IT2023 Lecture Diary
(Oct 6, 2023; Oct 13, 2023; Oct 20, 2023)

1 Assignment 1, Question 1

(a) You are given a set of 9 balls. All of them look the same. Eight of them weight 2000 grams and one of them weights 1999 grams. Human is unable to sense this little difference. Now, the only tool you have is a pan balance. Your job is to find out which of them is lighter. Describe in detail, step by step, how do you use the pan balance to find the lighter ball.

Answer: There are two solutions for solving the above problem.

Solution 1: Initially, label the balls with numbers $B1, B2, \cdots, B9$.

```
Step 1: Weight B1 and B2. GOTO Step 2.
Step 2: IF B1 < B2, B1 is abnormal. STOP! ELSE GOTO Step 3.
Step 3: IF B1 > B2, B2 is abnormal. STOP! ELSE GOTO Step 4.
Step 4: SET N = 3. GOTO Step 5.
Step 5: 5.1: Weight B1 and BN.
        5.2: IF B1 > BN, BN is abnormal. STOP! ELSE GOTO Step 5.3.
        5.3: N = N+1. GOTO Step 5.1.
```

For the above algorithm, the max number of WEIGHT for finding the abnormal ball is 7. It happens if the abnormal ball is B_9.

Solution 2: Initially, arbitrarily partition the balls in three groups, say Group A, B and C. Each group has three balls. Label the balls A_1, A_2 and A_3 for the balls in Group A. Label the balls B_1, B_2 and B_3 for the balls in Group B. Label the balls C_1, C_2 and C_3 for the balls in Group C.

```
Step 1: Weight (A1,A2,A3) and (B1,B2,B3). GOTO Step 2.
Step 2: IF (A1,A2,A3) < (B1,B2,B3), GOTO Step 5.
Step 3: IF (A1,A2,A3) > (B1,B2,B3), GOTO Step 6.
Step 4: IF (A1,A2,A3) = (B1,B2,B3), GOTO Step 7.
Step 5: 5.1: Weight A1 and A2. GOTO Step 5.2.
        5.2: IF A1 < A2, A1 is abnormal. STOP! ELSE GOTO Step 5.3.
        5.3: IF A1 > A2, A2 is abnormal. STOP!
             ELSE A3 is abnormal. STOP!
Step 6: 6.1: Weight B1 and B2. GOTO Step 6.2.
        6.2: IF B1 < B2, B1 is abnormal. STOP! ELSE GOTO Step 6.3.
        6.3: IF B1 > B2, B2 is abnormal. STOP!
             ELSE B3 is abnormal. STOP!
Step 7: 7.1: Weight C1 and C2. GOTO Step 7.2.
        7.2: IF C1 < C2, C1 is abnormal. STOP! ELSE GOTO Step 7.3.
        7.3: IF C1 > C2, C2 is abnormal. STOP!
             ELSE A3 is abnormal. STOP!
```

For the above algorithm, the number of WEIGHT for finding the abnormal ball is 2.

(b) You are given a set of 9 balls. All of them look the same. Eight of them weight 2000 grams and one of them weights 2001 grams. Pan balance is the only tool to be used. Describe in detail, step by step, how do you use the pan balance to find the heavier ball.

Answer: Similar to the solutions for Question 1a.

(c) Again, you are given a set of 9 balls which are looked and sensed the same. Eight of them are 2000 grams. For the abnormal ball, we do not know if it is lighter or heavier. Describe in detail, step by step, how do you use the pan balance to find the abnormal ball.

Answer: There are two solutions for the above problem.

Solution 1: Initially, label the balls with numbers $1, 2, \cdots, 9$.

```
Step 1: Weight B1 and B2. GOTO Step 2.
Step 2: IF (B1 < B2) or (B1 > B2), GOTO Step 3. ELSE GOTO Step 4.
Step 3: 3.1: Weight B1 and B3.
        3.2: IF B1 = B3, B2 is abnormal.
             ELSE, B1 is abnormal. STOP!
Step 4: SET N = 3. GOTO Step 5.
Step 5: 5.1: Weight B1 and BN.
        5.2: IF B1 = BN, GOTO Step 5.3. ELSE, BN is abnormal. STOP!
        5.3: N = N+1. GOTO Step 5.1.
```

For the above algorithm, the max number of WEIGHT for finding the abnormal ball is 7. It happens if the abnormal ball is B_9.

Solution 2: Initially, arbitrarily partition the balls in three groups, say Group A, B and C. Each group has three balls. Label the balls A_1, A_2 and A_3 for the balls in Group A. Label the balls B_1, B_2 and B_3 for the balls in Group B. Label the balls C_1, C_2 and C_3 for the balls in Group C.

```
Step 1: 1.1: Weight (A1,A2,A3) and (B1,B2,B3). GOTO Step 1.2.
        1.2: SET (GA > GB) = TRUE, IF (A1,A2,A3) > (B1,B2,B3).
             GOTO Step 2. ELSE GOTO Step 1.3.
        1.3: SET (GA < GB) = TRUE, IF (A1,A2,A3) < (B1,B2,B3).
             GOTO Step 2. ELSE SET (GA = GB) = TRUE. GOTO Step 2.
Step 2: 2.1: Weight (A1,A2,A3) and (C1,C2,C3). GOTO Step 2.2.
        2.2: SET (GA > GC) = TRUE, IF (A1,A2,A3) > (C1,C2,C3).
             GOTO Step 3. ELSE GOTO Step 2.3.
        2.3: SET (GA < GC) = TRUE, IF (A1,A2,A3) < (C1,C2,C3).
             GOTO Step 3. ELSE SET (GA = GC) = TRUE. GOTO Step 3.

[Note: After Step 2, it comes up with six possible outcomes only.]

Step 3: 3.1: IF (GA < GB) and (GA < GC), GOTO Step 4.
             ELSE GOTO Step 3.2.
        3.2: IF (GA > GB) and (GA = GC), GOTO Step 5.
             ELSE GOTO Step 3.3.
        3.3: IF (GA = GB) and (GA > GC), GOTO Step 6.
             ELSE GOTO Step 3.4.
        3.4: IF (GA > GB) and (GA > GC), GOTO Step 7.
             ELSE GOTO Step 3.5.
```

```
        3.5: IF (GA < GB) and (GA = GB), GOTO Step 8.
             ELSE GOTO Step 3.6.
        3.6: IF (GA = GB) and (GA < GC), GOTO Step 9.

[The abnormal ball is lighter.]
Step 4: 4.1: Weight A1 and A2. GOTO Step 4.2.
        4.2: IF A1 < A2, A1 is abnormal. STOP! ELSE GOTO Step 4.3.
        4.3: IF A1 > A2, A2 is abnormal. STOP!
             ELSE A3 is abnormal. STOP!
Step 5: 5.1: Weight B1 and B2. GOTO Step 5.2.
        5.2: IF B1 < B2, B1 is abnormal. STOP! ELSE GOTO Step 5.3.
        5.3: IF B1 > B2, B2 is abnormal. STOP!
             ELSE B3 is abnormal. STOP!
Step 6: 6.1: Weight C1 and C2. GOTO Step 6.2.
        6.2: IF C1 < C2, C1 is abnormal. STOP! ELSE GOTO Step 6.3.
        6.3: IF C1 > C2, C2 is abnormal. STOP!
             ELSE A3 is abnormal. STOP!

[The abnormal ball is heavier.]
Step 7: 7.1: Weight A1 and A2. GOTO Step 7.2.
        7.2: IF A1 > A2, A1 is abnormal. STOP! ELSE GOTO Step 7.3.
        7.3: IF A1 < A2, A2 is abnormal. STOP!
             ELSE A3 is abnormal. STOP!
Step 8: 8.1: Weight B1 and B2. GOTO Step 8.2.
        8.2: IF B1 > B2, B1 is abnormal. STOP! ELSE GOTO Step 8.3.
        8.3: IF B1 < B2, B2 is abnormal. STOP!
             ELSE B3 is abnormal. STOP!
Step 9: 9.1: Weight C1 and C2. GOTO Step 9.2.
        9.2: IF C1 > C2, C1 is abnormal. STOP! ELSE GOTO Step 9.3.
        9.3: IF C1 < C2, C2 is abnormal. STOP!
             ELSE A3 is abnormal. STOP!
```

For the above algorithm, the number of WEIGHT for finding the abnormal ball is 3.

2 Assignment 1, Question 5

Similar to the previous question, you are given a set of 9 balls which are looked and sensed the same. Now, there are two abnormal balls inside and their weights are unknown. Describe in detail, step by step, how do you use the pan balance to find the abnormal balls.

Answer: Weight B_1 to B_N for $N = 2, \cdots, 9$ and record all the results. Three cases of results will be observed. Let me denote these cases as Case 1.1, Case 1.2 and Case 1.3.

Case 1.1: B_1 is a normal ball. There will be exactly six "=" signs recorded.

B_1 is a normal ball.

$(1,2)$	$(1,3)$	$(1,4)$	$(1,5)$	$(1,6)$	$(1,7)$	$(1,8)$	$(1,9)$	Abnormal balls
$=$	$=$	$\neq$	$=$	$=$	$=$	$=$	$\neq$	B_3, B_9

Once the above result has been observed, the abnormal balls are identified. There is no further work to be done.

Case 1.2: B_1 is lighter than the normal balls. Thus, the other abnormal ball must be the same weight as B_1 or heavier than B_1. Under such circumstance, there will be at least seven "$\neq$" signs recorded.

B_1 is lighter than the normal balls.

$(1,2)$	$(1,3)$	$(1,4)$	$(1,5)$	$(1,6)$	$(1,7)$	$(1,8)$	$(1,9)$	Abnormal balls
$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	B_1
$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$=$	B_1, B_9

For the first situation, only B_1 can be identified as an abnormal ball. A second round of weighting is needed. For the second situation, the abnormal balls are identified. There is no further work to be done.

Case 1.3: B_1 is heavier than the normal balls. Thus, the other abnormal ball must be the same weight as B_1 or lighter than B_1. Under such circumstance, there will be at least seven "$\neq$" signs recorded.

B_1 is lighter than the normal balls.

$(1,2)$	$(1,3)$	$(1,4)$	$(1,5)$	$(1,6)$	$(1,7)$	$(1,8)$	$(1,9)$	Abnormal balls
$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	B_1
$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$=$	B_1, B_9

For the first situation, only B_1 can be identified as an abnormal ball. A second round of weighting is needed. For the second situation, the abnormal balls are identified. There is no further work to be done.

Second Round: As long as B_1 has been identified as an abnormal ball, we could thus weight B_2 to B_N for $N = 3, \cdots, 9$ and record all the results. Again, three cases will be observed. Let me denote these cases as Case 2.1, Case 2.2 and Case 2.3.

Case 2.1: B_2 is a normal ball. There will be exactly five "=" signs recorded.

B_2 is a normal ball.

$(2,3)$	$(2,4)$	$(2,5)$	$(2,6)$	$(2,7)$	$(2,8)$	$(2,9)$	Abnormal balls
$=$	$=$	$=$	$=$	$=$	$=$	$\neq$	B_1, B_9

All abnormal balls are identified. There is no further work to be done.

Case 2.2: B_2 is lighter than the normal balls. There will be exactly seven "$\neq$" signs recorded.

B_2 is lighter than the normal balls.

$(2,3)$	$(2,4)$	$(2,5)$	$(2,6)$	$(2,7)$	$(2,8)$	$(2,9)$	Abnormal balls
$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	B_1, B_2

All abnormal balls are identified. There is no further work to be done.

Case 2.3: B_2 is heavier than the normal balls. There will be exactly at seven "$\neq$" signs recorded.

B_2 is heavier than the normal balls.

$(2,3)$	$(2,4)$	$(2,5)$	$(2,6)$	$(2,7)$	$(2,8)$	$(2,9)$	Abnormal balls
$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	$\neq$	B_1, B_2

All abnormal balls are identified. There is no further work to be done.

While the tricks behind the solution for solving the problem are complicated, the procedure for solving the above problem is simple. In summary, the algorithm for solving this 2-abnormal ball problem is depicted as below.

```
Step 1: 1.1: FOR N = 2 to 9
                 Weight B1 with BN.
                 Record the result.
             END
```

```
        1.2: IF exactly six '=' signs are recorded,
                the balls with unequal signs are abnormal balls.
                STOP.
             ELSE
                B1 is an abnormal ball.
                GOTO Step 2.1.
             END
Step 2: 2.1: FOR N = 3 to 9
              Weight B2 with BN.
              Record the result.
             END
        2.2: IF exactly six '=' signs are recorded,
                the ball with unequal sign is abnormal balls.
                STOP.
             ELSE
                B2 is an abnormal ball.
                STOP.
             END
```

It should be noted that B_1 has already been identified as an abnormal ball if Step 2.1 has to be conducted. Therefore, there is only one abnormal ball in B_2, B_3 to B_9.

Note: The idea behind the above algorithm can be applied in solving a larger scale problem, in which there are M abnormal balls in a group of N balls. Here $N > 2M$. For instance, one problem is to identify all 10 abnormal balls in a group of 21 balls. In such case, there will require at most 10 rounds of weighting to identify all the abnormal balls.

3 Lecture Outline

- Comment on students' presentation of an algorithm and introduce some tricks for algorithm presentation.
  - IF-THEN-ELSE.
  - GOTO.
  - FOR loop.
  - WHILE-DO.
  - DO-WHILE.
  - STOP, ALGORITHM END.

  Note that an algorithm, a procedure and an operation are the same. They are a list of steps for the completion of a job, or for solving a problem.

- **Assumptions on the executor:** The executor is the one executing the instructions. While you are designing an algorithm, you have to make three assumptions. First, you need to assume that you are not the one to execute the algorithm. Someone else will execute. Second, you need to assume that the executor is a stupid person. The executor has no brain. The executor can only follow the instructions listed in the algorithm. Third, the executor has no way to ask you to clarify the steps listed in the algorithm.

- Generally speaking, ChatGPT or Google Bard could be treated as a **hypothesis generator**. With reference to Figure 1b in the lecture note *Introduction to Intelligent Technology (version September 23, 2023)*, ChatGPT and Google Bard are able to build the body of knowledge and generate hypothesis for further investigation.

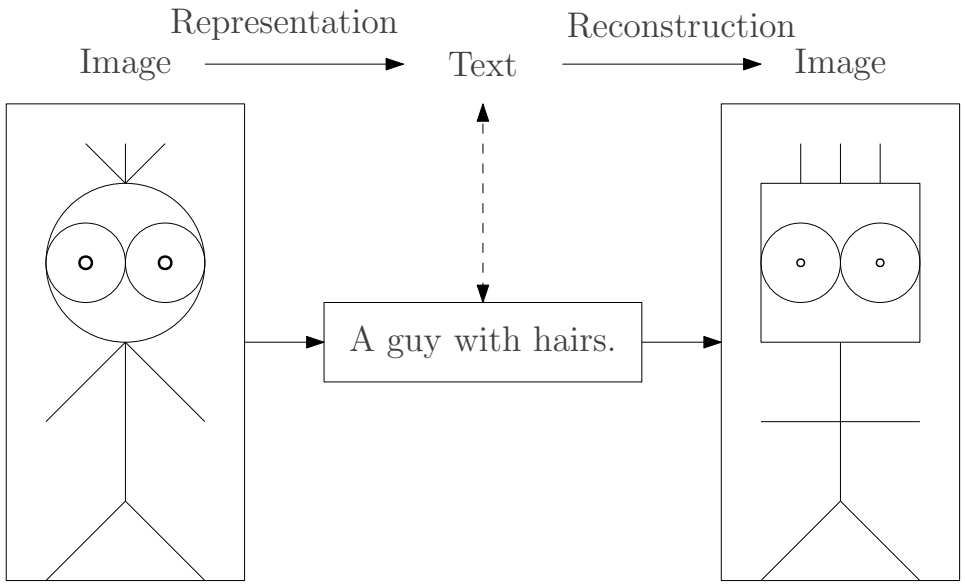


Figure 1: Representation of an image in a sentence and reconstruction of an image from a sentence. Usually, the AI system used for image reconstruction is a *generative AI* system. Reconstruction is a generative process. The AI system used for image representation could be a *generative AI* system or a *non-generative AI* system.

- Those technologies for XYZ-to-TEXT are the technologies for **representation learning**. The text for an XYZ is a representation of XYZ, see Figure 1.
- ChatGPT and Google Bard are two **text generators**, in which their text generation mechanisms are governed by their large language models. A **large language model** (LLM) is a mathematical model with huge number of parameters. To obtain the values of the parameters, each LLM needs to **learn** from a extremely large *text database* which consists of books, articles, Wikipedia pages, technical reports and other public available documents. Once a model parameters have been obtained, the LLM is able to generate a sequence of text of thousands words in response to a question.
  - Clearly, the text generation process of a LLM depends on its mathematical model, i.e. the parameters obtained. These parameters are depended on the text database. If a sentence like '*John Sum is a bad guy.*' appears almost 90% in the text database, the LLM will likely generate the text '*John Sum is evil.*', '*John Sum is bad.*' or '*John Sum is a guy.*' in response to a question to ChatGPT or Google Bard '*How is John Sum?*'. Clearly, '*John Sum is a bad guy.*' could also be generated in response to the question.
  - Thus, ChatGPT and Google Bard could better be described as two *Western-styled* LLMs, as they are trained by the text databases with English documents only. They are not *Asian-styled*. They can hardly generate a sequence of text conforming to Asian culture (resp. Eastern culture). Therefore, each country should develop her own country-oriented LLM.
  - While ChatGPT and Google Bard can support the languages other than English, they rely on translators to generate the non-English text. In simple words, the core text generation is relied on the *Western-styled* LLMs. Take a Chinese language question as an example. The system first translates the Chinese question to an English question. Then, the English question is passed to the LLM to generate a sequence of text in response to the question. Finally, the sequence of English text is passed to the translator to generate the corresponding sequence of Chinese text.

- In Mainland China, a number of LLMs have already been developed and they are developed based upon *Chinese text databases*.
  - Baidu Ernie (`https://yiyan.baidu.com/welcome`).
  - Tencent Hunyuan (`https://hunyuan.tencent.com/`).
  - Beijing Academy of Artificial Intelligence WuDao (`https://www.baai.ac.cn/portal/article/index/cid/49/id/518.html`).
  - Huawei Pangu Model.

  Note that Alibaba LLM Chatbot is powered by OpenAI ChatGPT. Therefore, its core LLM is not developed based on Chinese text database. Taiwan Web Service has released the Formosa Foundation Model (`https://tws.twcc.ai/afs-ffm/`). However, the services provided rely on two LLMs, BLOOM and Mistral-7B Large Language Model (`https://docs.mistral.ai/`), which are developed by Hugging Face (`https://huggingface.co/`) and its collaborators. Below is the report for the BLOOM.

  - Scao, T. L. *et al.* (2022). BLOOM: A 176B-parameter open-access multilingual language model. arXiv preprint arXiv:2211.05100. Available online `https://arxiv.org/abs/2211.05100`.

- In the area of brain-machine-interface (BMI), LLM has been applied to aid the text generation after the brain wave signal has been decoded. Based on recent technology advancement, decoding brain wave of a person to a sequence of text is hardly perfect. For instance, decoding of person thinking of a sentence *You are so beautiful.* might end up with *You a so beauty ful.*. If *You a so beauty ful.* is then passed to an LLM for paraphrasing, it is likely to get *You are so beautiful.* in return. It will clearly be very beneficial to the society. Below is the paper presenting this new application.

  - Tang, J., LeBel, A., Jain, S., & Huth, A. G. (2023). Semantic reconstruction of continuous language from non-invasive brain recordings. *Nature Neuroscience*, Vol.26, 858-866, May 1, 2023. `https://www.nature.com/articles/s41593-023-01304-9`.

- With **voice-to-text** and **text-to-voice** technologies (voice assistants), use of an LLM is a lot more convenience.

- **A generative (AI) model is a model which is trained to generate the outcomes conforming to the** *properties* **(equi. regularities) of the training dataset**. Technically, a generative model is a probabilistic model. That is to say, the outcome generated at the first time is likely different from the outcome generated at the second time. Equivalently, a generative model is in essence a random outcome generator.

- Clearly, given a set of samples with a specific distribution, it could be more than one generative model possibly generating the outcomes conforming the distribution. Here is an example. Suppose a number of observations have been recorded from a fair dice. The problem is to find a generative model which is able to generate the outcomes.

  - Figure 2 shows two generative models which are able to mimic the outcomes of a fair dice. Model 1 consists of two dices and a random switch. In each round, both Dice A and Dice B are rolled. Let their numbers are N_A and N_B. The switch randomly selects a number to output. The probability of Dice A (resp. Dice B) is selected is 1/2. If we repeat rolling the pair of dices for infinite time, the normalized distribution of the outcomes is identical to a fair dice.

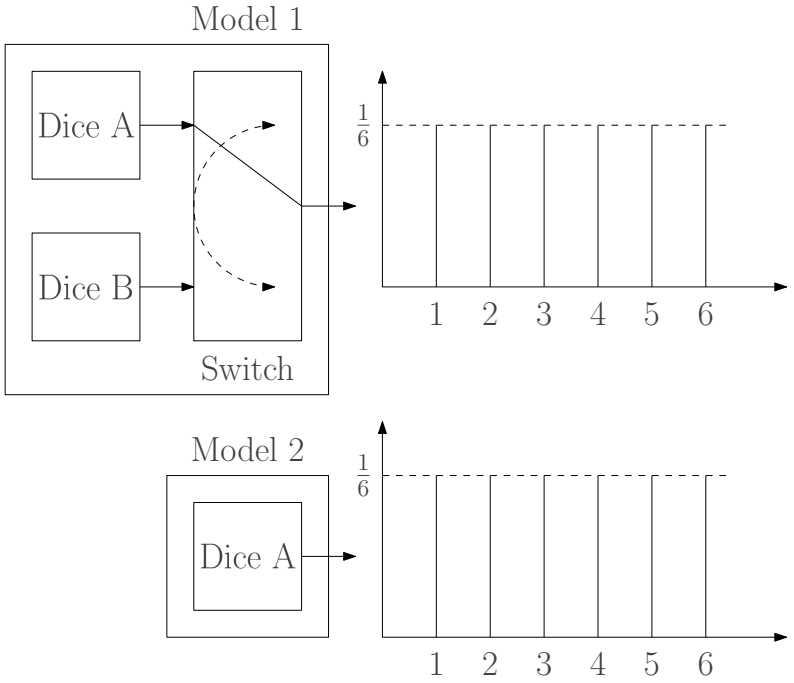


Figure 2: To mimic the properties of a fair dice, at least two generative models can be applied. Let X_A and X_B be the outcomes of *Dice A* and *Dice B* respectively. Moreover, we let Y_1 and Y_2 be the outcomes of the first and the second model. Thus, we can get that $Y_1 = p_A X_A + p_B X_B$ for the first model and $Y_2 = X_A$ for the second model. Let us further denote $P(N|A) = p_{AN}$ (resp. $P(N|B) = p_{BN}$), for $N = 1, 2, \cdots, 6$, as the probability of *Dice A* (resp. *Dice B*) giving outcome N. One should note that $p_A + p_B = 1$. So, $p_B = 1 - p_A$. The first model can be rewritten as $Y_1 = p_A X_A + (1 - p_A) X_B$. The number of parameters in the first generative model is 13, while the number of parameters in the second generative model is 6. In this illustration, $p_{AN} = p_{BN} = 1/6$ and $p_A = p_B = 1/2$. With these parameters, the values $P(N)$ can be obtained analytically. $P(1) = \cdots = P(6) = 1/6$ for both Model 1 and Model 2. In other words, both models are able to generate the outcomes with $P(1) = \cdots = P(6) = 1/6$. The number of parameters in a model is usually accounted for the complexity of a model.

- Model 2 consists of one dice only. If we repeat rolling the dice infinite time, the normalized distribution of the outcomes is identical to a fair dice. Thus, the normalized distributions of the outcomes generated by Model 1 and Model 2 are the same.
- **Model complexity** is always an issue to be concerned. Model 1 is clearly more complex (equi. complicated) than Model 2 as Model 1 has thirteen parameters and Model 2 has six parameters only. In principle, the generative model for generating the outcomes conforming the properties (equi. regularities) of the training dataset should be as simple as possible. *Simple is beautiful*. However, in current AI research, many AI models are very complex.
- A point should be noted. Model 1 will generate the outcomes conforming to the distribution as long as $p_A + p_B = 1$. That is to say, if $p_A = 0.1$ and $p_B = 0.9$, the distribution of the outcomes conforms the distributed observed from the training dataset.

- The challenge is that we have only obtained from the observation (resp. training dataset) $P(1) = \cdots = P(6) = 1/6$. **Which generative model is able to generate the outcomes with such statistical property?** What if $P(1), P(2), P(3), P(4), P(5)$ and $P(6)$ are not $1/6$, let say $P(1) = 1/2$ and $P(2) = \cdots = P(6) = 1/10$, what generative model is able to generate the outcomes with such statistical property? It is the key challenge.

- **A generative model is a random event generator**. Each time an information is input to the model, the model will randomly generate an outcome with the most likelihood. In another time the same information is input, the model will randomly generate an outcome its property is similar to the first outcome. To have a brief introduction on generative AI, one can access the website `generativeai.net`.

- It should be noted that the idea of generative model is not new. In 1983, the Boltzmann machine proposed by Geoffrey Hinton is already a generative model.

  - Fahlman, S. E., Hinton, G. E., & Sejnowski, T. J. (1983, June). Massively parallel architectures for Al: NETL, Thistle, and Boltzmann machines. In National Conference on Artificial Intelligence, AAAI.

  In the subsequence three decades, Geoffrey Hinton had proposed a few more generative models. The key idea behind a generative model is to randomly generate the outcomes fitting the statistical properties of the training dataset as illustrated in Figure 2.

  Geoffrey Hinton did not use the term generative model for his models. Until in 2007, Tu presented a model using the term generative model.

  - Tu, Z. (2007, June). Learning generative models via discriminative approaches. In 2007 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.

  Later in 2014, Ian Goodfellow and his colleagues proposed a model called *Generative Adversarial Net*. Subsequently, Alec Radford, Luke Metz and Soumith Chintala discovered that each image generated by the generative model could be represented by a unique input vector. Accordingly, the well-trained generative model is able to be applied to generate fake images.

  - Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27.
  - Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.

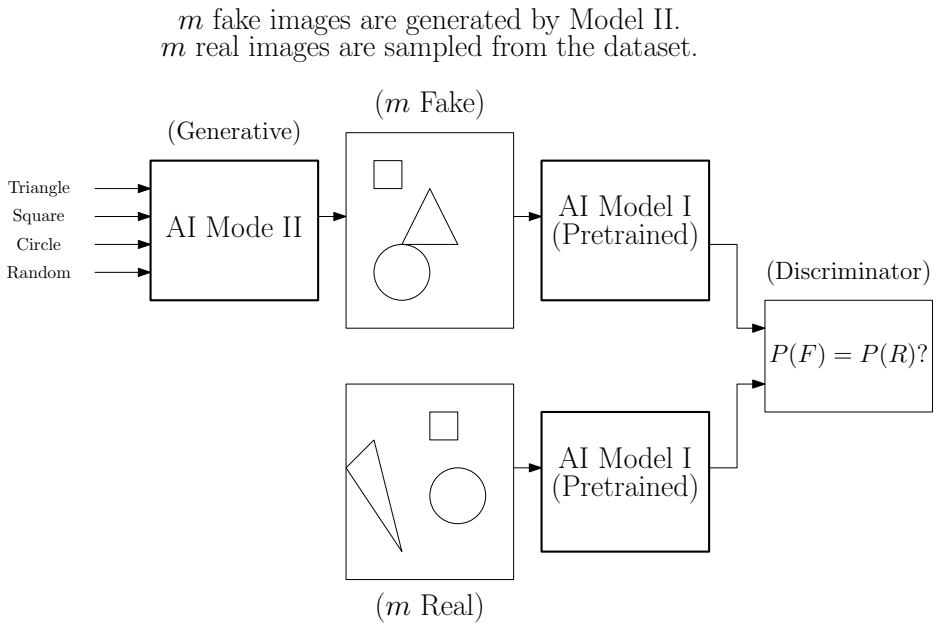


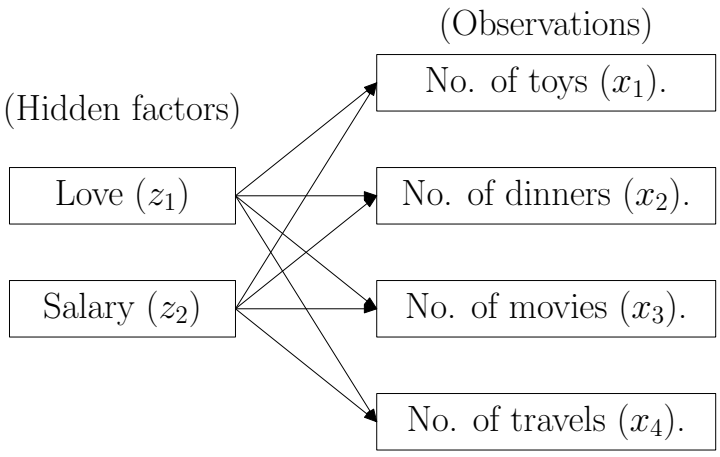
Figure 3: A generative adversarial network and the idea behind its training. If both the discriminator and the generator have been well trained, the generator is able to generate a set of random images its distribution $P(F)$ is almost the same as the distribution of a set of random sampled real images $P(R)$. Moreover, each randomly generated image can be represented numerical by its input vector. After getting the physical meaning of each of the input in the input vector, the generator Model II can thus be applied to generate numerous fake images.

Figure 3 shows a generative adversarial network and the idea behind its training. If both the discriminator and the generator have been well trained, the generator is able to generate a set of random images its distribution $P(F)$ is almost the same as the distribution of a set of random sampled real images $P(R)$. Moreover, each randomly generated image can be represented numerical by its input vector. After getting the physical meaning of each of the input in the input vector, the generator Model II can thus be applied to generate numerous fake images. After that, the term **generative model** has been popularized and led to the area of **Generative AI**.

- A student with undergraduate major in management or social science should be knowledgable about *factor analysis*, which is introduced either in the course *research method* or the course *multivariate statistics*. The model for *factor analysis* is in fact a generative model. If you are familiar with the *factor analysis* model, you could understand the idea about generative AI models.

  - **Example 1:** Consider the factor analysis model as shown in Figure 4. It is assumed that (i) the number of toys x_1 purchased to, the number of dinners together with x_2, (iii) the number of movies watching with and (iv) the number of travels with a daughter in a year depend on how much a father loving a daughter z_1 and his annual salary earned z_2.

$$\begin{aligned} x_1 &= a_{10} + a_{11}z_1 + a_{12}z_2 + \epsilon_1, \\ x_2 &= a_{20} + a_{21}z_1 + a_{22}z_2 + \epsilon_2, \\ x_3 &= a_{30} + a_{31}z_1 + a_{32}z_2 + \epsilon_3, \\ x_4 &= a_{40} + a_{41}z_1 + a_{42}z_2 + \epsilon_4, \end{aligned}$$



*** Factor Analysis (FA) Model ***

$$x_1 = a_{10} + a_{11}z_1 + a_{12}z_2 + \epsilon_1.$$
$$x_2 = a_{20} + a_{21}z_1 + a_{22}z_2 + \epsilon_2.$$
$$x_3 = a_{30} + a_{31}z_1 + a_{32}z_2 + \epsilon_3.$$
$$x_4 = a_{40} + a_{41}z_1 + a_{42}z_2 + \epsilon_4.$$

Figure 4: A factor analysis model assuming that (i) the number of toys x_1 purchased to, the number of dinners together with x_2, (iii) the number of movies watching with and (iv) the number of travels with a daughter in a year depend on how much a father loving a daughter z_1 and the annual salary earned z_2.

where ϵ_1, ϵ_2, ϵ_3 and ϵ_4 are zero mean normal distributed random variables. Furthermore, it is assumed that z_1 and z_2 follows normal distribution as well. So, the total number of parameters in the above model is 18. Apart from a_{ij}, we also need to find out the standard deviations for z_1, z_2, ϵ_1, ϵ_2, ϵ_3 and ϵ_4.

Now, a questionnaire has been designed. The questionnaires are distributed the female students in NCHU and 1000 respondents have been collected. With the data collected, the next problem to be solved is **in search of the parameters for the model** so that the data generated by the FA model conforms to the distribution of the data collected. Luckily, we do have the SPSS and SAS to help use to estimate the values of the parameters. **In search of the model parameters is the so-called learning or training in AI/ML.**

– **Example 2:** Figure 5 shows another example. For the sampled data, Figure 5(a), they are distributed on the line of circle. To find the generative model which is able to generate the data with distribution conforming to the sampled data, it is assumed that the generative model is given by

$$x_1 = \sin(az) \text{ and } x_2 = \cos(bz), \text{ where } z \sim U[0, 10].$$

Figure 5(b) shows the data generated by the model with $(a, b) = (1, 3)$. Figure 5(c) shows the data generated by the model with $(a, b) = (2, 1)$ and Figure 5(d) shows the data generated by the model with $(a, b) = (1, 1)$.

In this example, searching for the model parameters cannot be accomplished neither by SPSS nor SAS. Thus, a learning algorithm has to be developed. A **brute force search method** is tried as many possible cases of (a, b) as possible. That is to say, we try all cases for $a, b = -2, -1.9, \cdots, 1.9, 2$. So, the total number of cases is $41 \times 41 = 1681$. For each case, 1000 data (x_1, x_2) are randomly generated and compared with the distribution of the samples as shown in Figure 5(a). Finally, we can determine from the 1681 distributions for the best (a, b) to be the parameters of the generative model.

Figure 6 shows the schematic diagram for the family of generative model for fitting the sample data as shown in Figure 5(a).

- Following my definition of intelligent technology as presented in the lecture note *Introduction to Intelligent Technology* Section 1.1, the technology developed for generative AI is the Type II intelligent technology. Generative AI technology is developed to accomplish the task used to be solved by human.

- One should be note that the idea of **generative learning** has been advocated in 1970s in the areas of *developmental psychology* and *education psychology*.

  – Wittrock, M. C. (1974). Learning as a generative process. *Educational psychologist*, 11(2), 87-95.

  – Wittrock, M. C. (1974). A generative model of mathematics learning. *Journal for Research in Mathematics Education*, 181-196.

  – Fiorella, L., & Mayer, R. E. (2016). Eight ways to promote generative learning. *Educational Psychology Review*, 28, 717-741.

- In contrast to generative models, another type is non-generative model. Non-Generative model is in essence a deterministic model. An example is shown in Figure 7. To model the relation between the selling price P and the supply quantity Q of a product, a deterministic model $P = a - bQ$ is applied. This model does not consist of any random factor. Once the parameters a and b have been obtained, $a = 44/3$ and $b = 2/3$, the model can be applied to predict the selling price of a product for any supply quantity Q, say $Q = 10.5$. The answer is unique. $P = 23/3$.

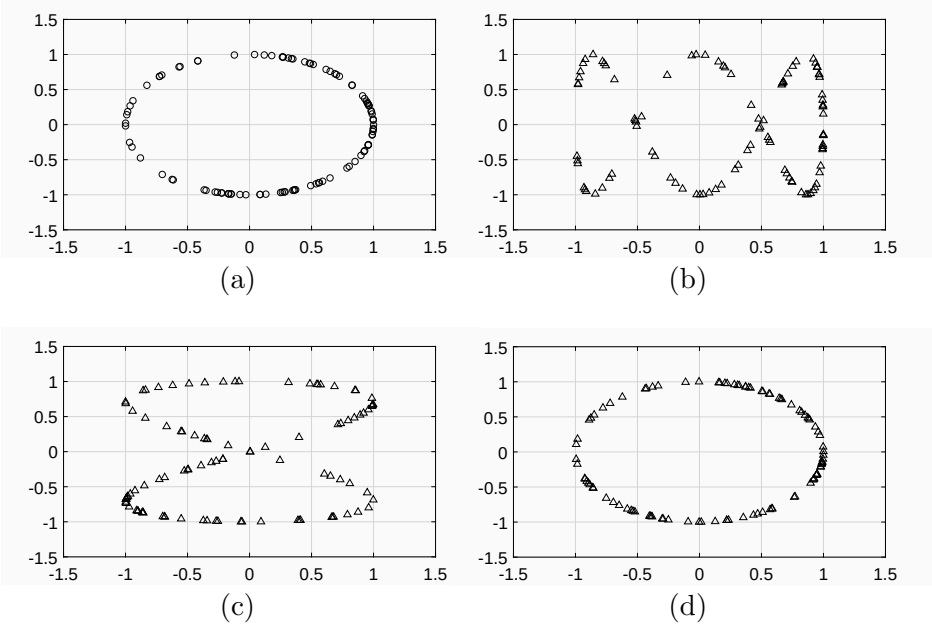


Figure 5: (a) Sampled data. (b) Data generated by the model $x_1 = \sin(z), x_2 = \cos(3z)$. (c) Data generated by the model $x_1 = \sin(2z), x_2 = \cos(z)$. (d) Data generated by the model $x_1 = \sin(z), x_2 = \cos(z)$.

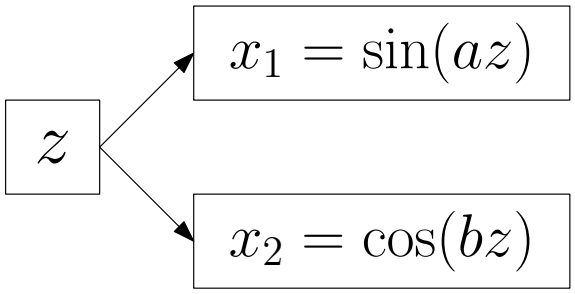


Figure 6: The schematic diagram for the *family of generative models* for fitting the sample data as shown in Figure 5(a).

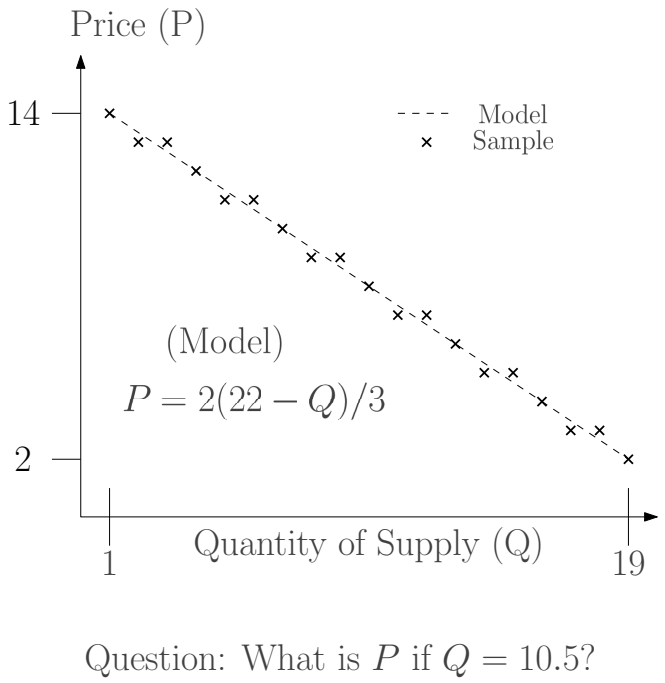


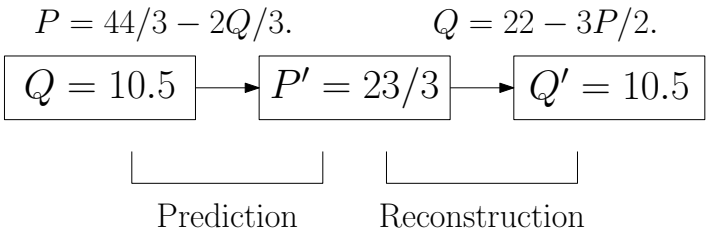
Figure 7: To mimic the relation between the selling price P of a product and the quantity supply Q, the model is a non-generative model given by $P = a - bQ$.

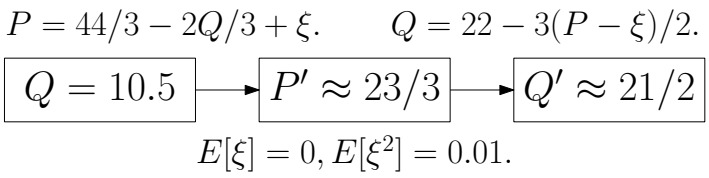
- If the model is deterministic, as shown on the top panel in Figure 8, the prediction P' is unique. Given the prediction P', the reconstruction of the quantity Q' is also unique. If the model applied is with a random factor, as shown in Figure 8 middle panel, the prediction P' is not unique. Thus, the reconstruction of the quantity Q' is not unique. As the noise variance of the random factor ξ is 0.01, one can claim that the noise is small. Thus, the prediction P' should be approximately equal to 23/3 and the quantity Q' is approximately equal to 21/2.

- One major challenge in AI research is to install an AI system in an iPhone or a notebook computer, i.e. the **edge computing** paradigm, see Figure 9. The user does not have to connect to the Internet for the AI services. Today, many AI services are made available on the cloud, such as Google Cloud and Amazon Web Service (AWS). Owing to reduce the loading of a cloud platform and the communication delay due to cloud-device data transfer, the idea of fog computing has been advocated. Certain geographical localized popular softwares are installed in some servers for serving those localized devices.

- One solution is to let an edge device to install the AI systems. However, it is a very difficult problem. One reason is due to the size of an AI model. Like the large language models ChatGPT4, it consists of over one trillion parameters, Table 1. If each parameter is encoded in single precision floating point format, the memory space for storing the model parameters is $7TB$, not to mention about the RAM size required for supporting the use of such model. Therefore, developing an *edge device* supporting such LLM AI service is definitely a difficult problem.

- Starting from 2021, various offline voice assistants and offline LLMs have been developed. Those systems can now be downloaded and installed in a cell phone or a personal computer. The technique behind is to quantize the parameters and hence the computation for text generation is performed by integer arithmetic. The memory space required for the model parameters and the time spent for text generation are largely reduced. Edge AI systems for voice assistant and LLM can now been realized.





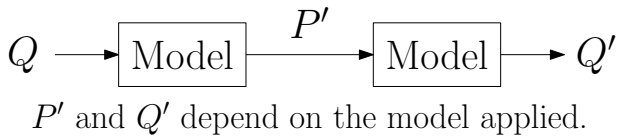


Figure 8: The prediction of the price P' and the reconstruction of the quantity Q' depend on the model applied. Top: The model is deterministic. No uncertainty is incurred in the model. Middle: The model is with a non-deterministic term ξ. Its mean is zero and its variance is 0.01. In a general sense, the model with random variable is also a generative model.

Table 1: Number of parameters in a LLM.

Large Language Model	Number of Parameters	Memory Space[a]
Apple Siri (2019)[b]	133 Kilo[b]	497KB[b]
OpenAI ChaatGPT 1	0.117 Billion	0.468GB[b]
OpenAI ChatGPT 2	1.5 Billion	6GB
OpenAI ChatGPT 3	175 Billion	700GB
OpenAI ChatGPT 4	1760 Billion	$\approx$ 7TB
Goolge Bard	137 Billion	548GB
BLOOM	176 Billion	704GB
Mistral 7B LLM	7 Billion	28GB
Meta LLaMA-7B	7 Billion	28GB
Meta LLaMA-13B	13 Billion	52GB
Meta LLaMA-33B	33 Billion	132GB
Meta LLaMA-65B	65 Billion	260GB
Meta LLaMA2-7B	7 Billion	28GB
Meta LLaMA2-13B	13 Billion	52GB
Meta LLaMA2-70B	70 Billion	280GB
Baidu Ernie	260 Billion	$\approx$ 1TB
Tencent Hunyuan	100 Billion	400GB
BAAI WuDao 2.0	1750 Billion	7TB
Huawei Pangu-α	$\approx$ 200 Billion	$\approx$ 400GB
Huawei Pangu Bot	0.35, 2.6 Billion	1.4GB, 10.4GB
Huawei Pangu-Σ	1085 Billion	$\approx$ 4.3TB

[a] Assume that each parameter is encoded in single precision floating point format. Thus, each parameter requires four bytes memory space for storage.

[b] Zhao, S. *et al.* (2019, July). Raise to Speak: An accurate, low-power detector for activating voice assistants on smartwatches. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2736-2744).

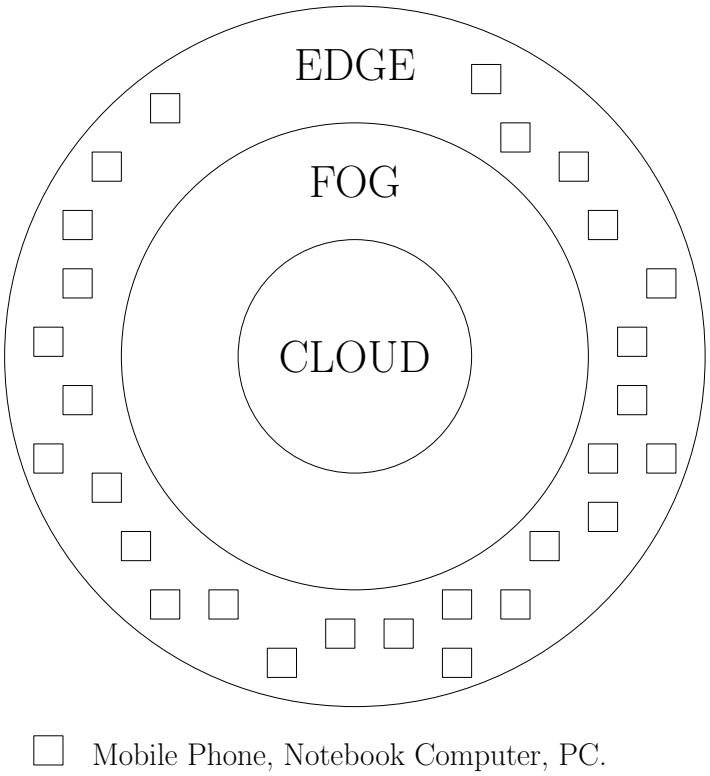


Figure 9: Cloud computing, fog computing and edge computing. The purpose of this three-level architecture is to release the loading of a cloud platform. The ultimate purpose is to let a user have good experience in the use of an AI service. Even if there is no any Internet connection, the devices at the edge are able to have the AI services. Achieving such goal is clearly very difficult for those applications with large language models.