidden layers and neurons in the MLP model. The experiment result shows that if a system with less than 10 parameters (online post), to generate a 1-11 grading scale (post rating), I can consider starting the testing from the parameters below for the MLP neural network:

- Number of hidden layers: 2 to 4
- Number of neurons on each layer: The multiple (1-10) for the average number of neurons on the input and output layers
- Training batch size: 1% of the number of samples
- Epochs for the number of rounds of training would be 500-2000, as long as we can see a stable curve.

This model, Cyclone Conjecture, can be used for using the MLP model to analyze similar parameters. I can refer to this rule for the number of hidden layers and neurons.

(IV) I set three goals for initial selection:

1. I want to develop a chatbot with an English version and provide multiple language versions to enhance usage.
2. I want to improve Google Vision or design an AI exclusive for training to recognize the abandoned bikes on the photo more accurately.
3. I want to verify what we found.

I achieved all the three goals above. Next, we plan to use the neural network design and the math principle of the activation formula of each neuron to find direct evidence for Cyclone Conjecture.

Five. References

- [1] Liberty Times ["Taipei Metropolis" abandoned bicycles as "Roadmaster", posted 500 announcements per month]
- [2] China Times [Invisible Roadhog abandoned bicycles to provoke public resentment] <https://www.chinatimes.com/newspapers/20160806000580-260107>
- [3]Wikipedia 【Artificial intelligence】 https://en.wikipedia.org/wiki/Artificial_intelligence
- [4] Richard Socher and Andrew Ng, “Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank”, [https://nlp.stanford.edu/~socherr/EMNLP2013_RNTN .pdf](https://nlp.stanford.edu/~socherr/EMNLP2013_RNTN.pdf)
- [5]Zhujian Smart Emotibot, "Extremely Validated: A Brief Analysis of Deep Learning Models and Applications", <https://kknews.cc/zh-tw/tech/m25ae6.html>
- [6] Web Article, "On NLP, CNN, RNN (I think that LSTM and other variants are also RNN), the simplest fully connected MLP, what are the advantages and disadvantages of the three?" <https://www.zhihu.com/question/41625896>
- [7] Marc Moreno Lopez , ‘Deep Learning applied to NLP’, <https://arxiv.org/pdf/1703.03091.pdf>
- [8] Jason Brownlee, ”When to Use MLP, CNN, and RNN Neural Networks”, <https://machinelearningmastery.com/when-to-use-mlp-cnn-and-rnn-neural-networks/>
- [9] Jason Brownlee, “Crash Course On Multi-Layer Perceptron Neural Networks“, <https://machinelearningmastery.com/neural-networks-crash-course/>
- [10] Angel Kuri-Morales, ‘Closed determination of the number of neurons in the hidden layer of a multi-layered perceptron network’, Soft Computing, February 2017, Volume 21, Issue 3, pp 597–609
- [11] Fletcher L, Katkovnik V, Steffens FE, Engelbrecht AP (1998) “Optimizing the number of hidden nodes of a feedforward artificial neural network.”, In: Proceedings of the IEEE International Joint Conference on Neural Networks, vol 2, pp 1608–1612

- [12] Rivals I, Personnaz L (2000), "A statistical procedure for determining the optimal number of hidden neurons of a neural model." In: Second International Symposium on Neural Computation (NC'2000), Berlin, May 23–26
- [13] Marie-Christine Suhner 1 Philippe Thomas, "Impact of hidden weights choice on accuracy of MLP with randomly fixed hidden neurons for regression problems", 9th International Joint Conference on Computational Intelligence, IJCCI'17, Nov 2017, Funchal, Portugal
- [14] Jason Brownlee, "How to Configure the Number of Layers and Nodes in a Neural Network", <https://machinelearningmastery.com/how-to-configure-the-number-of-layers-and-nodes-in-a-neural-network/>
- [15] Foram S. Panchal et al, "Review on Methods of Selecting Number of Hidden Nodes in Artificial Neural Network", International Journal of Computer Science and Mobile Computing, Vol.3 Issue.11, November- 2014, pg. 455-464
- [16] comp.ai.neural-nets FAQ, Part 3 of 7: Generalization Section - "How many hidden units should I use?", <http://www.faqs.org/faqs/ai-faq/neural-nets/part3/section-10.html>
- [17] Chester, D.L. (1990), "Why Two Hidden Layers are Better than One," IJCNN-90-WASH-DC, Lawrence Erlbaum, 1990, volume 1, 265-268.
- [18] DeVore, R.A., Howard, R., and Micchelli, C.A. (1989), "Optimal nonlinear approximation," *Manuscripta Mathematica*, 63, 469-478.