```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Fri Mar 19 18:32:11 2021

@author: cmheilig
"""

from bs4 import SoupStrainer

EID_BASE_PATH_b0 = normpath(expanduser('~/cdc-corpora/eid_b0/'))
x = create_mirror_tree(EID_BASE_PATH_b0, calculate_mirror_dirs(eid_dframe.path))
EID_BASE_PATH_u3 = normpath(expanduser('~/cdc-corpora/eid_u3/'))
x = create_mirror_tree(EID_BASE_PATH_u3, calculate_mirror_dirs(eid_dframe.path))

EID_BASE_PATH_u3 = normpath(expanduser('~/cdc-corpora/eid_u3_20210218/'))
eid_uni_html = [html_reduce_space_u(read_uni_html(EID_BASE_PATH_u3 + path))
                    for path in tqdm(eid_dframe.mirror_path)]
# 11504/11504 [00:09<00:00, 1153.86it/s]
# pickle.dump(eid_html_u3, open('eid_uni_html.pkl', 'wb'))
# pickle.dump(eid_uni_html, open('eid_uni_html.pkl', 'wb'))
# eid_uni_html = pickle.load(open('eid_uni_html_20210218.pkl', 'rb'))

eid_art_html = [html_reduce_space_u(read_uni_html(EID_BASE_PATH_u3 + path))
                    for path in tqdm(eid_art_frame.mirror_path)]
# 11211/11211 [02:07<00:00, 88.07it/s]
# pickle.dump(eid_art_html, open('eid_art_html.pkl', 'wb'))
# eid_art_html = pickle.load(open('eid_art_html.pkl', 'rb'))

only_title = SoupStrainer(name='title')

eid_uni_titles = [BeautifulSoup(html, 'lxml', parse_only=only_title).title.string.strip()
                  for html in tqdm(eid_uni_html)]
# pickle.dump(eid_uni_titles, open('eid_uni_titles.pkl', 'wb'))
# eid_uni_titles = pickle.load(open('eid_uni_titles.pkl', 'rb'))
eid_art_titles = [BeautifulSoup(html, 'lxml', parse_only=only_title).title.string.strip()
                  for html in tqdm(eid_art_html)]
# pickle.dump(eid_uni_titles, open('eid_art_titles.pkl', 'wb'))
# eid_art_titles = pickle.load(open('eid_art_titles.pkl', 'rb'))

sum([ title in ['500 - Emerging Infectious Diseases journal',
   'CDC - Website Temporarily Unavailable'] for title in eid_uni_titles ])

# Check for nonunique titles as a screen for errors
z_uni = { w: eid_uni_titles.count(w) for w in sorted(set(eid_uni_titles)) }
z_art = { w: eid_art_titles.count(w) for w in sorted(set(eid_art_titles)) }
# print([ v for v in z_uni.values() if v > 1 ]) # [260, 2, 2, 2, 2, 53, 2, ... ]

z_uni_freq = { k: v for (k, v) in z_uni.items() if v > 1 } # length 38
z_art_freq = { k: v for (k, v) in z_art.items() if v > 1 } # length 38
# z_uni_freq == z_art_freq # True
# { w: list(z_uni_freq.values()).count(w) for w in sorted(set(z_uni_freq.values())) }
# {2: 30, 3: 5, 4: 1, 53: 1, 260: 1} # focus on titles that occur 53 or 260 times
{ k: v for (k, v) in z_art.items() if v > 4 }
# {'500 - Emerging Infectious Diseases journal': 260,
#  'CDC - Website Temporarily Unavailable': 53}
```

```
 57
 58 l_uni = [len(v) for v in eid_uni_titles]
 59 l_art = [len(v) for v in eid_art_titles]
 60
 61 { w: l_uni.count(w) for w in sorted(set(l_uni)) }
 62 { w: l_art.count(w) for w in sorted(set(l_art)) }
 63
 64 # [ v for v in z_uni.values() if v > 1 ]
 65 { k: v for (k, v) in z_uni.items() if v > 1 } == \
 66    { k: v for (k, v) in z_art.items() if v > 1 } # True # only articles erred
 67
 68 # z_uni or z_art where count > 1
 69
 70 # limit to articles where
 71 #    <title> is '500 - Emerging Infectious Diseases journal' or 'CDC - Website Temporarily
 71 Unavailable'
 72
 73 #%%
 74
 75 # Iterate over all u3 HTML files in eid_dframe
 76 # 1. Read HTML file from disk to string in memory
 77 # 2. Parse soup and extract only soup.title.strip()
 78 # 3. Apply condition to detect titles indicative of errors
 79 #    '500 - Emerging Infectious Diseases journal' or
 80 #    'CDC - Website Temporarily Unavailable'
 81 #    anything else?
 82 # Iterate over eid_dframe rows corresponding to erroneous titles
 83 # 4. Attempt to retrieve b0 from web
 84 #    If length-0 retrieved, try again
 85 #    If length > 0, check soup.title
 86 #       If title indicative of error, try again
 87 #       Else write b0, write u3
 88
 89 EID_BASE_PATH_b0 = normpath(expanduser('~/cdc-corpora/eid_b0/'))
 90 EID_BASE_PATH_u3 = normpath(expanduser('~/cdc-corpora/eid_u3/'))
 91
 92 uni_redo_tf = [ title in ['500 - Emerging Infectious Diseases journal',
 93    'CDC - Website Temporarily Unavailable'] for title in eid_uni_titles ]
 94 eid_dframe_x = eid_dframe.loc[uni_redo_tf]
 95
 96 eid_sizes_x_b0 = [mirror_raw_html(url, EID_BASE_PATH_b0 + path, print_url = False, timeout
 96 = 8)
 97                      for url, path in tqdm(zip(eid_dframe_x.url,
 97                      eid_dframe_x.mirror_path),
 98                                            total=313)]
 99
100 eid_html_x_b0 = [read_raw_html(EID_BASE_PATH_b0 + path)
101                      for path in tqdm(eid_dframe_x.mirror_path)]
102
103 for path in tqdm(eid_dframe_x.mirror_path):
104    mirror_raw_to_uni(EID_BASE_PATH_b0 + path, EID_BASE_PATH_u3 + path, counter=None)
```