

- Things to do for Phase 1
  - Read the [competition guidelines](#)
  - Download the data from Google Drive.
  - Read the [start code](#) and implement your first model
    - TODOs: dataloader, a model class, training loop, inference loop, export as CSV
    - Start code provide a simple example on how to preprocess the audio data
    - A piece of audio = A piece of image [1,2]
    - Some helpful resources
      - PyTorch image classification example ([link](#))
      - PyTorch dataloader ([Link](#))
    - Colab tips
      - Use GPU to speed up training (images = images.to('cuda'))
      - Kill the Colab session when you are not working
  - Join the [Kaggle competition](#) and submit your CSV to the autograder
- Things you may need to consider for Phase 2
  - Use cross-validation to train and validate your model (recommended, you can also merge several folders to get an overall training set and use the remaining folders to serve as validation set)
  - Use some tricks like hyper-params tuning(e.g. [optuna](#)) to get the best checkpoint of your model and make your training process more efficient and stable.

## Reference

[1] Chen, H., Xie, W., Vedaldi, A., & Zisserman, A. (2020, May). Vggsound: A large-scale audio-visual dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 721-725). IEEE.

[2] Van Horn, Grant, et al. "Exploring fine-grained audiovisual categorization with the SSW60 dataset." *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2022.