**Finding duplicates using fuzzy**

Reference: This work is based on [“An Introduction to Duplicate Detection”](https://epdf.pub/an-introduction-to-duplicate-detection.html) by [Felix Naumann](https://scholar.google.com/citations?user=Pqf21y0AAAAJ&hl=en) and [Melanie Herschel](https://scholar.google.com/citations?user=K5VPw-IAAAAJ&hl=en).

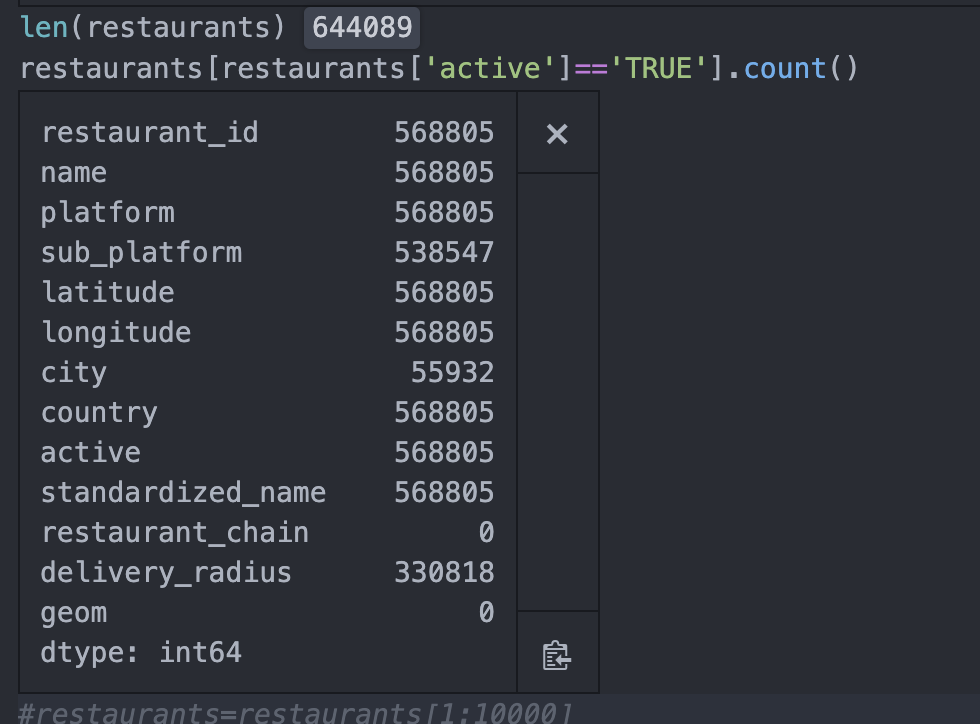
Understanding the Data Set:

As per my understanding below can be some of the reasons for duplicates and I need to work accordingly

* Somebody made a spelling mistake when entering data somewhere.
* Some people create duplicate accounts for multiple reasons like gain discounts etc.

|  | **restaurant\_id** | **name** | **platform** | **sub\_platform** | **latitude** | **longitude** | **city** | **country** | **active** | **standardized\_name** | **restaurant\_chain** | **delivery\_radius** | **geom** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 9744 | Loco Coco | caviar | NaN | 40.765004 | -73.965961 | NaN | US | TRUE | loco\_coco | NaN | NaN | NaN |
| **1** | 4903 | Musubi | caviar | NaN | 45.504641 | -122.643806 | NaN | US | FALSE | musubi | NaN | NaN | NaN |
| **2** | 4723 | The Bakers' Lounge | caviar | NaN | 38.924559 | -76.990115 | NaN | US | FALSE | the\_bakers\_lounge | NaN | NaN | NaN |
| **3** | 4128 | Parm - Battery Park City | caviar | NaN | 40.71418 | -74.015568 | NaN | US | TRUE | parm\_-\_battery\_park\_city | NaN | NaN | NaN |
| **4** | 5690 | Derek Test Merchant | caviar | NaN | 1 | 2 | NaN | US | FALSE | derek\_test\_merchant | NaN | NaN | NaN |

* As per my understanding name, latitude, longitude, city, country, standardized\_name, and restaurant\_chain can help us to find unique restaurants and remove duplicates.
* There is a high possibility for the restaurants to have different names in different platforms and sub-platforms.
* But some of the fields are very sparse and they ended up not being very useful.
* I wanted to use the city, country (address/ local area) or platforms, and sub-platforms combination as blocking factors to divide the whole data into small datasets to resolve this problem. But this data is also sparse so I was not able to utilize them in the duplicate removal process.



* Some records have inaccurate information like \[] or null values. I tried to remove those records of name field itself have this random data. If any other restaurant information fields (restaurant\_chain , standardized\_name , country ,city) have wrong information like “\[]” I tried to replace it with NULL values.
* I tried to depend on the below logic to consider the two records are duplicates.

1. When the restaurants have the same geolocation and similar name
2. same standardized name and similar geo. ( nearest geo points)
3. When the restaurants have the same name and similar geo.

* I have also tried to utilize the city and country values. But they are very sparse and were not much useful.
* I have also tried to replace the dining hall, Café any other restaurant's alternate names with the restaurant to improve results.

Top 10K results are saved in df2.csv

* I have also tried Clustering Methods Using NLP Techniques to remove duplicates, Overall fuzzy logic gave better results.

Sample code:

from sklearn.cluster import KMeans

clf = KMeans(n\_clusters=2,init='k-means++',n\_init=100,random\_state=0)

labels = clf.fit\_predict(restaurants)

previous\_cluster = 0

for index, sentence in enumerate(lst\_original):

if index > 0:

previous\_cluster = labels[index - 1]

cluster = labels[index]

if previous\_cluster != cluster: