import sys

import argparse

import os

import json

indir = '/u/cs401/A1/data/';

def preproc1( comment , steps=range(1,11)):

''' This function pre-processes a single comment

Parameters:

comment : string, the body of a comment

steps : list of ints, each entry in this list corresponds to a preprocessing step

Returns:

modComm : string, the modified comment

'''

modComm = ''

if 1 in steps:

print('TODO')

if 2 in steps:

print('TODO')

if 3 in steps:

print('TODO')

if 4 in steps:

print('TODO')

if 5 in steps:

print('TODO')

if 6 in steps:

print('TODO')

if 7 in steps:

print('TODO')

if 8 in steps:

print('TODO')

if 9 in steps:

print('TODO')

if 10 in steps:

print('TODO')

return modComm

def main( args ):

allOutput = []

for subdir, dirs, files in os.walk(indir):

for file in files:

fullFile = os.path.join(subdir, file)

print( "Processing " + fullFile)

data = json.load(open(fullFile))

# TODO: select appropriate args.max lines

# TODO: read those lines with something like `j = json.loads(line)`

# TODO: choose to retain fields from those lines that are relevant to you

# TODO: add a field to each selected line called 'cat' with the value of 'file' (e.g., 'Alt', 'Right', ...)

# TODO: process the body field (j['body']) with preproc1(...) using default for `steps` argument

# TODO: replace the 'body' field with the processed text

# TODO: append the result to 'allOutput'

fout = open(args.output, 'w')

fout.write(json.dumps(allOutput))

fout.close()

if \_\_name\_\_ == "\_\_main\_\_":

parser = argparse.ArgumentParser(description='Process each .')

parser.add\_argument('ID', metavar='N', type=int, nargs=1,

help='your student ID')

parser.add\_argument("-o", "--output", help="Directs the output to a filename of your choice", required=True)

parser.add\_argument("--max", help="The maximum number of comments to read from each file", default=10000)

args = parser.parse\_args()

if (args.max > 200272):

print( "Error: If you want to read more than 200,272 comments per file, you have to read them all." )

sys.exit(1)

main(args)