```
#!/usr/bin/env python
# coding: utf-8

# In[85]:


import numpy as np
import pandas as pd


# In[2]:


buisness_df = pd.read_csv("yelp_academic_dataset_business.csv", encoding="utf-8")


# In[86]:


buisness_df.head(2)


# In[4]:


buisness_df["hours"].isnull()


# In[5]:


buisness_df["hours"].loc[192604]


# ## List SubAttr

# In[6]:


attr_list = ['BusinessParking', 'Music', 'GoodForMeal', 'NoiseLevel', 'Ambience', 'Dietar
yRestrictions']
sub_attr_list = []
for h, sub_attr in enumerate(attr_list):
    res = buisness_df['attributes'].str.extract(sub_attr + '[\\\':\s]+["]+{*((?:[a-zA-Z-_
\\\':,\s])+)')
    res_no_na = res.dropna().values
    res_no_na = [i[0] for i in res_no_na]
    #print(res_no_na)
    sub_attr = set()
    for i in res_no_na:
        i_splits = i.split(",")
        for j in i_splits:
            a = j.split(":")
            a = a[0].strip()
            a = a.replace('"', "")
            sub_attr.add(a)
    print(sub_attr)
    sub_attr_list.append(list(sub_attr))


# In[26]:


max_len = max([len(i) for i in sub_attr_list])
print(max_len)
len(sub_attr_list)
for el in sub_attr_list:
    while len(el) < max_len:
        el.append("")
```

```
# In[27]:


import csv
with open('sub_attr.csv', 'w', newline='') as csvfile:
    spamwriter = csv.writer(csvfile, delimiter=',',
                            quotechar='|', quoting=csv.QUOTE_MINIMAL)
    spamwriter.writerow(["index"] + attr_list)
    for nb, i in enumerate(range(max_len)):
            spamwriter.writerow([nb] + [sub_attr_list[j][i] for j in range(len(sub_attr_l
ist))])


# ## Subattr tables

# In[28]:


sub_attr_list


# In[31]:


re1 = buisness_df['attributes'].str.extractall('BusinessParking' + '[\\\':\s]+["]+{*((?:[
a-zA-Z-_\\\':,\s])+)')
idx = re1.index
idx = [i[0] for i in idx]

re1 = re1[0].values
ret = [create_dict(i) for i in re1]

BusinessParking_df = pd.DataFrame(ret, index=idx)


# In[32]:


BusinessParking_df


# In[33]:


re1 = buisness_df['attributes'].str.extractall('Music' + '[\\\':\s]+["]+{*((?:[a-zA-Z-_\\
\':,\s])+)')
idx = re1.index
idx = [i[0] for i in idx]

re1 = re1[0].values
ret = [create_dict(i) for i in re1]

Music_df = pd.DataFrame(ret, index=idx)


# In[34]:


re1 = buisness_df['attributes'].str.extractall('GoodForMeal' + '[\\\':\s]+["]+{*((?:[a-zA
-Z-_\\\':,\s])+)')
idx = re1.index
idx = [i[0] for i in idx]

re1 = re1[0].values
ret = [create_dict(i) for i in re1]

GoodForMeal_df = pd.DataFrame(ret, index=idx)
```

```
# In[35]:


re1 = buisness_df[’attributes’].str.extractall(’NoiseLevel’ + ’[\\\’:\s]+["]+{*((?:[a-zA-
Z-_\\\’:,\s])+)’)
idx = re1.index
idx = [i[0] for i in idx]

re1 = re1[0].values
ret = [create_dict(i) for i in re1]

NoiseLevel_df = pd.DataFrame(ret, index=idx)


# In[36]:


re1 = buisness_df[’attributes’].str.extractall(’Ambience’ + ’[\\\’:\s]+["]+{*((?:[a-zA-Z-
_\\\’:,\s])+)’)
idx = re1.index
idx = [i[0] for i in idx]

re1 = re1[0].values
ret = [create_dict(i) for i in re1]

Ambience_df = pd.DataFrame(ret, index=idx)


# In[37]:


re1 = buisness_df[’attributes’].str.extractall(’DietaryRestrictions’ + ’[\\\’:\s]+["]+{*(
(?:[a-zA-Z-_\\\’:,\s])+)’)
idx = re1.index
idx = [i[0] for i in idx]

re1 = re1[0].values
ret = [create_dict(i) for i in re1]

DietaryRestrictions_df = pd.DataFrame(ret, index=idx)


# In[30]:


test_str = buisness_df[’attributes’].loc[100]
import re
z = re.findall(’BusinessParking’ + ’[\\\’:\s]+["]+{*((?:[a-zA-Z-\\\’:,\s])+)’, test_str)
print(z[0])

def to_bool(s):
    if(isinstance(s,str)):
        s = s.strip()
    else:
        return s
    if (s.lower() == ’true’):
        return True
    elif (s.lower() == ’false’):
        return False
    else: print("eerr")

def create_dict(str1):
    res = dict()
    if(len(str1) <= 0):
        return res
    list1 = str1.split(",")
    for i in list1:
        a = i.split(":")
        b = a[0].strip()
        b = b.replace("’", "")
        if(len(a) <= 1):
            a.append(True)
        res[b] = to_bool(a[1])
    #print(res)
    return res

result = create_dict(z[0])
print(result)



# In[38]:


buisness_df_copy = buisness_df.copy()


# In[39]:


buisness_df_copy[buisness_df_copy["business_id"] == "huZ1fY8x0-915Mo-lUxt7Q"]


# In[40]:


BusinessParking_df = BusinessParking_df.join(buisness_df_copy["business_id"])
Music_df = Music_df.join(buisness_df_copy["business_id"])
GoodForMeal_df = GoodForMeal_df.join(buisness_df_copy["business_id"])
NoiseLevel_df = NoiseLevel_df.join(buisness_df_copy["business_id"])
Ambience_df = Ambience_df.join(buisness_df_copy["business_id"])
DietaryRestrictions_df = DietaryRestrictions_df.join(buisness_df_copy["business_id"])


# In[45]:


BusinessParking_df.sample(10)


# In[42]:


validated_idx = BusinessParking_df[BusinessParking_df["validated"] == True]["business_id"
]
lot_idx = BusinessParking_df[BusinessParking_df["lot"] == True]["business_id"]
garage_idx = BusinessParking_df[BusinessParking_df["garage"] == True]["business_id"]
street_idx = BusinessParking_df[BusinessParking_df["street"] == True]["business_id"]
valet_idx = BusinessParking_df[BusinessParking_df["valet"] == True]["business_id"]


# In[43]:


tuple_list_val = [(0,i) for i in validated_idx]
tuple_list_l = [(1,i) for i in lot_idx]
tuple_list_g = [(2,i) for i in garage_idx]
tuple_list_s = [(3,i) for i in street_idx]
tuple_list_v = [(4,i) for i in valet_idx]

final_list_businessparking = tuple_list_val + tuple_list_l + tuple_list_g + tuple_list_s
+ tuple_list_v


# In[44]:


with open("final_list_businessparking.csv", ’w’, newline=’\n’) as myfile:
    wr = csv.writer(myfile, quoting=csv.QUOTE_ALL)
    for el in final_list_businessparking:
        wr.writerow(el)


# In[46]:


Music_df.head()


# In[47]:


dj_idx = Music_df[Music_df["dj"] == True]["business_id"]
juke_idx = Music_df[Music_df["jukebox"] == True]["business_id"]
kara_idx = Music_df[Music_df["karaoke"] == True]["business_id"]
vid_idx = Music_df[Music_df["video"] == True]["business_id"]
live_idx = Music_df[Music_df["live"] == True]["business_id"]
bg_idx = Music_df[Music_df["background_music"] == True]["business_id"]
no_idx = Music_df[Music_df["no_music"] == True]["business_id"]

tuple_list_d = [(0,i) for i in dj_idx]
tuple_list_b = [(1,i) for i in juke_idx]
tuple_list_n = [(2,i) for i in kara_idx]
tuple_list_j = [(3,i) for i in vid_idx]
tuple_list_l = [(4,i) for i in live_idx]
tuple_list_v = [(5,i) for i in bg_idx]
tuple_list_k = [(6,i) for i in no_idx]

final_list_music = tuple_list_d + tuple_list_b + tuple_list_n + tuple_list_j + tuple_list
_l + tuple_list_v + tuple_list_k


# In[48]:


with open("final_list_music.csv", 'w', newline='\n') as myfile:
    wr = csv.writer(myfile, quoting=csv.QUOTE_ALL)
    for el in final_list_music:
        wr.writerow(el)


# In[49]:


GoodForMeal_df


# In[50]:


no_idx = GoodForMeal_df[GoodForMeal_df["lunch"] == True]["business_id"]
dj_idx = GoodForMeal_df[GoodForMeal_df["dessert"] == True]["business_id"]
live_idx = GoodForMeal_df[GoodForMeal_df["brunch"] == True]["business_id"]
vid_idx = GoodForMeal_df[GoodForMeal_df["breakfast"] == True]["business_id"]
juke_idx = GoodForMeal_df[GoodForMeal_df["dinner"] == True]["business_id"]
bg_idx = GoodForMeal_df[GoodForMeal_df["latenight"] == True]["business_id"]

tuple_list_d = [(0,i) for i in no_idx]
tuple_list_b = [(1,i) for i in dj_idx]
tuple_list_n = [(2,i) for i in live_idx]
tuple_list_j = [(3,i) for i in vid_idx]
tuple_list_l = [(4,i) for i in juke_idx]
tuple_list_v = [(5,i) for i in bg_idx]

final_list_goodformeal = tuple_list_d + tuple_list_b + tuple_list_n + tuple_list_j + tupl
e_list_l + tuple_list_v


# In[51]:
```

```
with open("final_list_goodformeal.csv", 'w', newline='\n') as myfile:
    wr = csv.writer(myfile, quoting=csv.QUOTE_ALL)
    for el in final_list_goodformeal:
        wr.writerow(el)


# In[52]:


NoiseLevel_df.fillna(False, inplace=True)
NoiseLevel_df["average"] = np.logical_or(NoiseLevel_df["average"], NoiseLevel_df["uaverage"])
NoiseLevel_df["very_loud"] = np.logical_or(NoiseLevel_df["very_loud"], NoiseLevel_df["uvery_loud"])
NoiseLevel_df["loud"] = np.logical_or(NoiseLevel_df["loud"], NoiseLevel_df["uloud"])
NoiseLevel_df["quiet"] = np.logical_or(NoiseLevel_df["quiet"], NoiseLevel_df["uquiet"])


# In[53]:


NoiseLevel_df = NoiseLevel_df[['very_loud', 'loud', 'average', 'quiet','business_id']]


# In[54]:


NoiseLevel_df


# In[58]:


bg_idx = NoiseLevel_df[NoiseLevel_df["very_loud"] == True]["business_id"]
no_idx = NoiseLevel_df[NoiseLevel_df["loud"] == True]["business_id"]
dj_idx = NoiseLevel_df[NoiseLevel_df["average"] == True]["business_id"]
juke_idx = NoiseLevel_df[NoiseLevel_df["quiet"] == True]["business_id"]

tuple_list_d = [(0,i) for i in bg_idx]
tuple_list_b = [(1,i) for i in no_idx]
tuple_list_n = [(2,i) for i in dj_idx]
tuple_list_j = [(3,i) for i in juke_idx]

final_list_noiselevel = tuple_list_d + tuple_list_b + tuple_list_n + tuple_list_j


# In[59]:


with open("final_list_noiselevel.csv", 'w', newline='\n') as myfile:
    wr = csv.writer(myfile, quoting=csv.QUOTE_ALL)
    for el in final_list_noiselevel:
        wr.writerow(el)


# In[60]:


Ambience_df


# In[61]:


bg_idx = Ambience_df[Ambience_df["romantic"] == True]["business_id"]
no_idx = Ambience_df[Ambience_df["intimate"] == True]["business_id"]
dj_idx = Ambience_df[Ambience_df["classy"] == True]["business_id"]
juke_idx = Ambience_df[Ambience_df["hipster"] == True]["business_id"]
juke1_idx = Ambience_df[Ambience_df["divey"] == True]["business_id"]
```

```
juke2_idx = Ambience_df[Ambience_df["touristy"] == True]["business_id"]
juke3_idx = Ambience_df[Ambience_df["trendy"] == True]["business_id"]
juke4_idx = Ambience_df[Ambience_df["upscale"] == True]["business_id"]
juke5_idx = Ambience_df[Ambience_df["casual"] == True]["business_id"]

tuple_list_a = [(0,i) for i in bg_idx]
tuple_list_b = [(1,i) for i in no_idx]
tuple_list_c = [(2,i) for i in dj_idx]
tuple_list_d = [(3,i) for i in juke_idx]
tuple_list_e = [(4,i) for i in juke1_idx]
tuple_list_f = [(5,i) for i in juke2_idx]
tuple_list_i = [(6,i) for i in juke3_idx]
tuple_list_j = [(7,i) for i in juke4_idx]

final_list_ambience = tuple_list_a + tuple_list_b + tuple_list_c + tuple_list_d + tuple_l
ist_e + tuple_list_f + tuple_list_i + tuple_list_j


# In[62]:


print(len(final_list_ambience), len(set(final_list_ambience)))


# In[63]:


with open("final_list_ambience.csv", 'w', newline='\n') as myfile:
    wr = csv.writer(myfile, quoting=csv.QUOTE_ALL)
    for el in final_list_ambience:
        wr.writerow(el)


# In[64]:


DietaryRestrictions_df.head(2)


# In[65]:


bg_idx = DietaryRestrictions_df[DietaryRestrictions_df["dairy-free"] == True]["business_i
d"]
no_idx = DietaryRestrictions_df[DietaryRestrictions_df["gluten-free"] == True]["business_
id"]
dj_idx = DietaryRestrictions_df[DietaryRestrictions_df["vegan"] == True]["business_id"]
juke_idx = DietaryRestrictions_df[DietaryRestrictions_df["kosher"] == True]["business_id"
]
juke1_idx = DietaryRestrictions_df[DietaryRestrictions_df["halal"] == True]["business_id"
]
juke2_idx = DietaryRestrictions_df[DietaryRestrictions_df["soy-free"] == True]["business_
id"]
juke3_idx = DietaryRestrictions_df[DietaryRestrictions_df["vegetarian"] == True]["busines
s_id"]

tuple_list_a = [(0,i) for i in bg_idx]
tuple_list_b = [(1,i) for i in no_idx]
tuple_list_c = [(2,i) for i in dj_idx]
tuple_list_d = [(3,i) for i in juke_idx]
tuple_list_e = [(4,i) for i in juke1_idx]
tuple_list_f = [(5,i) for i in juke2_idx]
tuple_list_i = [(6,i) for i in juke3_idx]


final_list_dietaryrestrictions = tuple_list_a + tuple_list_b + tuple_list_c + tuple_list_
c + tuple_list_d + tuple_list_e + tuple_list_f + tuple_list_i


# In[66]:
```

```
with open("final_list_dietaryrestrictions.csv", ’w’, newline=’\n’) as myfile:
    wr = csv.writer(myfile, quoting=csv.QUOTE_ALL)
    for el in final_list_dietaryrestrictions:
        wr.writerow(el)


# In[ ]:




# ## Friends

# In[67]:


user_df = pd.read_csv("yelp_academic_dataset_user.csv", encoding="utf-8")


# In[68]:


user_df.head(3)


# In[69]:


user_df.columns


# In[70]:


user_df_minimal = user_df[[’elite’, ’friends’, ’user_id’]].copy()


# In[71]:


user_df_minimal.head(2)


# In[77]:


def split_friends(str1):
    if not(isinstance(str1, str)):
        return []
    str1 = str1.strip()
    str1 = str1.replace(’"’, "")
    str1 = str1.replace("’", "")
    str1 = str1.replace(’]’, "")
    str1 = str1.replace(’[’, "")
    str1 = str1.strip()
    str1 = str1.split(",")
    for i in str1:
        i = i.strip()
        i = i.replace(’"’, "")
        i = i.replace("’", "")
        i = i.strip()
    return str1

res_friends = user_df_minimal["friends"].apply(split_friends)
print(res_friends)
```

```
# In[78]:


combined_df = pd.DataFrame(res_friends).join(user_df_minimal["user_id"])


# In[79]:


combined_df.loc[3]["friends"][0]


# In[80]:


def create_friendly_tuples(row):
    res = []
    for f in row["friends"]:
        res.append((f.strip(), row["user_id"]))

    return set(res)

res_friends2 = combined_df.apply(create_friendly_tuples, axis=1)


# In[81]:


res2 = set()
for row in res_friends2:
    for tuple1 in row:
        res2.add(tuple1)



# In[87]:


res2


# In[88]:


print(len(res2))


# In[93]:


res300 = set()
for pair in res2:
    res300.add(tuple(sorted(pair)))


# In[94]:


print(len(res300))


# In[95]:


res300


# In[96]:
```

```
with open("user_friends.csv", ’w’, newline=’\n’) as myfile:
    wr = csv.writer(myfile, quoting=csv.QUOTE_ALL)
    for el in res300:
        wr.writerow(el)


# In[580]:


user_df


# ## Category

# In[97]:


new_buisness_df = pd.read_csv("yelp_academic_dataset_business.csv", encoding="utf-8")


# In[98]:


new_buisness_df = new_buisness_df[["categories", "business_id"]]


# In[99]:


arr_cat = new_buisness_df["categories"].unique()


# In[100]:


pd.DataFrame(range(arr_cat.shape[0]), arr_cat).to_csv("categories.csv")


# In[ ]:





# In[101]:


new_buisness_df["categories"].values


# In[103]:


def split_cat(str1):
    if not(isinstance(str1, str)):
        return np.nan
    str1 = str1.split(",")
    for i in str1:
        i = i.lower()
        i = i.strip()
    return str1

res = new_buisness_df["categories"].apply(split_cat)


# In[104]:


res5 = set()
res = res[˜res.isna()]
```

```
for row in res:
    for el in row:
        el = el.strip()
        el = el.lower()
        res5.add(el)


# In[105]:


res5


# In[106]:


pd.DataFrame(range(len(res5)), res5).to_csv("cat_new.csv")


# In[134]:


# using newly built res5
cat_map_df = pd.DataFrame(range(len(res5)), res5)

#using already existing cat_lower map
#cat_map_df = pd.read_csv("cat_lower.csv")


# In[ ]:





# In[136]:


cat_map_df.to_dict()[0]


# In[137]:


cat_map_dict = cat_map_df.to_dict()[0]


# In[138]:


cat_map_dict["active life"]


# In[111]:


new_buisness_df


# In[112]:


new_buisness_df["categories"] = new_buisness_df["categories"].apply(split_cat)


# In[113]:


cat_map_complete_df = pd.DataFrame(new_buisness_df["categories"].explode()).join(new_buis
ness_df["business_id"])


# In[114]:


cat_map_reduced_df = cat_map_complete_df[˜cat_map_complete_df["categories"].isna()].copy(
)


# In[116]:


def cat_mapping(str1):
    if not isinstance(str1,str):
        return np.nan
    str1 = str1.strip()
    return cat_map_dict[str1]

cat_map_reduced_df["categories"] = cat_map_reduced_df["categories"].apply(str.lower)
cat_map_reduced_df["categories"] = cat_map_reduced_df["categories"].apply(cat_mapping)


# In[117]:




# In[118]:


len(cat_map_reduced_df)


# In[121]:


cat_map_reduced_df.drop_duplicates(inplace = True)


# In[139]:


cat_map_reduced_df.to_csv("cat_map.csv")


# ## New separated Sub_attributes maps

# In[7]:


sub_attr_list


# In[15]:


clean_sub_attr = []

def clean_sub_attr_list(list_of_list):
        clean_sub_attr = []
        for list1 in list_of_list:
            temp_list = []
            for el in list1:
                if el[0] == "u":
                    el = el[1:]
                el = el.strip()
                el = el.replace("’", "")
                el = el.replace(’"’, ’’)
                el = el.strip()
                el = el.lower()
                temp_list.append(el)
            temp_list = list(set(temp_list))
            clean_sub_attr.append(temp_list)
        return clean_sub_attr

clean_sub_attr = clean_sub_attr_list(sub_attr_list)


# In[16]:


clean_sub_attr


# In[55]:


business_parking = ['validated', 'lot', 'garage', 'street', 'valet']
music =  ['dj', 'jukebox', 'karaoke', 'video', 'live', 'background_music', 'no_music']
good_for_meal = ['lunch', 'dessert', 'brunch', 'breakfast', 'dinner', 'latenight']
noise_level = ['very_loud', 'loud', 'average', 'quiet']
ambience = ['classy','trendy','touristy', 'intimate', 'divey', 'upscale', 'casual', 'hips
ter','romantic']
dietary_restrictions = ['kosher', 'vegan', 'halal', 'vegetarian', 'dairy-free', 'gluten-f
ree', 'soy-free']





# In[57]:


pd.Series(business_parking, name="sub_attr").to_csv("business_parking.csv")
pd.Series(music, name="sub_attr").to_csv("music.csv")
pd.Series(good_for_meal, name="sub_attr").to_csv("good_for_meal.csv")
pd.Series(noise_level, name="sub_attr").to_csv("noise_level.csv")
pd.Series(ambience, name="sub_attr").to_csv("ambience.csv")
pd.Series(dietary_restrictions, name="sub_attr").to_csv("dietary_restrictions.csv")




# In[ ]:
```