

---

# Machine Learning HW5

ML TAs

[mlta-2023-spring@googlegroups.com](mailto:mlta-2023-spring@googlegroups.com)

---

# Outline

1. Machine translation
2. Workflow
3. Training tips
4. Requiements
5. Gradescope
6. JudgeBoi Guide
7. Regulation and Grading policy

# Links

[JudgeBoi](#)

[Gradescope](#)

[Cool discussion](#)

[Colab sample code](#)

[Kaggle sample code](#)

# Machine Translation

# Machine Translation

In this homework, we'll translate English to Traditional Chinese

e.g.

- Thank you so much, Chris. -> 非常謝謝你, 克里斯。

Since sentences are with different length in different languages, the seq2seq framework is applied to this task.

# Training datasets

- Paired data
  - TED2020: TED talks with transcripts translated by a global community of volunteers to more than 100 language
  - We will use (en, zh-tw) aligned pairs
- Monolingual data
  - More TED talks in traditional Chinese

# Evaluation

source: Cats are so cute

target: 貓咪真可愛

output: 貓好可愛

## BLEU

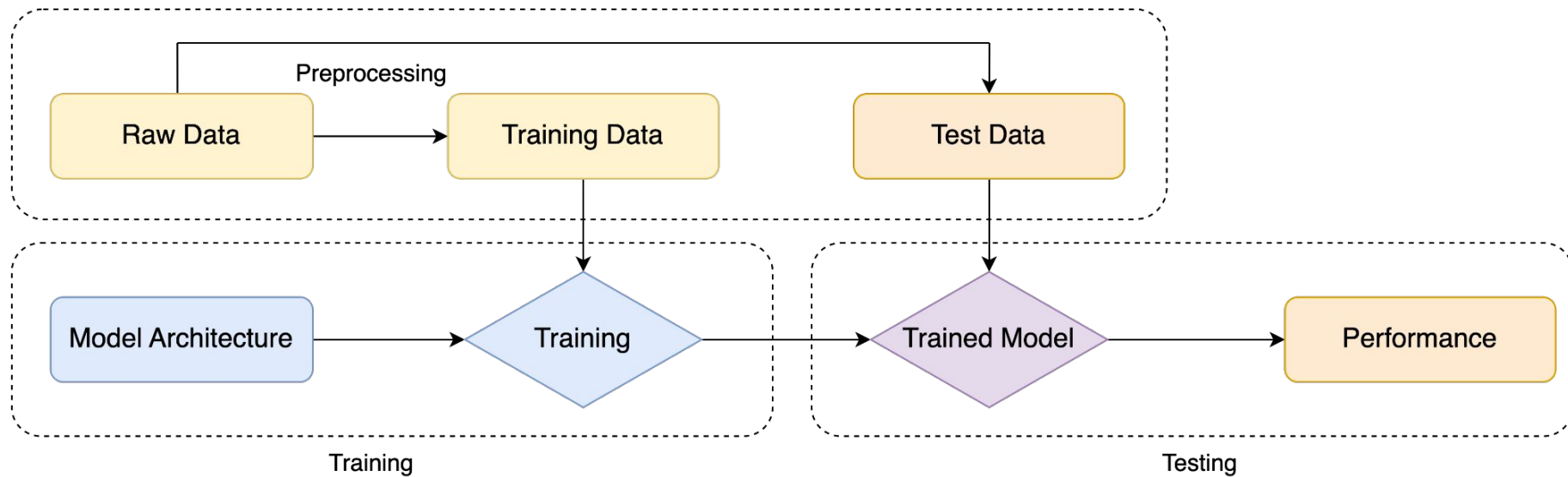
- Modified n-gram precision (n = 1~4)

$$\text{BLEU} = \text{BP} \cdot \exp \left( \sum_{n=1}^N w_n \log p_n \right)$$

- BP: brevity penalty,  $w_n$ : weights,  $p_n$ : n-gram precision
- In brevity, BLEU measures the similarity between the output and target
- Due to the high variance, training more epochs may stabilize your results

# Workflow

# Workflow





# Workflow

## 1. Preprocessing

- a. download raw data
- b. clean and normalize
- c. remove bad data (too long/short)
- d. tokenization

## 2. Training

- a. initialize a model
- b. train it with training data

## 3. Testing

- a. generate translation of test data
- b. evaluate the performance

# Training tips

# Training tips

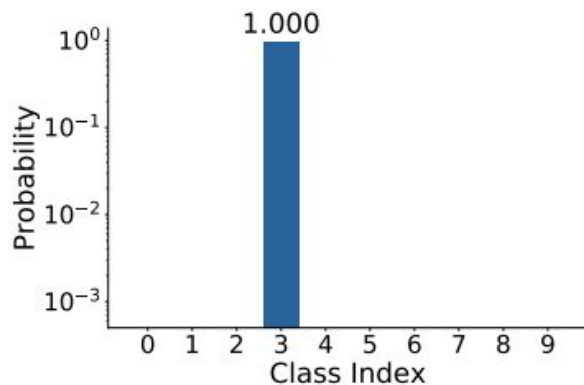
- Tokenize data with sub-word units
- Label smoothing regularization
- Learning rate scheduling
- Back-translation

# Tokenize

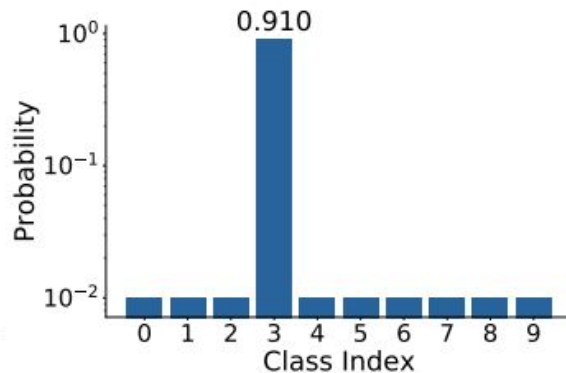
- Tokenize data with subword units
  - Reduce the vocabulary size
  - Alleviate the open vocabulary problem
  - Example
    - \_put \_your s el ve s \_in \_my \_po s ition \_.
    - Put yourselves in my position.

# Label smoothing

- Label smoothing regularization
  - When calculating loss, reserve some probability for incorrect labels
  - Avoids overfitting



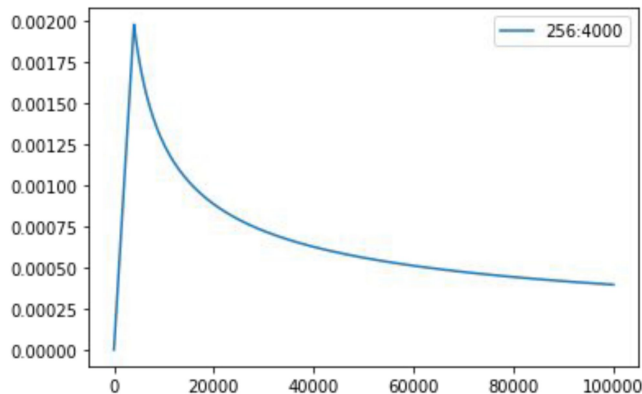
(a) Hard Label



(b) LS

# Learning rate scheduling

- Learning rate scheduling
  - Increasing the learning rate linearly for the first warmup\_steps training steps, and decreasing it thereafter proportionally to the inverse square root of the step number.
  - Stabilizing training for transformers in early stages

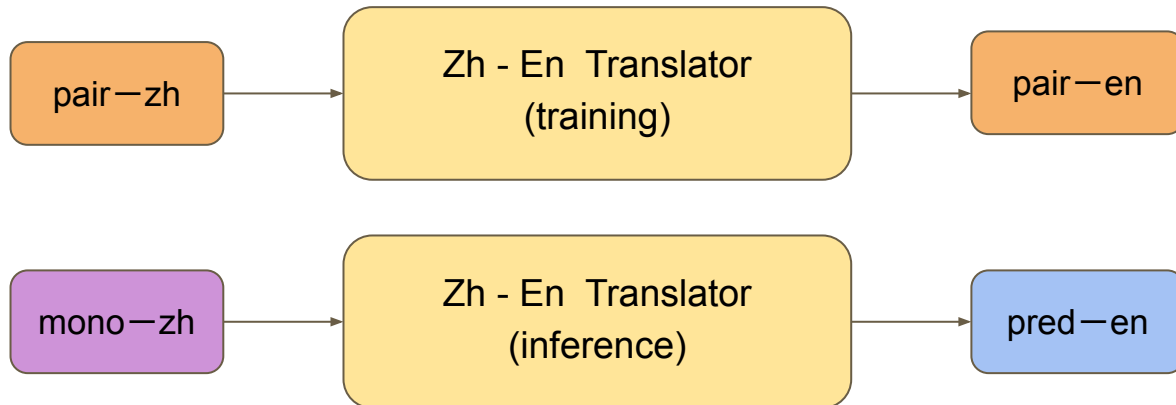


# Back translation

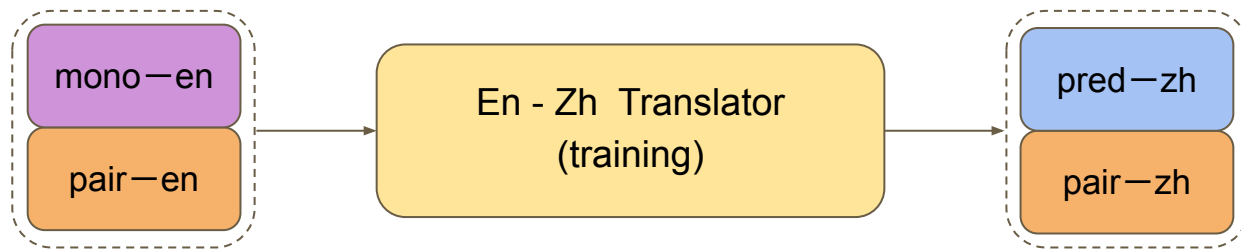
pair—zh

pair—en

mono—zh



**Backward Translation**



**Forward Translation**

# Back translation

Using monolingual data for creating synthetic translation data

1. Train a translation system in the **opposite direction**
2. Collect monolingual data in target side and apply machine translation
3. Use the translated and original monolingual datasets as additional parallel data to train stronger translation systems



# Back translation

Some points to note about back-translation

1. Monolingual data should be in the same domain as the parallel corpus
2. The performance of the backward model is critical

# Requirements

# Baselines

| Baseline | Public score | Estimated time (kaggle) |
|----------|--------------|-------------------------|
| Simple   | 15.05        | 1 hour                  |
| Medium   | 18.44        | 2 hours                 |
| Strong   | 23.57        | 3 hours                 |
| Boss     | 30.08        | > 12 hours              |

# Baseline Guide

- Simple Baseline: Train a simple RNN seq2seq to achieve translation
- Medium Baseline: Add learning rate scheduler and train longer
- Strong Baseline: Switch to Transformer and tuning hyperparameter
- Boss Baseline: Apply back-translation

# Simple Baseline

Train a simple RNN seq2seq to achieve translation

- Running the sample code should pass the baseline

# Medium Baseline

Add learning rate scheduler and train longer

$$lr = d_{\text{model}}^{-0.5} \cdot \min(step\_num^{-0.5}, step\_num \cdot warmup\_steps^{-1.5})$$

```
def get_rate(d_model, step_num, warmup_step):  
    # TODO: Change lr from constant to the  
    # equation shown above  
    lr = 0.001  
    return lr
```

```
config = Namespace(  
    .  
    .  
    .  
    # maximum epochs for training  
    max_epoch=15, # medium: → 30  
    start_epoch=1,  
    .  
    .  
    .  
)
```

# Strong Baseline

Switch to Transformer and tuning hyperparameter

```
encoder = RNNEncoder(args, src_dict, encoder_embed_tokens)
decoder = RNNDecoder(args, tgt_dict, decoder_embed_tokens)
→ # encoder = TransformerEncoder(args, src_dict, encoder_embed_tokens)
    # decoder = TransformerDecoder(args, tgt_dict, decoder_embed_tokens)
```

```
arch_args = Namespace(
    encoder_embed_dim=256,
    encoder_ffn_embed_dim=512,
    encoder_layers=1, # recommend to increase → 4
    decoder_embed_dim=256,
    decoder_ffn_embed_dim=1024,
    decoder_layers=1, # recommend to increase → 4
    share_decoder_input_output_embed=True,
    dropout=0.3,
)
```

for other hyperparameters for  
transformer-base, please refer to  
Table 3 in [Attention is all you need](#)

# Boss Baseline

## Apply back-translation

1. Train a backward model by switching languages

```
source_lang = "zh",  
target_lang = "en",
```

2. Translate monolingual data with backward model to obtain synthetic data
  - a. Complete TODOs in the sample code
  - b. All the TODOs can be completed by using commands from earlier cells
3. Train a stronger forward model with the new data
  - a. If done correctly, ~30 epochs on new data should pass the baseline



# Gradescope

# Gradescope Overview

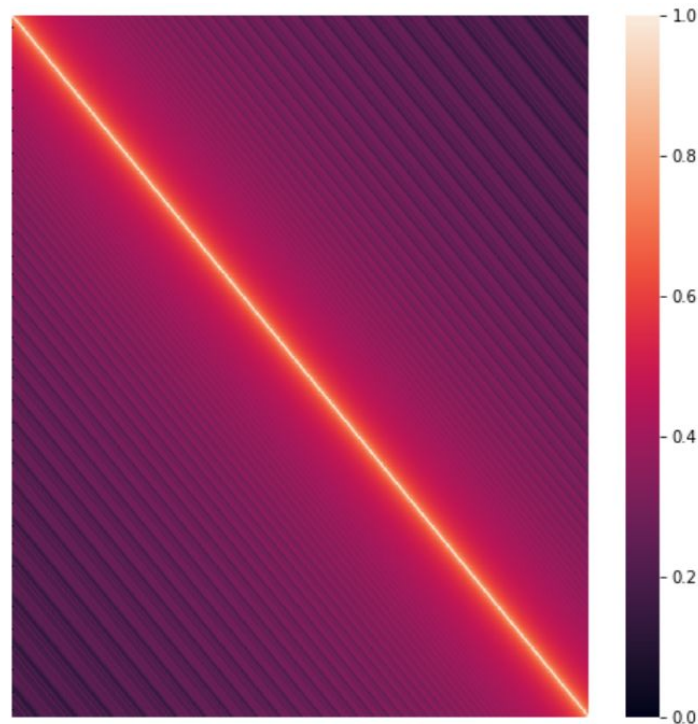
- (2pts) Problem 1
  - Visualize the similarity between different pairs of positional embedding and briefly explain the result.
  - Additionally, attach the code that you used for visualization.
- (2pts) Problem 2
  - Clip gradient norm and visualize the changes of gradient norm in different steps. Circle two places with gradient explosion.

# Problem 1: Visualize Positional Embedding

Given a  $(N \times D)$  positional embedding lookup table, you aim to get a  $(N \times N)$  “similarity matrix” by calculating similarity between different pairs of embeddings in the table.

You need to **visualize the similarity matrix and briefly explain the result.**

In this problem, we focus on the positional embeddings of the **decoder**



# Problem 1: Similarity Matrix

|    | p1  | p2  | p3  | p4  | p5  |
|----|-----|-----|-----|-----|-----|
| p1 | 1   | 0.8 | 0.6 | 0.4 | 0.3 |
| p2 | 0.8 | 1   | 0.8 | 0.6 | 0.4 |
| p3 | 0.6 | 0.8 | 1   | 0.8 | 0.6 |
| p4 | 0.4 | 0.6 | 0.8 | 1   | 0.8 |
| p5 | 0.3 | 0.4 | 0.6 | 0.8 | 1   |

In the sense of encoding positional information, **we expect that the similarity between the embedding of close positions is stronger.**

# Problem 1: Cosine Similarity

We recommend you to measure the similarity between two vectors by cosine similarity.

There is a pytorch implementation of cosine similarity. Check more detail in the following link.

[https://pytorch.org/docs/stable/generated/torch.nn.functional.cosine\\_similarity.html](https://pytorch.org/docs/stable/generated/torch.nn.functional.cosine_similarity.html)

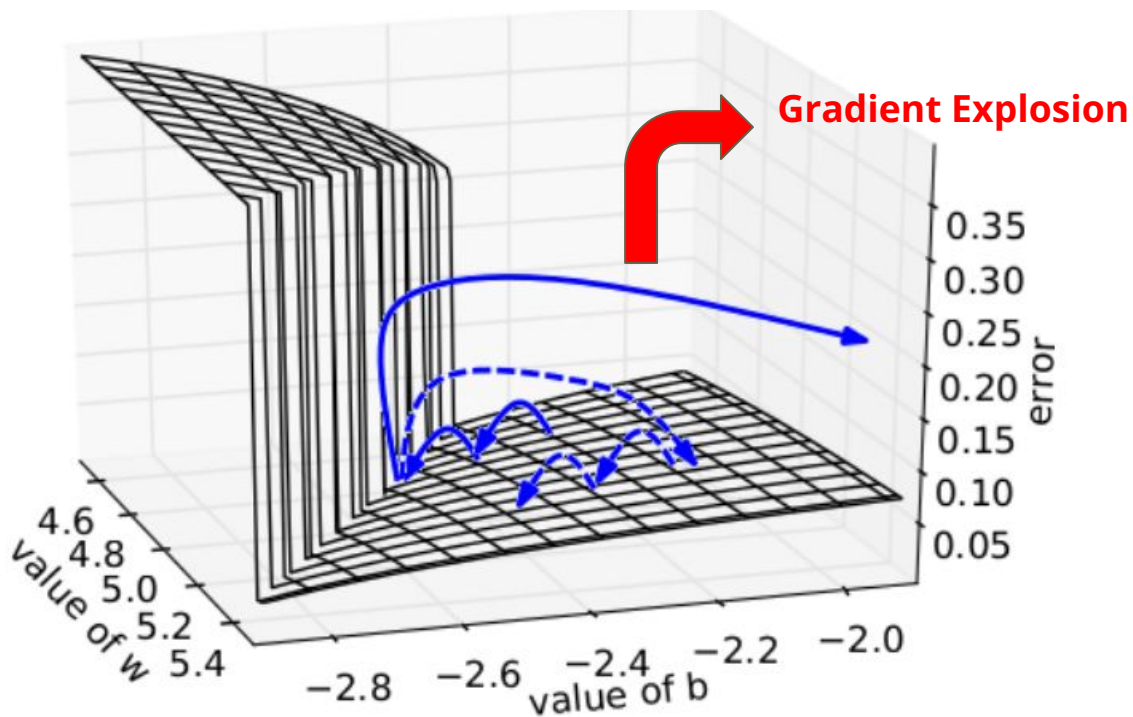
$$\text{similarity} = \frac{x_1 \cdot x_2}{\max(\|x_1\|_2 \cdot \|x_2\|_2, \epsilon)}$$

## Problem 1: Tips and Hint

You could get the positional embeddings of decoder by following codes

```
pos_emb = model.decoder.embed_positions.weights.cpu().detach()
```

## Problem 2: Gradient Explosion



[宏毅老師講解: ML2017 - RNN](#)

[ICML 2013, Razvan Pascanu](#)

## Problem 2: Clipping Gradient Norm

1. Set up a maximum norm value *max\_norm*
2. Collecting the gradient of each parameters to be a vector. Calculate the **p-norm of the vector** to be *Lnorm*
3. If *Lnorm* <= *max\_norm*, do nothing. Otherwise calculate the scale factor *scale\_factor* = *max\_norm* / *Lnorm* and multiply each gradient by the scale factor.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$\|\mathbf{x}\|_p := \left( \sum_{i=1}^n |x_i|^p \right)^{1/p}.$$

p-norm

$$\|\mathbf{x}\|_2 := \sqrt{x_1^2 + \cdots + x_n^2}.$$

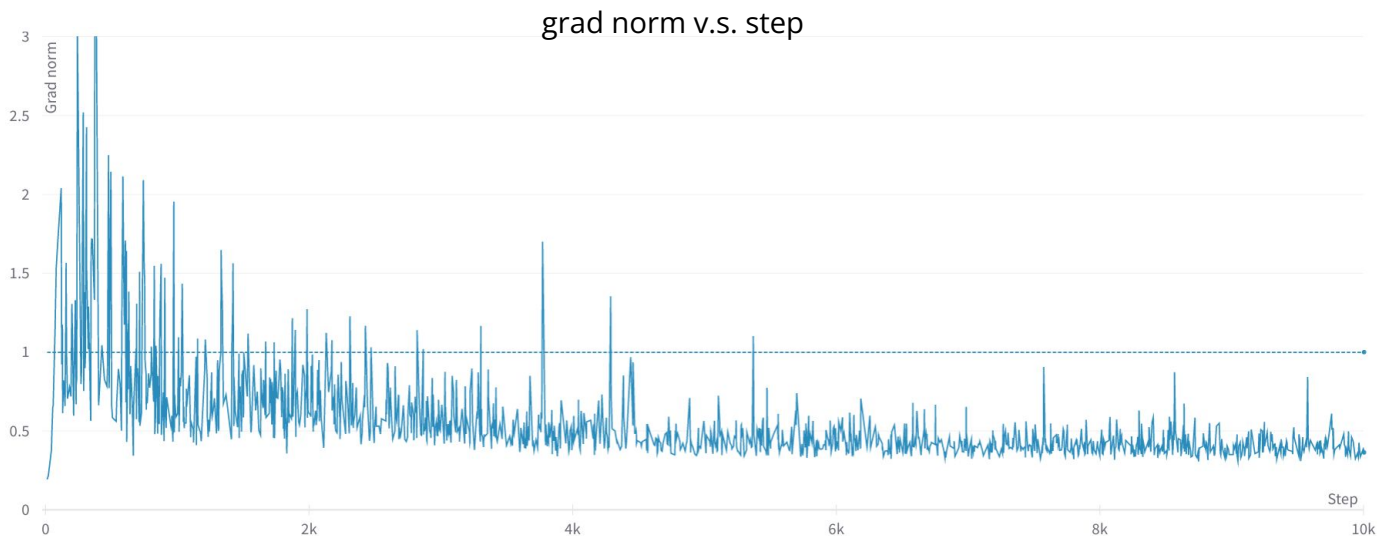
2-norm



## Problem 2: Visualize Gradient Norm

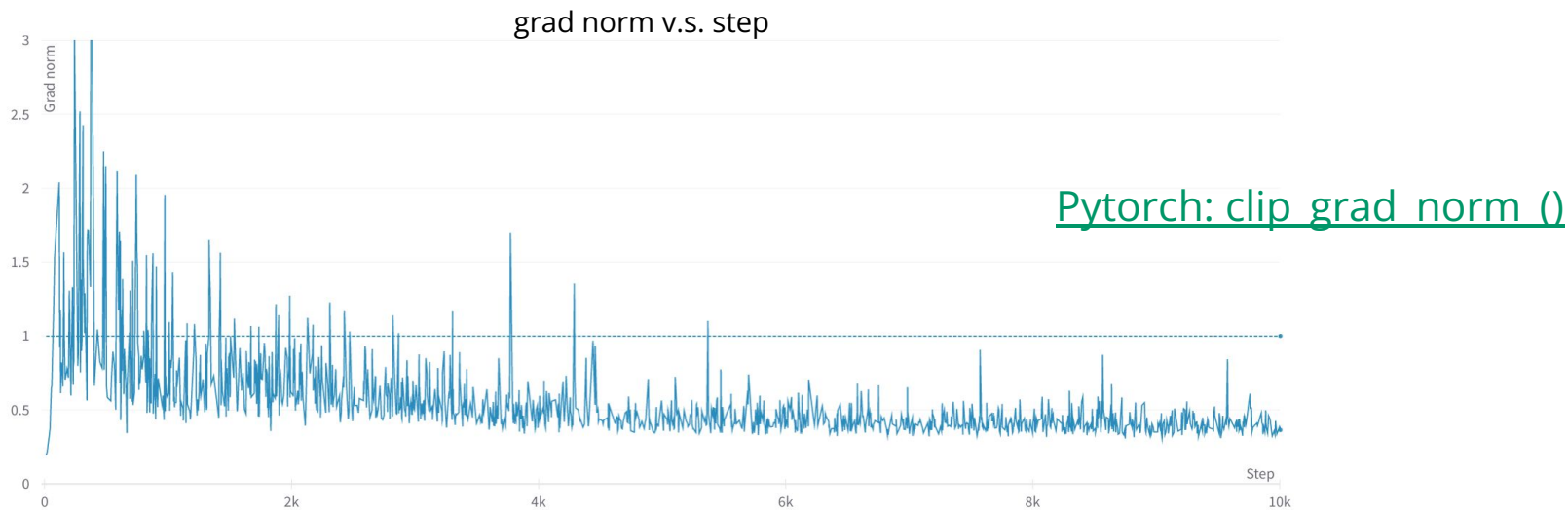
Step1: Apply clips gradient norm and set  $\text{max\_norm} = 1.0$  .

Step2: Make a plot of “gradient norm v.s step”.



## Problem 2: Visualize Gradient Norm

Step3: Circle two places with gradient explosion (where the clip\_grad\_norm function take effect)



# Problem 2: Overview

In this problem, you need to do

1. Plot the grad\_norm

```
def train_one_epoch(epoch_itr, model, task, criterion, optimizer, accum_steps=1):  
    .  
    .  
    .  
    optimizer.multiply_grads(1 / (sample_size or 1.0)) # (sample_size or 1.0) handles  
the case of a zero gradient  
    gnorm = nn.utils.clip_grad_norm_(model.parameters(), config.clip_norm) # grad norm  
clipping prevents gradient exploding → this is the grad_norm for every step
```

2. Circle two place with gradient explosion (if there is gradient explosion)

# JudgeBoi Guide

# Previously... Github Email Settings

Make sure that your NTU mail address is added to your Github account

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

**WeiChihChen**  
Your personal account

- Public profile
- Account
- Appearance
- Accessibility
- Notifications

Access

- Billing and plans
- Emails**
- Password and authentication
- Sessions

## Emails

|  |  |
|--|--|
| <b>timmy90617055@gmail.com</b> – Primary   |  |
| <ul style="list-style-type: none"><li>Not visible in emails</li><li>Receives notifications</li></ul> |  |
| <b>b08611018@ntu.edu.tw</b>  |  |
| <ul style="list-style-type: none"><li>Not visible in emails</li></ul>                                |  |

← 4 ) Check if NTU mail is added

Add email address

Email address Add

← 3 ) Add NTU mail address

Signed in as figisiwrf

- Set status
- Your profile
- Your repositories
- Your projects
- Your stars
- Your gists
- Your sponsors
- Upgrade
- Try Enterprise
- Feature preview
- Help
- Settings**

1 )

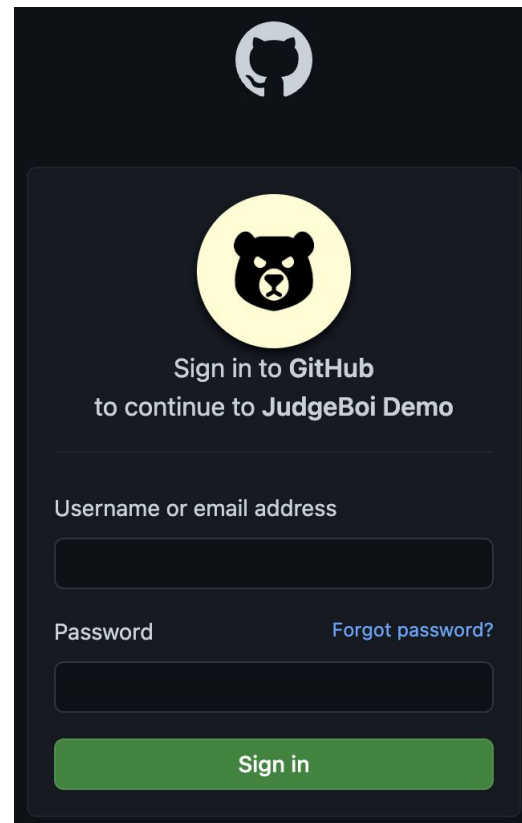
2 )

# Step 1: Register for Submission

Go to JudgeBoi, and Sign-in with Github

Fill in Github Username →

Fill in Github Password →



The screenshot shows a dark-themed login interface for 'JudgeBoi Demo'. At the top, there is a GitHub logo. Below it is a large yellow circular profile picture of a black bear. The text 'Sign in to GitHub' and 'to continue to JudgeBoi Demo' is centered. There are two input fields: 'Username or email address' and 'Password'. A 'Forgot password?' link is next to the password field. A green 'Sign in' button is at the bottom.

Sign in to GitHub  
to continue to JudgeBoi Demo

Username or email address

Password [Forgot password?](#)

Sign in

## Step 2: Select Homework

Choose the language you prefer



**JudgeBoi** **Select Homework** **testuser (Student)**

You have to make sure your role is "Student", or you will not be able to submit.

**Contact Info:** mlta-2023-spring@googlegroups.com  
**Course Website:** Machine Learning (2022 Spring) @ National Taiwan University  
**Designed by:** Zih-Ching Chen I-Ching Tseng Kai-Chieh Chang  
**Maintained by:** Zih-Ching Chen I-Ching Tseng

# GitHub account issues

- To login with a different GitHub account
  - You **must** logout from the Github website first
  - Then, you can click the logout button of the JudgeBoi website
- If you encounter problems when verifying the GitHub account with the NTU Mail, please post in the NTUCOOI [discussion](#).



## Step 3: Submit your Results

You can now submit results to the server and view the leaderboard.

**2) Check the leaderboard**

**3) Select your submissions**

**1) Submit results**



**Description**



Leaderboard



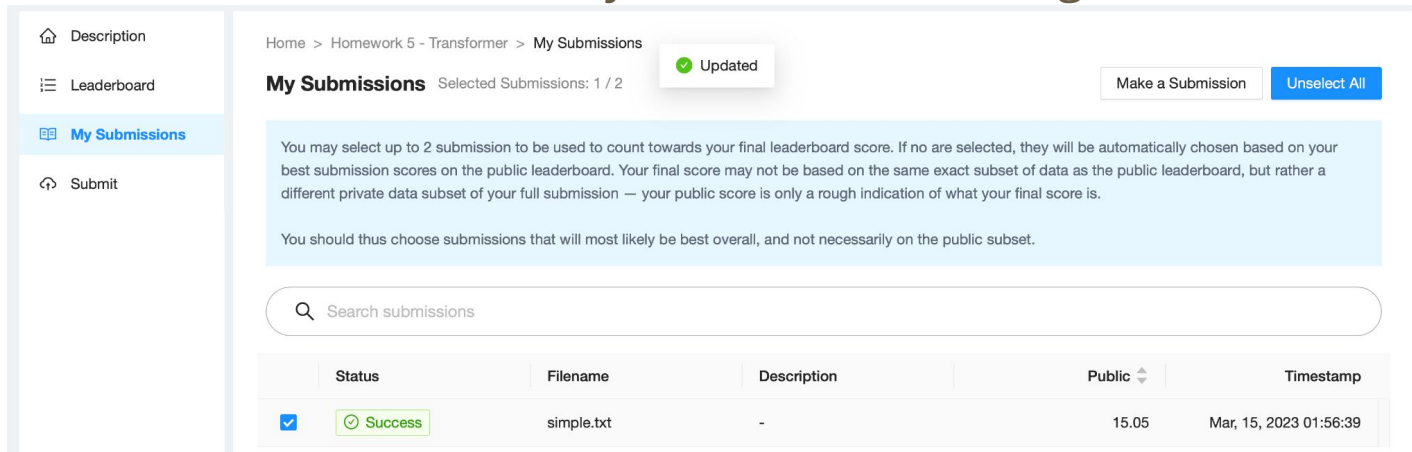
My Submissions



Submit

## Step 4: Select your submissions

- You can select up to 2 submissions.
- If none of your submissions is chosen, we will use the submission with the best public score.
- If your selection is successful, you will see a message box as below:



The screenshot shows the 'My Submissions' page for 'Homework 5 - Transformer'. The left sidebar contains links for 'Description', 'Leaderboard', 'My Submissions' (active), and 'Submit'. The main content area has a breadcrumb 'Home > Homework 5 - Transformer > My Submissions' and a green 'Updated' status box. Below this is a 'My Submissions' header with 'Selected Submissions: 1 / 2' and buttons for 'Make a Submission' and 'Unselect All'. A light blue informational box explains the selection process: 'You may select up to 2 submission to be used to count towards your final leaderboard score. If no are selected, they will be automatically chosen based on your best submission scores on the public leaderboard. Your final score may not be based on the same exact subset of data as the public leaderboard, but rather a different private data subset of your full submission — your public score is only a rough indication of what your final score is. You should thus choose submissions that will most likely be best overall, and not necessarily on the public subset.' Below this is a search bar 'Search submissions'. At the bottom is a table with columns: Status, Filename, Description, Public, and Timestamp.

| Status   | Filename   | Description | Public | Timestamp              |
|--|------------|-------------|--------|------------------------|
| <input checked="" type="checkbox"/> <span>Success</span> | simple.txt | -           | 15.05  | Mar, 15, 2023 01:56:39 |

# JudgeBoi Rules

- 5 submission quota per day, reset at **midnight**.
  - Guest users have no quota.
- Only **\*.txt** file is allowed, filesize should be smaller than **2MB**.
- We do limit the number of connections and request rate for each IP.
  - If you cannot access the website temporarily, please wait a moment.
- The system can be very busy as the deadline approaches.
  - If this prevents uploads, we do not offer additional submission opportunities.
- Please do not attempt to attack JudgeBoi.
- Every **Saturday** from **6:00 to 9:00** is our system maintenance time.
- For any JudgeBoi issues, please post on NTUCOOL discussion.
  - Discussion Link: [https://cool.ntu.edu.tw/courses/24108/discussion\\_topics/182915](https://cool.ntu.edu.tw/courses/24108/discussion_topics/182915)

# Regulations and Grading Policy

# Grading

- simple (public) +0.5 pts
- simple (private) +0.5 pts
- medium (public) +0.5 pts
- medium (private) +0.5 pts
- strong (public) +0.5 pts
- strong (private) +0.5 pts
- boss (public) +0.5 pts
- boss (private) +0.5 pts
- code submission +2 pts
- gradescope +4 pts

Total : 10 pts

# Code Submission

- **NTU COOL (2pts)**

- Compress your code and report into

**<student ID>\_hw5.zip**

**\* e.g. b08901020\_hw5.zip**

- We can only see your last submission.
- Do not submit your model or dataset.
- If your code is not reasonable, your semester grade x 0.9.

# GradeScope Submission

Answer the questions on GradeScope

# Deadlines

- JudgeBoi (Leaderboard)

**2023/04/14 23:59 (UTC+8)**

- Cool (Code submission)

**2023/04/14 23:59 (UTC+8)**

- Gradescope

**2023/04/14 23:59 (UTC+8)**



# Regulation

- You should NOT plagiarize, if you use any other resource, you should cite it in the reference. ( \* )
- You should NOT modify your prediction files manually.
- Do NOT share codes or prediction files with any living creatures.
- Do NOT attempt to submit your results more than 5 times a day.
- **Do NOT search or use additional data or pre-trained models.**
- Your **final grade x 0.9** 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)
  - [https://cool.ntu.edu.tw/courses/24108/discussion\\_topics/184642](https://cool.ntu.edu.tw/courses/24108/discussion_topics/184642)
- Email
  - [mlta-2023-spring@googlegroups.com](mailto:mlta-2023-spring@googlegroups.com)
  - The title should begin with “[hw5]”
- TA hour
  - Each Friday during class
  - Each Monday night on [Google meet](#)
    - 19:00 - 20:00 (Mandarin)
    - 20:00 - 21:00 (English)

- If you encounter this when downloading prediction files in Kaggle,

**Right Click** and **Save As** to save the file in your local directory.

