The Best Chinese Restaurants In New York !

Using FourSquare API and KNN method to map **the most favorable** dinning places in Chinatown

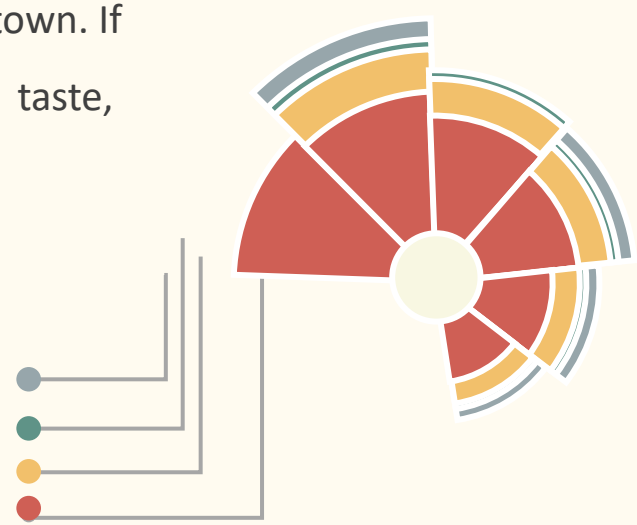
By MATT HAN LIU

1

Background

1 Background

New York attracts people from all over the world not only for its culture and fashion but also for variety of delicious food! When people especially the Chinese come to NY, the first place they will go might be Chinatown. If you want to taste Chinese food mixed with traditional and western taste, this is the place you must experience!



Among hundreds of Chinese restaurants in Chinatown, which of them worth trying? Let big data tell you!

Data

2 Data Source

In this case:

1. We use Foursuquare normal API (explore) to retrieve Chinese restaurants
   'Name', 'ID', 'Address'
2. By using each 'ID' we got, we use a premium API (venues detail) to retrieve its
   'Price_Tier', 'Likes_Count', 'Rating', 'Rating_Signals',
   'Polular_Timeframe_Today', 'Latitude', 'Longitude'
3. By using each 'ID' we got, we use a premium API (tips) to retrieve their
   'Tips', 'Agree_Count'
4. For Chinatown location, we use 'newyork_data.json' provided by Coursera

2

How to use data

For recommendations on the map

1. ‘Name’, ‘ID’, ‘Address’ will be used to build initial dataframe
2. ‘Price_Tier’, ‘Likes_Count’, ‘Rating’, ‘Rating_Signals’, ‘Polular_Timeframe_Today’, ‘Tips’, ‘Agree_Count’ will be added into initial dataframe, we will show different color for different price tier on the map, and also in case people don’t want to go to one restaurant during busy timeframe, we will show ‘Polular_Timeframe_Today’ in the tooltip of each point on the map
3. ‘Latitude’, ‘Longitude’ will be used to point each location on the map

For KNN method

1. ‘Name’, ‘ID’, ‘Address’, ‘Polular_Timeframe_Today’, ‘Tips’, ‘Agree_Count’ will be deleted and normalization will be provided.

Methodology

3 Methodology

We want to achieve three goals in the end:

1. A list with top 20 rating Chinese restaurants with most liked tip inside, so that we are able to know what attracts people the most in this restaurant.

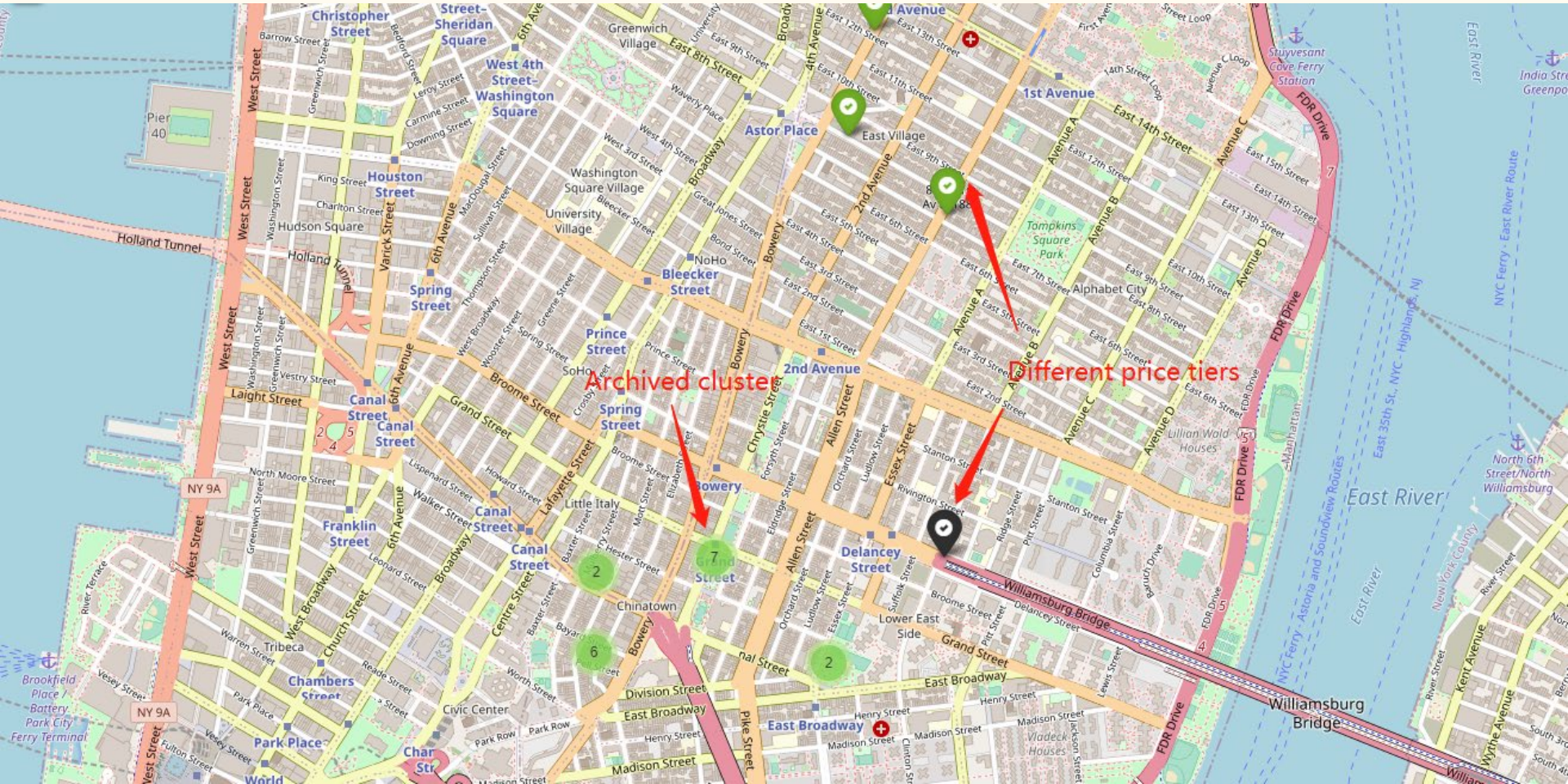
	NAME	ID	ADDRESS	Category	Price_Tier	Likes_Count	Rating	Rating_Signals	Tips	Agree_Count	Polular_Timeframe_Today
0	Big Wong King 大旺	3fd66200f964a520ede41ee3	67 Mott St		1	422	8.4	646	Take rice box 2 go - BEST BBQ Roast Pork (beat...	6	10:00 AM–9:00 PM
1	Great N.Y. Noodletown	3fd66200f964a520b1ea1ee3	28 Bowery		1	547	8	910	If it's soft-shell crab season, you know what ...	9	Noon–Midnight
2	Mission Chinese Food	4fa89bb2e4b0bad89524b84a	171 E Broadway		3	945	8.1	1374	NYers keep turning out in droves at this nonde...	21	Noon–3:00 PM5:00 PM–10:00 PM
3	Nom Wah Tea Parlor	49f220a3f964a520ee691fe3	13 Doyers St		1	1258	8.4	1790	This 92-year-old Chinatown restaurant serves t...	28	10:00 AM–9:00 PM
4	Wah Fung Number 1 Fast Food 華豐快飯店	4a96bf8ff964a520ce2620e3	79 Chrystie St		1	192	8.5	281	Go big or go home. $3 for more-than-you-can-ea...	13	11:00 AM–7:00 PM
6	Deluxe Green Bo Restaurant	3fd66200f964a520ceea1ee3	66 Bayard St		1	185	8.1	321	The Shanghainese food here is great. Order the...	5	11:00 AM–10:00 PM
7	Hop Kee	3fd66200f964a5206be61ee3	21 Mott St		2	203	7.9	353	Rarely do you return to a childhood spot and n...	3	Noon–10:00 PM

3 Methodology

2. A map showing top 20 choices in which:

Different colors represent different price tiers

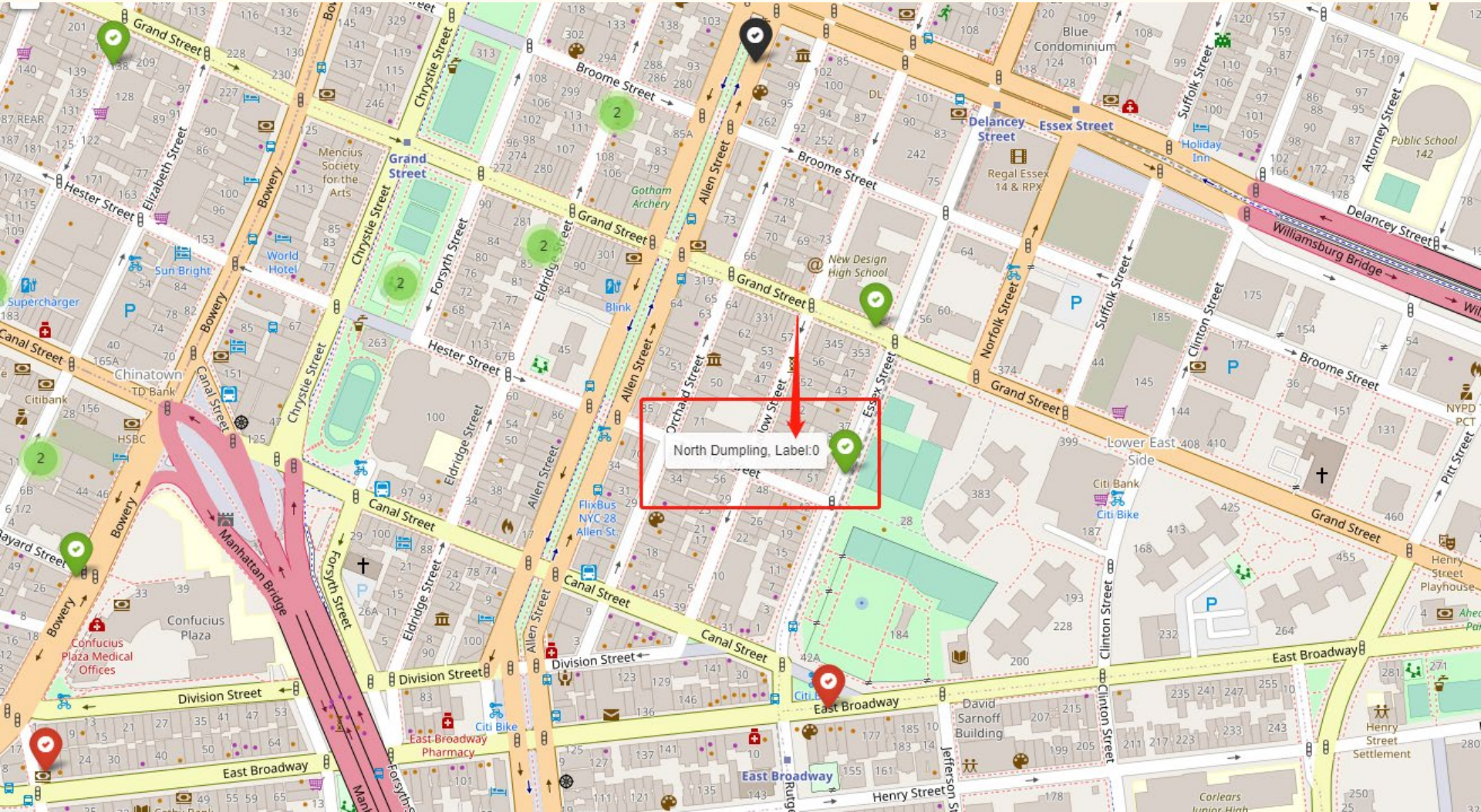
If two or more places nearby, then automatically achieved into one cluster



3 Methodology

2. A map similar to previous one but after KNN analysis :

Adding cluster label into the mouseover tooltip with different colors



3 Methodology

In order to achieve the First Goal:

1. Import json to read newyork_data.json provided
2. Use for loop to add all Neighborhoods into dataframe we created
3. Use .loc to locate the latitude and longitude of Chinatown

```
neighborhoods.loc[neighborhoods['Neighborhood']=='Chinatown']
```

	Borough	Neighborhood	Latitude	Longitude
100	Manhattan	Chinatown	40.715618	-73.994279

```
#get chinatown latitude and longtitude
latitude=neighborhoods.iloc[100,-2]
longitude=neighborhoods.iloc[100,-1]
print('Chinatown in New York is at ({},{})'.format(latitude,longitude))
```

```
Chinatown in New York is at (40.71561842231432,-73.99427936255978)
```

3 Methodology

4. Use requests.get(url).json() to retrieve Name, ID and Address from ‘explore’ endpoint of API
5. Use ID to retrieve rest parameters from ‘venue detail’ and ‘venue tips’ endpoints of API

```
for index,row in df_init.iterrows():
    id_venue=row['ID']
    url_venue='https://api.foursquare.com/v2/venues/{}?&client_id={}&client_secret={}&v={}'.format(id_venue,CLIENT_ID, CLIENT_SECRET,VERSION)
    url_tips='https://api.foursquare.com/v2/venues/{}/tips?&sort=popular&client_id={}&client_secret={}&v={}'.format(id_venue,CLIENT_ID, CLIENT_SECRET,VERSION)
    venue_detail=requests.get(url_venue).json()
    tips_detail=requests.get(url_tips).json()

    try:
        Price_Tier=venue_detail['response']['venue']['price']['tier']
    except:
        Price_Tier='N/A'
    try:
        Likes_Count=venue_detail['response']['venue']['likes']['count']
    except:
        Likes_Count='N/A'
    try:
        Rating=venue_detail['response']['venue']['rating']
    except:
        Rating='N/A'
    try:
        Rating_Signals=venue_detail['response']['venue']['ratingSignals']
    except:
        Rating_Signals='N/A'
    #use API parameter 'sort=popular' to retrive most liked tips
    Tips=''
    Agree_Count=0
    try:
```

3 Methodology

In order to achieve the Second Goal:

1. We delete non rating ones and sort by rating then get top 20 Chinese restaurants

```
# take out rating=N/A items
df1=df[df['Rating']!='N/A']
df1
```

	NAME	ID	ADDRESS	Category	Price_Tier	Likes_Count	Rating	Rating_Signals	Tips	Agree_Count	Polular_Timeframe_Today	Latitude	Longitude
0	Big Wong King 大旺	3fd66200f964a520ede41ee3	67 Mott St		1	422	8.4	646	Take rice box 2 go - BEST BBQ Roast Pork (beat...	6	10:00 AM–9:00 PM	40.7162	-73.9983
1	Great N.Y. Noodletown	3fd66200f964a520b1ea1ee3	28 Bowery		1	547	8	910	If it's soft-shell crab season, you know what ...	9	Noon–Midnight	40.715	-73.9969

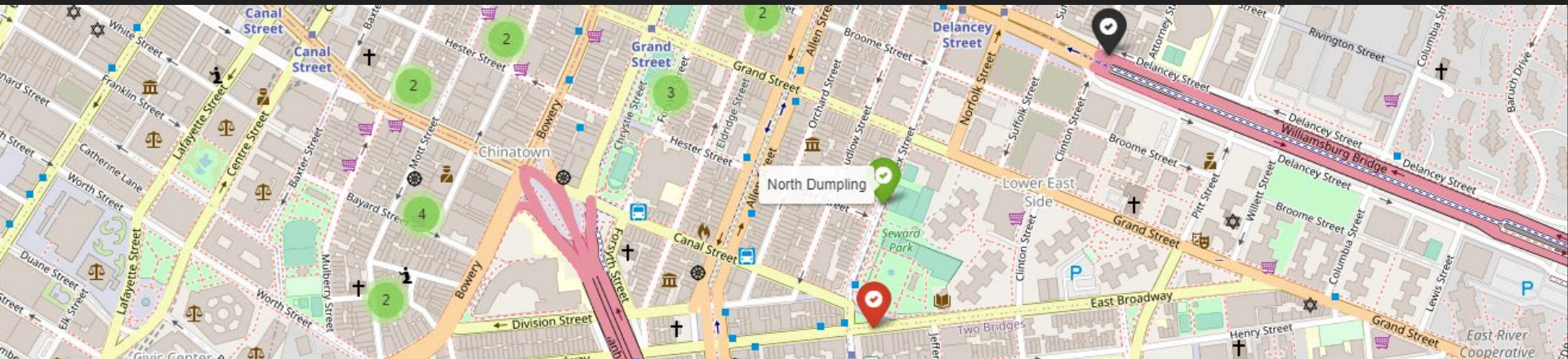
```
#sort by rating for next analysis
df2=df1.sort_values(by='Rating', ascending=False)
df2
```

	NAME	ID	ADDRESS	Category	Price_Tier	Likes_Count	Rating	Rating_Signals	Tips	Agree_Count	Polular_Timeframe_Today	Latitude	Longitude
23	Kings County Imperial	5b380f649deb7d00399fdf9d	168 1/2 Delancey St		N/A	67	9	91	Delicious and great for a group dinner. Soup d...	3	Noon–1:00 PM6:00 PM–10:00 PM	40.7178	-73.9856
34	Málà Project	5647ee82498e8bfc0ddef53d	122 1st Ave		2	436	8.9	550	I could come here just for the Mala peanuts (a...	10	Noon–3:00 PM5:00 PM–10:00 PM	40.7271	-73.9855

3 Methodology

2. Import folium to draw a NY map
3. From folium.plugins import MarkerCluster in order to gather nearby points into one cluster
4. We set up a rule to give different price tiers three colors, this can greatly help us identify price tiers without looking at words
5. Use tooltip to show names when mouseover

```
marker_cluster = folium.plugins.MarkerCluster().add_to(map_chinatown)
for index, row in df3.iterrows():
    locationlist = row[['Latitude','Longitude']].values.tolist()
    tooltip= '{}'.format(row['NAME'])
    if row['Price_Tier']==1 or row['Price_Tier']==2:
        folium.Marker(location=locationlist, popup='Address:{}, Rating:{}, Price Tier:{},/n Avoid Time:{}'.format(row['ADDRESS'],row['Rating'],row['Price_Tier'],row['Polular_Timeframe_To
    elif row['Price_Tier']==3 or row['Price_Tier']==4:
        folium.Marker(location=locationlist, popup='Address:{}, Rating:{}, Price Tier:{},/n Avoid Time:{}'.format(row['ADDRESS'],row['Rating'],row['Price_Tier'],row['Polular_Timeframe_To
    else:
        folium.Marker(location=locationlist, popup='Address:{}, Rating:{}, Price Tier:{},/n Avoid Time:{}'.format(row['ADDRESS'],row['Rating'],row['Price_Tier'],row['Polular_Timeframe_To
map_chinatown
```



3 Methodology

In order to achieve the Third Goal:

1. Preprocessing by

   (1) Take out items with N/A

   (2) Standardize all columns

Pre-processing

```
#take out N/A items and columns ID/ADDRESS/TIPS/AGREE COUNT/POPULAR TIMEFRAME/LATTITUDE/LONGITUDE
df4=df1[df1['Price_Tier']!='N/A']
df4=df4.iloc[:,[0,4,5,6,7]]
df4
```

	NAME	Price_Tier	Likes_Count	Rating	Rating_Signals
0	Big Wong King 大旺	1	422	8.4	646
1	Great N.Y. Noodletown	1	547	8	910

```
from sklearn.preprocessing import StandardScaler

X = df4.values[:,1:]
X = np.nan_to_num(X)
cluster_dataset = StandardScaler().fit_transform(X)
cluster_dataset
```

```
array([[-0.8273403 , -0.05733765,  0.88008132, -0.00942921],
       [-0.8273403 ,  0.30102265, -0.0325956 ,  0.52838487],
```

3 Methodology

2. Use KNN method to fit data, K=3
3. Add labels back to original dataframe

```
from sklearn.cluster import KMeans

num_clusters = 3

k_means = KMeans(init="k-means++", n_clusters=num_clusters, n_init=12)
k_means.fit(cluster_dataset)
labels = k_means.labels_

print(labels)
```

```
[0 0 1 1 0 0 2 2 1 1 0 2 0 0 1 1 1 0 0 0 0 2 2 0 0 2 1 0 2 2 0 0 1 2 0]
```

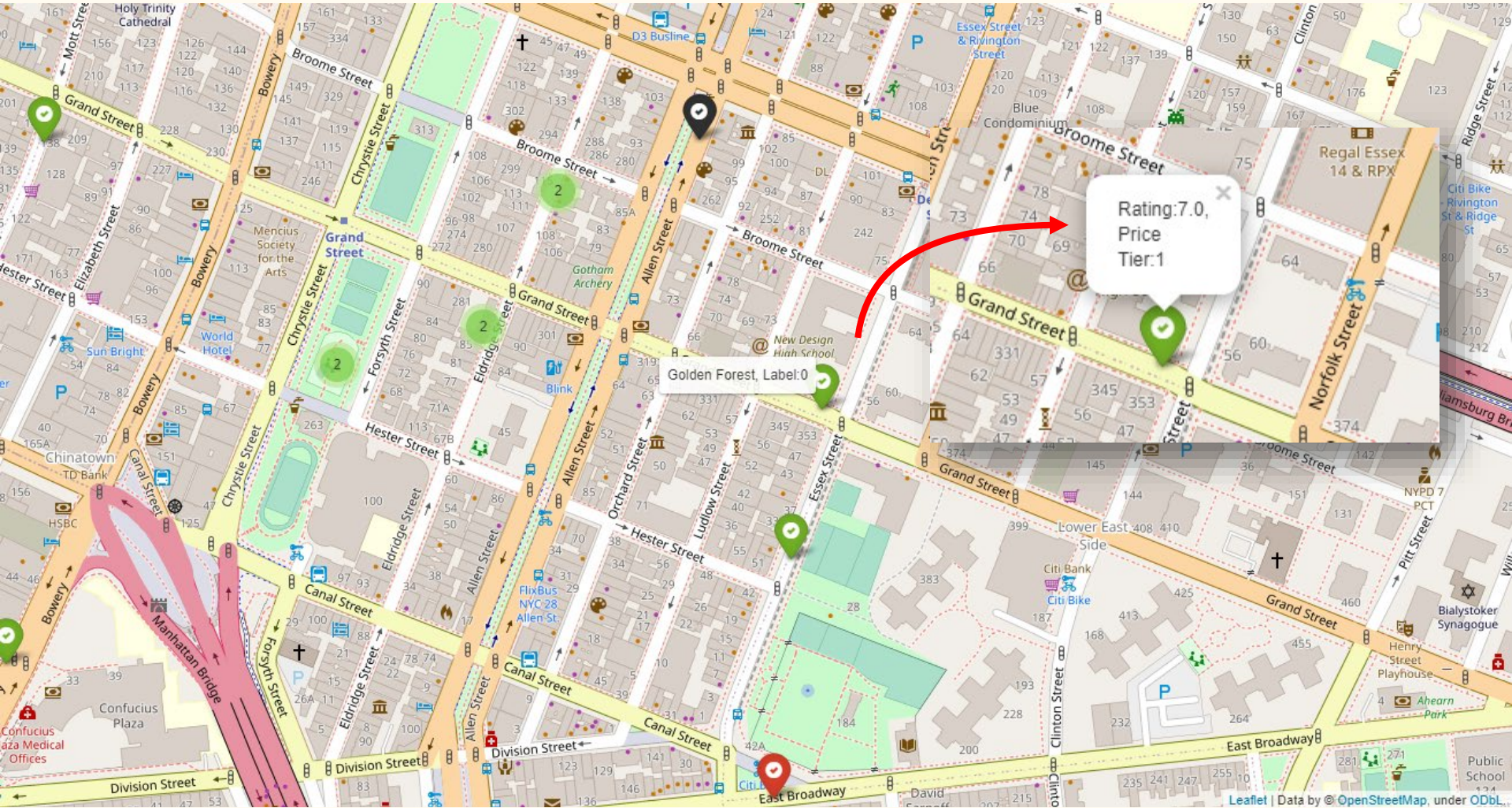
```
# add clustering labels
df_final=df1[df1['Price_Tier']!='N/A']
df_final["Labels"] = labels
df_final
```

	NAME	ID	ADDRESS	Category	Price_Tier	Likes_Count	Rating	Rating_Signals	Tips	Agree_Count	Polular_Timeframe_Today	Latitude	Longitude	Labels
0	Big Wong King 大旺	3fd66200f964a520ede41ee3	67 Mott St		1	422	8.4	646	Take rice box 2 go - BEST BBQ Roast Pork (beat...	6	10:00 AM–9:00 PM	40.7162	-73.9983	0
1	Great N.Y. Noodletown	3fd66200f964a520b1ea1ee3	28 Bowery		1	547	8	910	If it's soft-shell crab season, you know what ...	9	Noon–Midnight	40.715	-73.9969	0
2	Mission Chinese Food	4fa89bb2e4b0bad89524b84a	171 E Broadway		3	945	8.1	1374	NYers keep turning out in droves at this nonde...	21	Noon–3:00 PM5:00 PM–10:00 PM	40.7141	-73.9898	1
3	Nom Wah Tea Parlor	49f220a3f964a520ee691fe3	13 Doyers St		1	1258	8.4	1790	This 92-year-old Chinatown restaurant serves t...	28	10:00 AM–9:00 PM	40.7145	-73.9982	1
4	Wah Fung Number 1 Fast Food 華豐快飯店	4a96bf8ff964a520ce2620e3	79 Chrystie St		1	192	8.5	281	Go big or go home. $3 for more-than-you-can-ea...	13	11:00 AM–7:00 PM	40.7173	-73.9942	0

3 Methodology

4. Finally show on the map with labels on mouseover tooltips,

but remain the same info with previous map when you click the button



4

Results

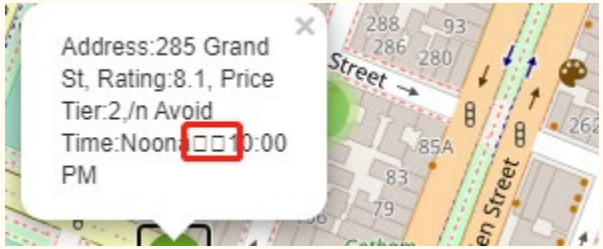
4 Results

- Most Chinese Restaurants’ ratings are among 8.0 and 9.0
- Taking 1st July 2020 as an example, the most common popular timeframes are 10:00 AM–9:00 PM, meaning they are popular all day long
- Most Chinese Restaurants 'prices are comfortable, price tiers from 1 to 2
- Most of the top 20 places are within 10 mins walking distance

Discussion

5 Discussion

- Due to the limt of this API, we can only retrieve at most 100 records one time, then after preprocessing, there will not be enough samples for KNN analysis
- There are display problems of Chinese words or ‘-’ with folium map



- For further development, we can consider get photos from API and try to save on local notebook then insert them into map
- There is display error on github for folium map, now temporary solution is to use nbviewer.jupyter.org to preview

6

Conclusion

6 Conclusion

- We successfully developed a one stop solution for new comers to New York Trying to find the best Chinese restaurant in New York
- We provide 3 ways to display, one is dataframe which can show most liked tips text, one is a map showing different price tiers and ratings, one is similar map using KNN analysis with 3 labels
- For further development, we may need higher rank API developer account to retrieve more records

THANK YOU