



Pattern Recognition Final Project Announcement

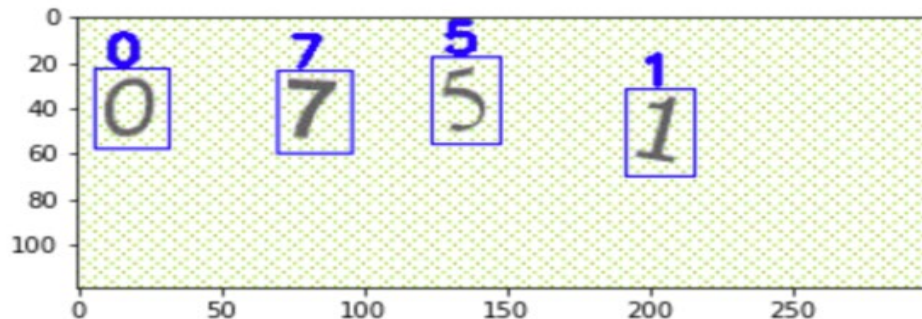
Lastest update: **2023.05.03 12:00**

Final Project

- Deadline: **June. 7, Wed. at 23:59**
 - Code assignment (60%)
 - Participate in a competition and put forth your best effort to achieve a strong performance.
 - Report (40%)
 - Include a detailed description of your research process and implementation in the report.
- Competition: [Link](#)


Captcha Recognition

- Train a model to predict all the digits in the image.
 - Task 1: Single character in the image.
 - Task 2: Two characters in the image (order matters).
 - Task 3: Four characters in the image (order matters).



40 '0751'

Join the Competition [\[HERE\]](#)

 Community Prediction Competition

Captcha Hacker 2023 Spring

PR Final Project

9 days to go

[Overview](#) [Data](#) [Code](#) [Discussion](#) [Leaderboard](#) [Rules](#) [Team](#) [Host](#) [Submissions](#) [Submit Predictions](#) [...](#)

Overview [Edit](#)



Description	(2023 Spring) Pattern Recognition Final Project - captcha recognition
Evaluation	


[+ Add Page](#)

Download the Dataset


[Overview](#) [Data](#) [Code](#) [Discussion](#) [Leaderboard](#) [Rules](#) [Team](#) [Host](#) [Submissions](#) [Submit Predictions](#) [...](#)

png, csv


dataset (2 directories, 1 files)  

About this directory 


This file does not have a description yet.



test
3 directories



train
3 directories, 1 files



sample_submission.csv
290.02 kB

Data Explorer

79.87 MB

- dataset
 - test
 - train
 - sample_submission.csv

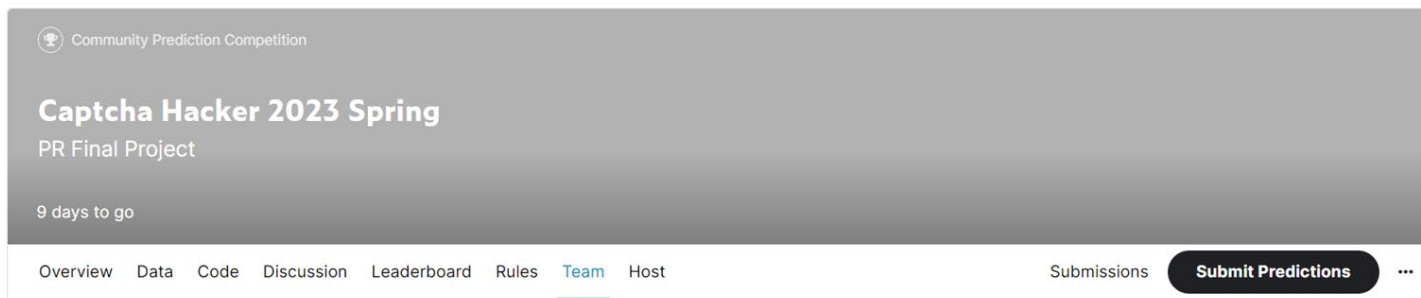
Summary

- 18.5k files
- 4 columns

[Download All](#) [+ New Version](#)

Kaggle Team Name

- You MUST set the team name as your Student_ID.



Your Team

Everyone that competes in a Competition does so as a team - even if you're competing by yourself. [Learn more.](#)

General

TEAM NAME

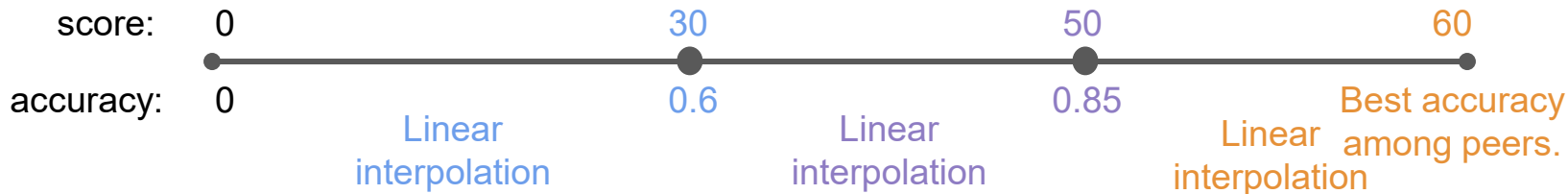
This name will appear on your team's leaderboard position.

Competition

- Can we use the pre-trained weights, e.g., ImageNet?
 - Pre-trained weights are available. Please indicate their usage in your reports.
- How can I get better performance?
 - Data pre-processing and augmentation.
 - Hyperparameter searching for model structure and optimizer (learning rate, ...).
 - Find some techniques from SoTA paper in the [paper with codes](#).
- Reasonable resources and techniques are permitted for use but you need to state in the report.
- If you are unsure whether certain techniques are allowed, please ask in the discussion forum first.

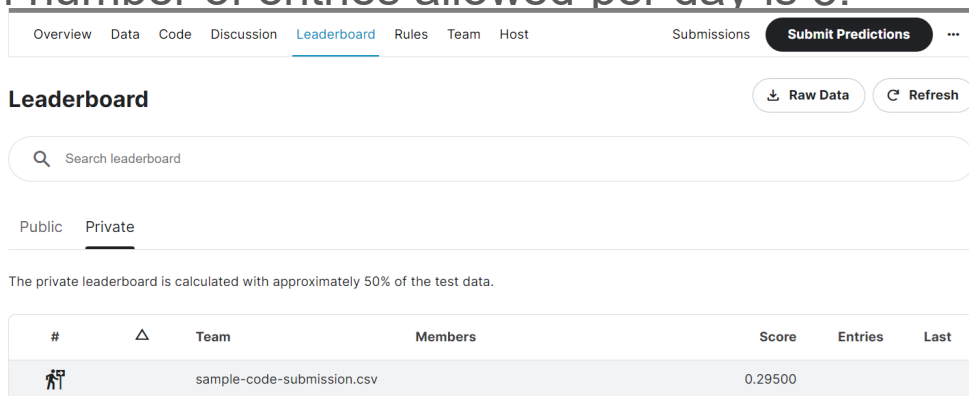
Grading

- Report: (40 points)
- Performance score (50 points): accuracy on the **private set**.
 - Weak Baseline: 60% (30 points)
 - Strong Baseline: 85% (20 points)
- Engage in competition with your peers (10 points):
 - The final 10 points will be awarded based on how you perform in comparison to your peers.
- Partial credit will be given if you are unable to meet the baselines.




Kaggle Submission (60%)

- Take a screenshot of your results on the **public leaderboard** and paste it on the report.
- Your score will be determined by the **private leaderboard**, but we will verify if your results match.
- The maximum number of entries allowed per day is 5.



The screenshot shows the Kaggle Leaderboard interface. At the top, there is a navigation bar with tabs: Overview, Data, Code, Discussion, Leaderboard (selected), Rules, Team, and Host. To the right of these tabs are 'Submissions' and a 'Submit Predictions' button. Below the navigation bar, the 'Leaderboard' section is visible, with a 'Raw Data' button and a 'Refresh' button. A search bar labeled 'Search leaderboard' is present. Below the search bar, there are tabs for 'Public' and 'Private' (selected). A note states: 'The private leaderboard is calculated with approximately 50% of the test data.' At the bottom, a table displays the leaderboard data.

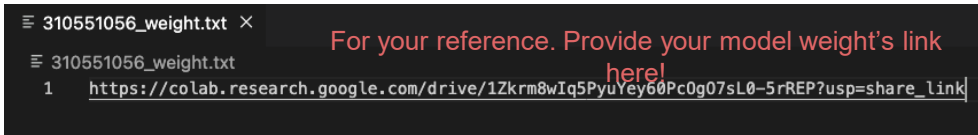
#	Team	Members	Score	Entries	Last
	sample-code-submission.csv		0.29500		

Report (40%)

- Environment details
 - Python version
 - ... (the more complete, the better)
- Implementation details
 - Model architecture
 - Hyperparameters
 - A detailed description of your experimental design, including the methodology and procedures employed in your study.
 - Compare with different method, ablation study, result analysis, ...
 - ... (the more complete, the better)
- No environment or implementation details may result in additional penalties.
- Note that you must provide a comprehensive and detailed explanation to receive a higher score (35 ~ 40 points).

Submission

- <STUDENT ID>_final.zip
 - Training code
 - Place all of your training code in the ./training/ directory.
 - **Inference code**
 - <STUDENT_ID>_inference.ipynb/.py
 - Report
 - <STUDENT_ID>_report.pdf
 - **Model weight**
 - <STUDENT ID>_weight.txt
 - Provide **a Google Drive link to your weights** & ensure access permissions are granted.
 - Environmental setting
 - If you implement your code using .py files, please also include a **requirements.txt** file.



Environment Setup

- For python file (.py)
 - It will be checked in our lab's servers. (Nvidia 2080Ti, cuda 11.3)
 - Providing a requirements.txt file can help us quickly rebuild your environment and accurately reproduce your results. (see [tutorial](#))
- For jupyter notebook file (.ipynb)
 - It will be checked in [google colab](#).
 - Please include the necessary pip install instructions in the first cell.

Example requirements.txt file

```
matplotlib>=2.2
numpy>=1.15.0, <1.21.0
pandas
pytest==4.0.1
```

```
[1] 1 # 若你要的套件原本 colab 沒有，請保留這些手動安裝指令，讓我們能快速重建你的環境
      2 !pip install transformers
      3 !pip install wandb
```

```
0s 1 # 以下套件 colab 原本就有，不用額外裝
      2 import numpy
      3 import torch
      4 import sklearn
      5
      6 # 以下套件 colab 原本沒有，須額外裝 (例如cell 1)
      7 import wandb
      8 import transformers
```

Test inference.py on our lab's server.

- We will take the following steps to reproduce your result. (**You can check it by yourself before submission**)
 - Build the virtual environment ([tutorial](#)) via

```
$ virtualenv -p <python version that you provided in report> myenv
```

```
$ source ./myenv/bin/activate
```

```
$ pip install -r requirements.txt
```
 - Download the <pre-trained weight that you provided in txt file>
 - **Modify the necessary path** in [inference code](#) (testing data, model weight, and so on...)
 - The code should be able to run successfully after modifications (**Otherwise, no points will be given**)
 - Check the results we reproduced and the Kaggle leaderboard.

Test inference.ipynb file on google colab

- We will take the following steps to reproduce your result. (**You can check it by yourself before submission**)
 - Download the <pre-trained weight that you provided in txt file>
 - **Modify the necessary path** in **inference code** (colab mount, testing data, model weight, and so on...)
 - Restart and run all, the code should be able to run successfully after modifications (**Otherwise, no points will be given**)
 - Check the results we reproduced and the Kaggle leaderboard.

Note

- Plagiarism
 - No points will be awarded if any plagiarism is discovered.
- Late policy
 - There is no late submission policy for the final project.
- For students who are new to deep learning
 - You may start with this [sample code](#) first

Have fun!

