In [1]:

```
import py_entitymatching as em #Import megallan entity matching library
import distance
```

In [7]:

```
# We have modified the schema of yelp and zomato for schema matching task.
yelp = em.read_csv_metadata("yelp_2.csv",key="id")
zomato = em.read_csv_metadata("zomato_2.csv",key="id")
```

In [3]:

```
# Yelp table schema
for attr in yelp.keys():
    print attr
```

```
id
Name
Phone Number
Zipcode
State
City
Address
Has Delivery
Has Takeout
Outdoor_seating
```

In [4]:

```
# Zomato table schema
for attr in zomato.keys():
    print attr
```

```
id
Restaurant Name
Contact Number
Zipcode
State
City
Address
Delivery
Takeout
Outdoor seating
```

In [5]:

```
# Carry out Schema Matching based on Jaccard distance and generate matching attribu
pairs = []
min_distance = 100
for yelp_attr in yelp.keys():
    for zomato_attr in zomato.keys():
        dist = distance.jaccard(yelp_attr,zomato_attr)
        if dist < min_distance:
            min_distance = dist
            selected_attribute = zomato_attr
    pairs.append([yelp_attr,selected_attribute])
    min_distance = 100
```

In [6]:

```
# Display the matched attribute pairs
pairs
```

Out[6]:

```
[['id', 'id'],
 ['Name', 'Restaurant Name'],
 ['Phone Number', 'Contact Number'],
 ['Zipcode', 'Zipcode'],
 ['State', 'State'],
 ['City', 'City'],
 ['Address', 'Address'],
 ['Has Delivery', 'Delivery'],
 ['Has Takeout', 'Takeout'],
 ['Outdoor_seating', 'Outdoor seating']]
```