ML2025 Homework 5
Finetuning Is Powerful

TA：上官世昀、陳又華、鄭席鈞

Deadline: 2025/4/18 23:59:59 (UTC+8)

Links

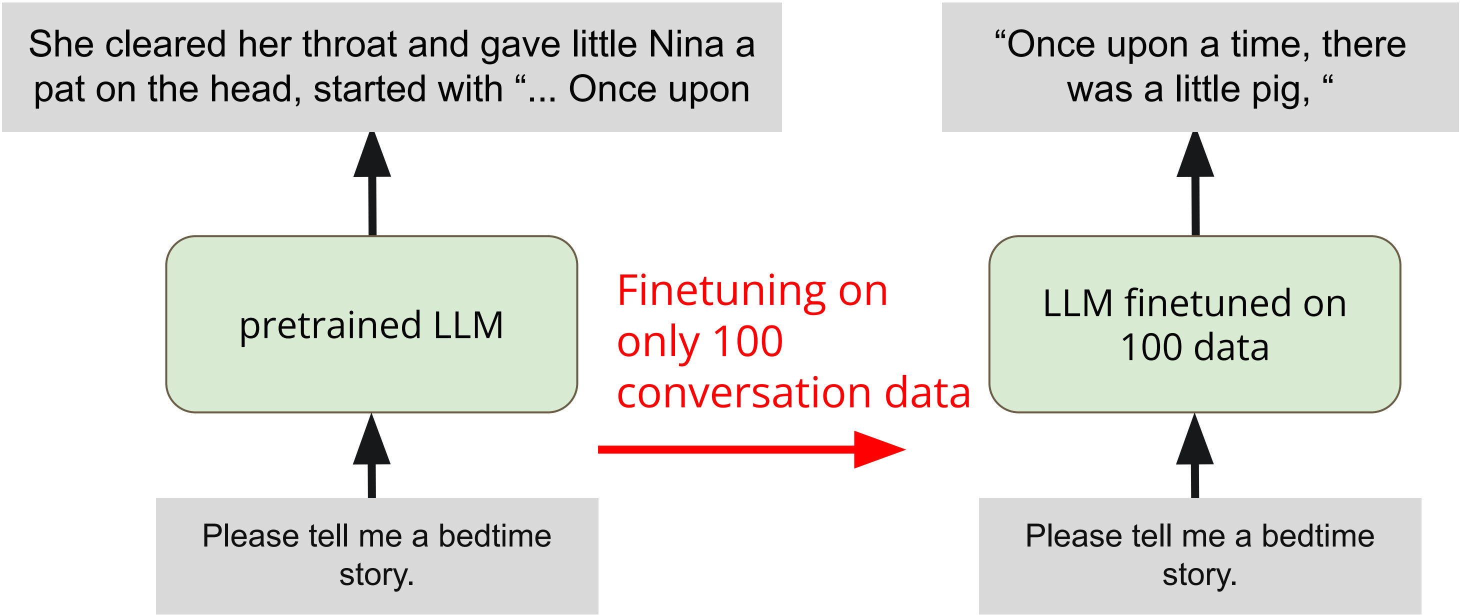
1. Course website : http://speech.ee.ntu.edu.tw/~hylee/ml/2025-spring.php
2. JudgeBoi : https://ml.ee.ntu.edu.tw/home
3. HW5 Google Colab :
   https://colab.research.google.com/drive/1XTLX9o2QveOs71-njzw2ZLNcz7RIDhyX?usp=sharing
4. Long Is More for Alignment: A Simple but Tough-to-Beat Baseline for Instruction Fine-Tuning : https://arxiv.org/pdf/2402.04833

Outline

1. Task Introduction
2. Dataset
3. Eval Metric
4. TODO
5. Submissions and Gradings
6. Regulations
7. Reference

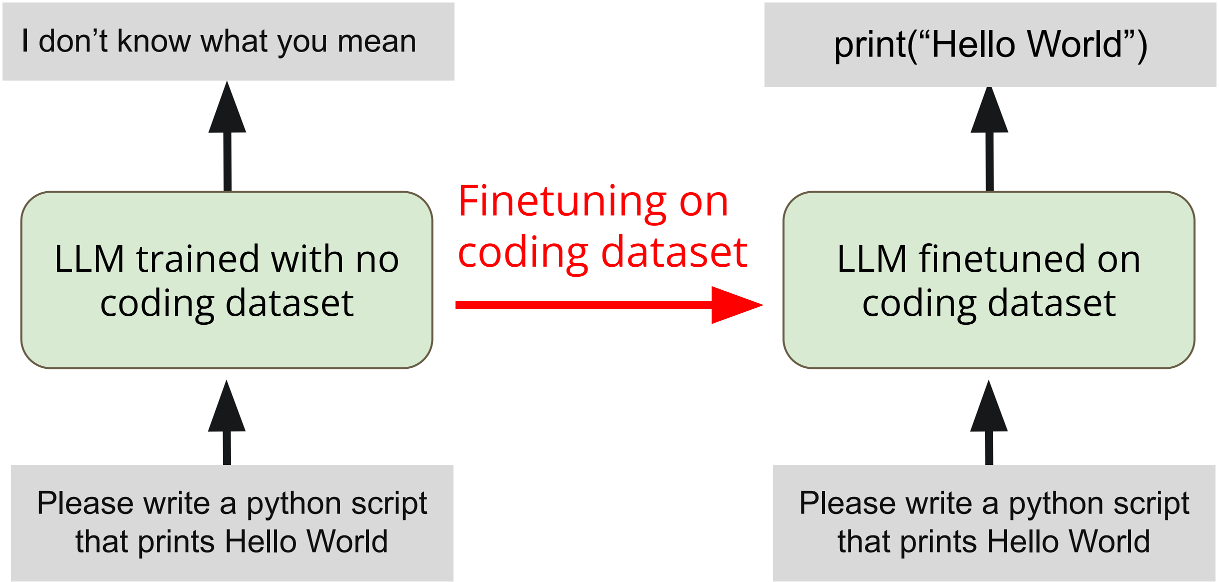
Task Introduction

Task Goal : We can finetune pretrained LLMs to complete tasks it originally performed poorly with only few data and few computational resources.



Task Introduction

Other objectives of finetuning : LLMs can perform better on domains it is not originally good at



Task Introduction

Our Goal : Enable the **LLaMA2-7B** model to effectively understand and execute human instructions by using the Alpaca 52k dataset.

Note : You are limited to use only the **LLaMA2-7B** model.

Dataset

Original Alpaca Dataset : https://huggingface.co/datasets/tatsu-lab/alpaca

To exhibit the potential of finetuning with small amount of data and to reduce the training effort, you are limited to use a subset of only 100 data of the original 52k data.

You should select 100 data among the 52k data provided. Using data from other datasets or data you created yourself is forbidden.

Dataset Format


```
[ {
        "id": "identity_7",
        "conversations": [
            {
                "from": "human",
                "value": "Below is an instruction that describes a task. Write a response that appropriately completes the request. ### Instruction: Write a short story in third person narration about a protagonist who has to make an important career decision."
            },
            {
                "from": "gpt",
                "value": "John was at a crossroads in his life. He had just graduated college and was now facing the big decision of what career to pursue. After much deliberation, he decided that he wanted to be an accountant and help the financially disadvantaged. He had always been good with numbers and enjoyed seeing the tangible results of his work. \n\nJohn enrolled in accounting courses and initially found it quite challenging. He had to learn multiple systems and regulations quickly, but he worked hard and eventually excelled in his studies. After a few years,"
            }
        ],
        "score": 3.5
    },
]
```

Eval Metric

We evaluate the finetuned model on the Evol_Instruct dataset, which is formatted the same as Alpaca dataset but slightly more diverse and complex.

We employ gpt-4o-mini as evaluation assistant to calculate the score of each response by its helpfulness, relevance, accuracy and level of details. The evaluation code is provided for your reference here.

TODO

1. Tune Hyperparameters
2. Filter and Refine dataset
3. Implement Curriculum Learning

TODO : Tune Hyperparameters

Tunable Parameters :

- Training hyperparameters : LoRA rank, LoRA alpha, training epoch, learning rate, ……
- Decoding hyperparameters : temperature,Top-k, Top-p, max_length

ref : https://arxiv.org/abs/2106.09685

TODO : Tweak LoRA Hyperparameters

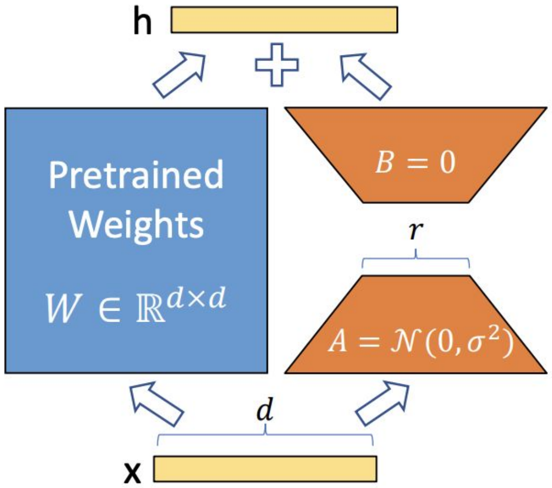


Figure 1: Our reparametrization. We only train A and B.

LoRA Introduction

Why LoRA ?

Modern LLMs have hundreds of billions of parameters => finetuning is costly

LoRA was introduced to tackle this problem by making only a subset of parameters learnable during training.

This kind of technique is thus called Parameter-Efficient Fine-Tuning (PEFT).

LoRA Introduction

LoRA adds two low-rank matrices, denoted as A and B, to the original weight matrix.

Instead of learning a full ΔW update of size d x d, LoRA learns two smaller matrices of size d x r and r x d (where r << d).

r is the "rank" of the matrix here.

W_new = W_original + (α / r) * (A × B)

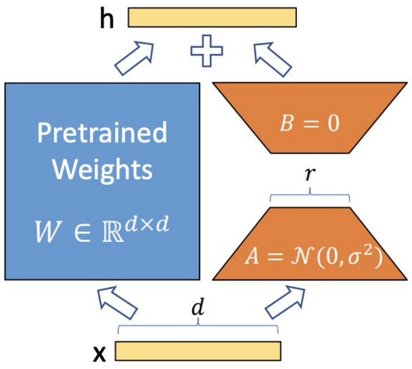


Figure 1: Our reparametrization. We only train A and B.

TODO : Tweak LoRA Hyperparameters – LoRA Rank

Definition : the dimensionality of the low-rank matrices used to approximate the weight updates

Trade-off :

Lower rank: fewer parameters, faster training, risk of underfitting.

Higher rank: more parameters, potentially better performance, risk of overfitting.

TODO : Tweak LoRA Hyperparameters – LoRA Alpha

Definition : how strongly the low-rank adaptation influences the original weight matrix, often applied as: $W\_new = W\_original + (\alpha / r) * (A \times B)$

Smaller alpha: reduces the influence of LoRA updates, preventing large weight shifts.

Larger alpha: increases update magnitude, can improve performance if the model underfits, but may overfit or destabilize training if set too high.

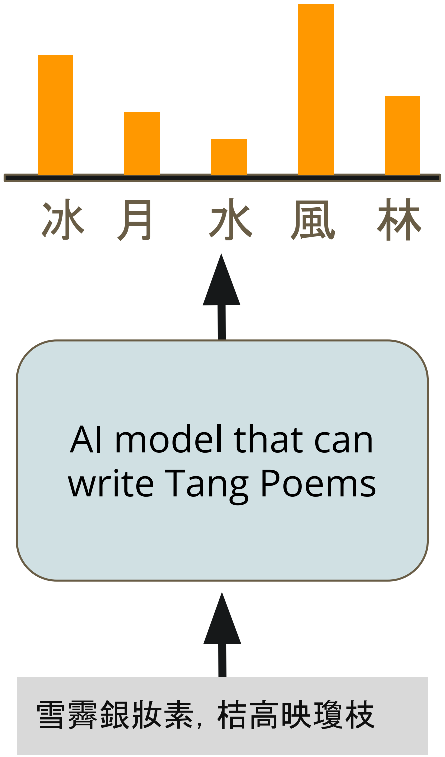
slides by : 陳光銘 林熙哲 余奇恩

TODO : Decoding Hyperparameters

- temperature
- Top-k
- Top-p
- max_length

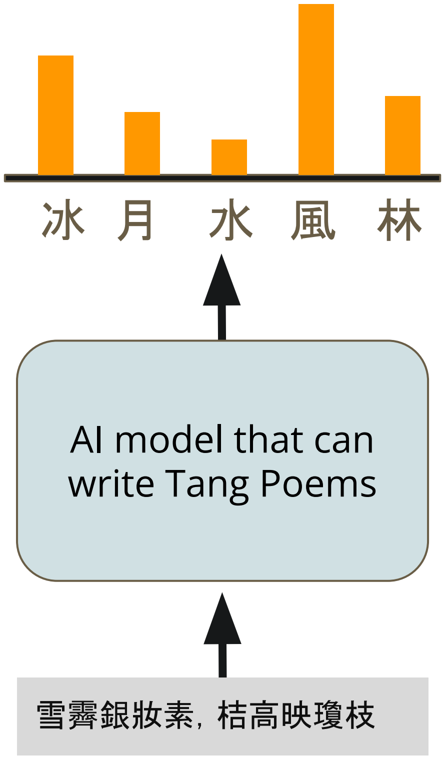
Generation from Language Models is Sampling

- When generating from a language model, we **sample a *token* from the next-token distribution** to determine what the next token is



Generation from Language Models is Sampling

- By changing how we sample from this distribution, we can change how the language model generates the next token



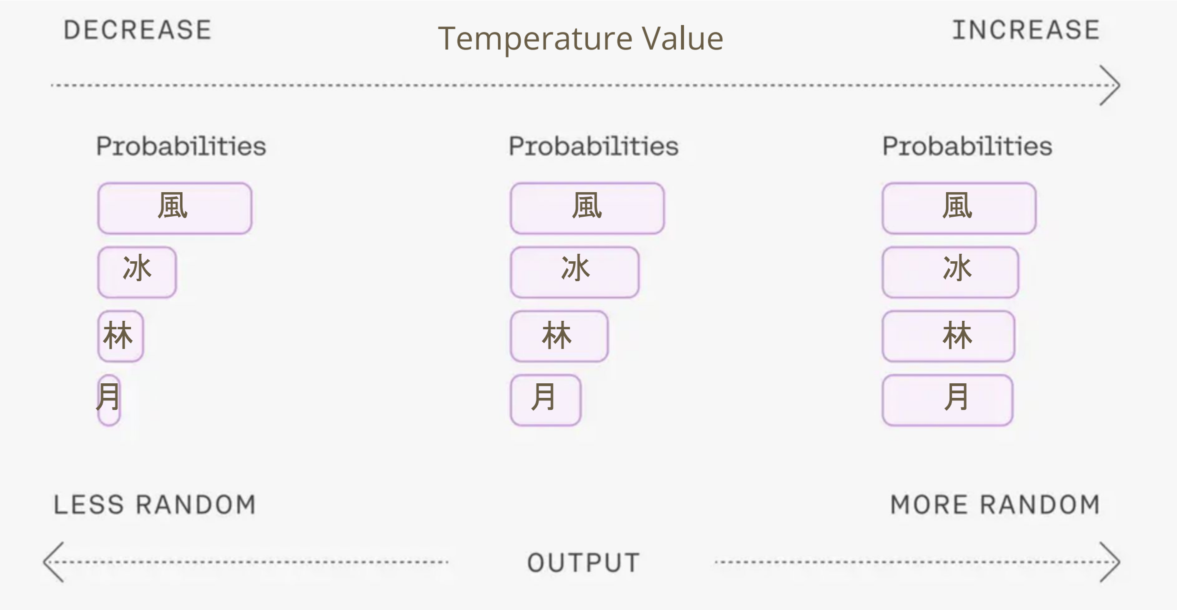
Commonly Used Parameters

- When using LLM for generation, there are some hyperparameters:
  - temperature
  - Top-k
  - Top-p
  - max_length

- We can adjust these hyperparameters to control the behaviors of LLM.
  e.g., longer vs. shorter; diverse vs. static

Temperature

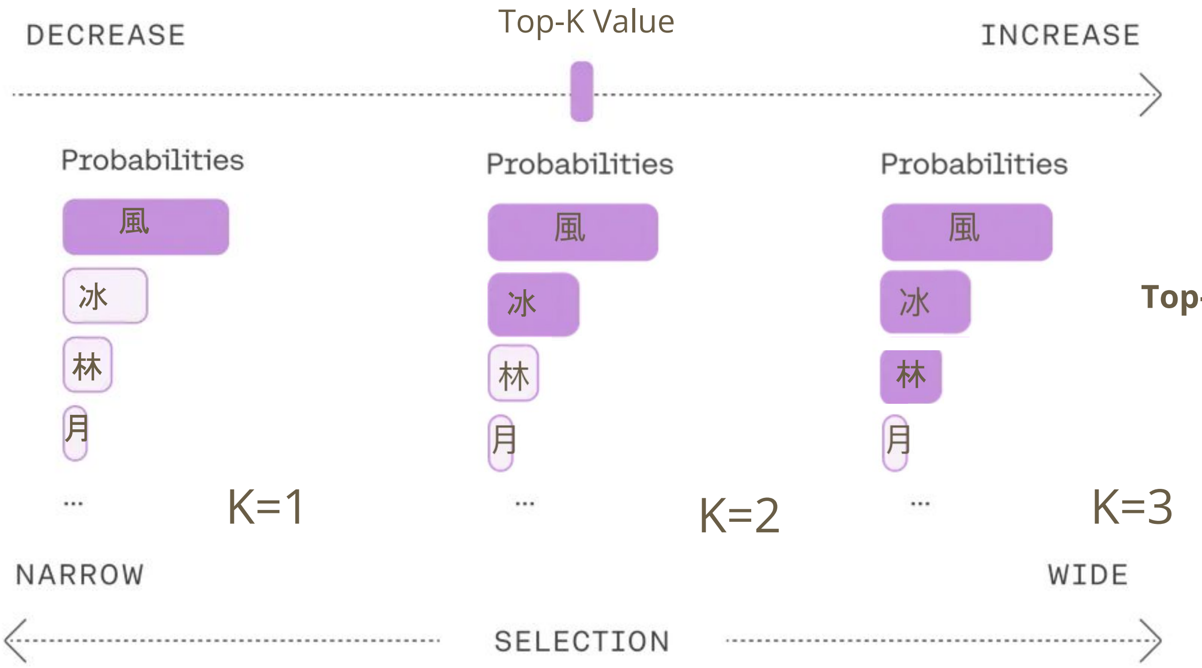
- Related to the diversity of the output, **0.0 ≤ temperature**
- Higher temperature for better diversity



Ref.

Top-K

- Number of selection when LLM makes decision for each word

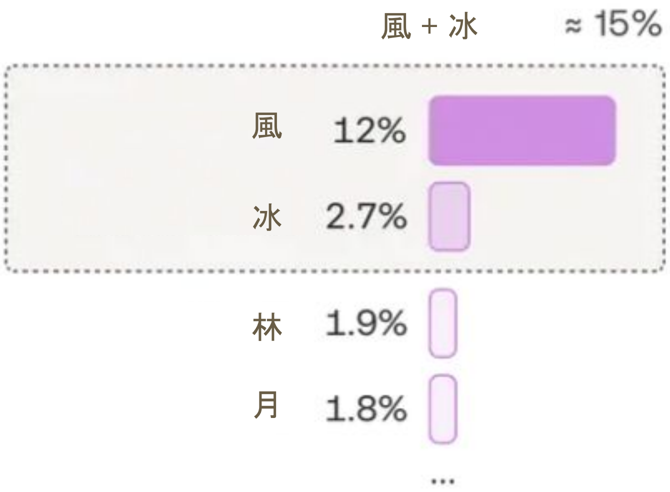


Top-K is a positive integer

Top-P

1. Consider only the top tokens whose likelihoods add up to 15%. Ignore all others.

p = 0.15



2. Sample from them based on their likelihood scores.

0.0 ≤ Top-P ≤ 1.0

Max_length

- The max length of the generated token sequence.
- If you found that the generated sentence seems truncated, check max_length!

指令： 有機電致發光材料及裝置
回覆：
本研究所提供的有機發光二極體及其相關裝置，是一種使用有機材料（通常為硅）進行光電效應的光電元件及其應用。該裝置具有以下特點：

1. 光電導性；
2. 具有高光電容率與低光電阻率；
3. 能夠產生高頻率的光，並且具有較低的光損失率。
4. 可以應用於各種應用領域、如顯示器、顯示屏、通訊裝置、照明裝置等。
5. 對環境的影響較小。
6. 適用於低光環境下的應用﹌如夜間環境、低光照明環境等。

研究結果顯示：使用這種裝置可以有效地提高光學顯示技術的效能（如電視、電

TODO : decoding parameters

You are encouraged to tune decoding parameters to generate higher quality responses.

```
# Generate model outputs
outputs = model.generate(
    input_ids=inputs,
    do_sample=True,
    max_new_tokens=100,
    use_cache=True,
    temperature=1.5,
    #min_p=0.1,
    top_p = 0.9,
    top_k = 30,
)
```

TODO : Filter and Refine Dataset

One of the key factors influencing the success of instruction fine-tuning is the **quality** of the training data.

Recommended Reading

Long Is More for Alignment: A Simple but Tough-to-Beat Baseline for Instruction Fine-Tuning : This paper explores various criteria for curating a high-quality instruction fine-tuning dataset. It might give you some insights on how to refine the dataset.

TODO : Filter and Refine Dataset

Your Task :

Sort the Dataset :

Implement a sorting function (such as the one provided in the sample code) to rank the dataset based on conversation length or any other statistics. Here, conversation length is measured by the total number of words across all messages (both question and answer) in an example.

TODO : Filter and Refine Dataset

Your Task :

Select a Subset for Fine-Tuning :

Experiment with slicing out the top N examples as your training subset. You may also try more advanced methods by combining the conversation length with other metrics (such as the provided score) using a weighted approach.

Hints : Filter and Refine Dataset

Plotting the conversation length and score of the 52000 data points in the dataset might help you better filtering and choosing the subset

TODO : Implement Curriculum Learning

Curriculum learning is a training strategy in which rather than training on all data at once, we structure the training process in “stages” or “levels” of increasing complexity.

This helps the model build a solid foundation on simpler tasks before tackling more challenging ones.

TODO : Implement Curriculum Learning

Your Task :

Create Data Subsets

Based on your chosen metric, split the dataset into multiple “levels” (e.g., Easy/Hard or Easy/Medium/Hard).

Staged Training

First, fine-tune your model on the easiest subset.

Next, progressively incorporate more challenging data in additional training rounds.

Submissions and Gradings

Code Submission (+4 pts)	Submit your code to NTU COOL	Estimated Training Time (in all attempts) (on Colab T4)	Inference Time (on Colab T4)
Public Simple Baseline (+1 pt)	Just run the provided sample code and submit the”pred.json” to Judgeboi	10 minutes	15 minutes
Private Simple Baseline (+1 pt)			
Public Medium Baseline (+1 pt)	Improve the training or refine the data subset by: ● Tuning Hyperparameters ● Refine Dataset ● Curriculum Learning	1~2 hours	15 minutes
Private Medium Baseline (+1 pt)			
Public Strong Baseline (+1 pt)		3~4 hours	15 minutes
Private Strong Baseline (+1 pt)			

Submissions and Gradings – Baselines

Baselines	GPT-4o-mini score
Simple Public	2.4
Medium Public	4.15
Strong Public	4.4

Submissions and Gradings – Judgeboi

- Please submit the **pred.json** to judgeboi. (only .json file is allowed)
- **5** submission quota per day, reset at **23:59** (UTC+8).
- Each submission takes about a minimum of 2-3 minutes to evaluate.

Submissions and Gradings – Code Submission

- Submit code to NTU COOL
- Deadline: 4/18 (Fri.) 23:59 (UTC +8)
- Compress your code into <student ID>_hw5.zip (e.g. b12901001_hw5.zip)
- We can only see your last submission.
- Do not submit model weights or dataset.
- If your code is not reasonable or reproducible, you will receive 0 points for this homework.

Regulations

- Do NOT share codes or prediction files with any living creatures.
- Do NOT use any approaches to submit your results more than 5 times a day.
- The training data is limited to **no more than 100 data points**.
- Do NOT search for or use additional data for training or the answers for the testing data.
- All data filtering and refining code should also be included in the code submission.
- Do not use the test set in any way to modify the training data.
- Do not change the seed for reproducibility.
- You should NOT modify your input file or prediction files manually.
- Make sure that TAs can reproduce the predictions using the code you submit.
- You will receive 0 points for this homework if you violate any of the above rules.
- Prof. Lee & TAs preserve the rights to change the rules & grades.

If any questions, you can ask us via...

- NTU COOL (recommended)

- Email
  ntu-ml-2025-spring-ta@googlegroups.com
  The title should begin with “[hw5]”

- TA hours
  Each Friday During Class
  Time : 13:30 - 14:10 ; 17:30 - 18:00