Hello, my Kaggle team name is 20664537\_ma.

MODEL summary – line 214:

* Categorical data: Fully-connected neural networks
* Image data: CNN (pretrained models)
* Noisy Text data: LSTM

Ensemble was done with different pretrained models for image data but the same architecture for the categorical data and noisy text data.

You can run my code in Linux environment. The code should be placed in the same parent folder as ‘uw\_cs480\_fall20’, where the data is stored. This code uses pytorch. Additional packages used includes PIL and nltk (for word processing). The entire list of packages/dependencies can be found in the appendix.

To train: put main(model\_name) under \_\_main\_\_ and run python main.py.

* e.x. main(‘resnet152’)
* This function will save best model ckpt of the MODEL mentioned above.

To validate: put ensemble\_val(model\_names) under \_\_main\_\_ and run python main.py.

* e.x. ensemble\_val([‘resnet152’, ‘densenet121’, ‘vgg19\_bn’])
* This function will return the accuracy on each of the models as well as the ensemble accuracy when combining them together

To test: put ensemble\_test(model\_names) under \_\_main\_\_ and run python main.py.

* e.x. ensemble\_test([‘resnet152’, ‘densenet121’, ‘vgg19\_bn’])
* This function will save a .csv containing the predicted category

The ckpt of the ensembled models can be found in my onedrive (they are too large to be submitted in the learn dropbox):

<https://uofwaterloo-my.sharepoint.com/:f:/g/personal/z74ma_uwaterloo_ca/En6-MHXglFdAnpFFs74i0jEBJfnlzRD1BTYL2zNpiJPLfQ?e=6tyXLq>

Appendix:

import numpy as np

import pandas as pd

import os

import math

from PIL import Image

import pickle

import matplotlib.pyplot as plt

import torch

from torch.utils.data import Dataset, DataLoader, WeightedRandomSampler, random\_split

from torch.autograd import Variable

import torch.nn as nn

import torch.nn.functional as F

from torch.optim import \*

from torchvision import transforms, models

# For noisyTextDescription cleaning

import nltk

# nltk.download('punkt')

# nltk.download('stopwords')

from nltk.tokenize import word\_tokenize

from nltk.corpus import stopwords

from nltk.stem.porter import PorterStemmer

import string

from scipy import stats