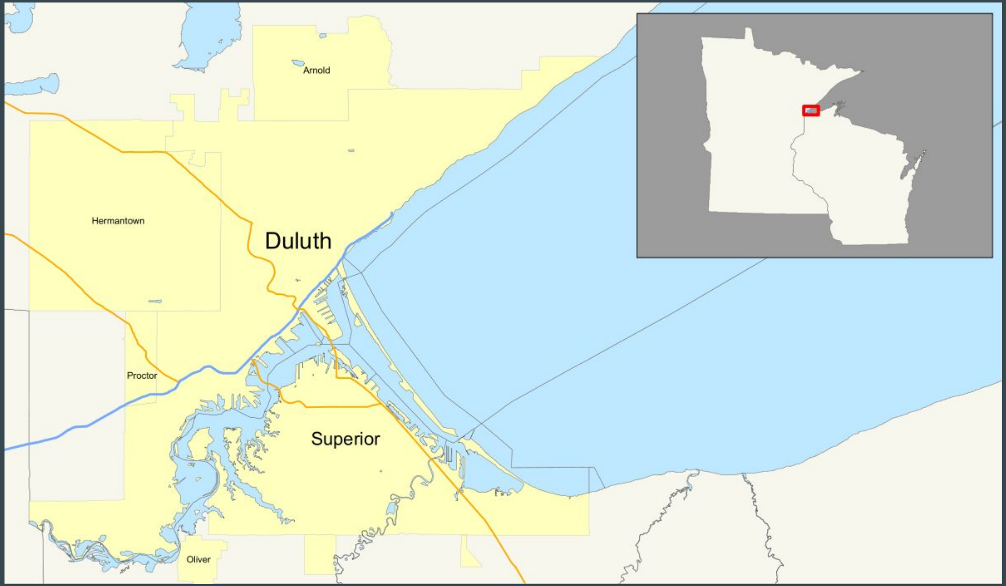
Duluth/Superior Map Processing Analysis

Riya Mokashi

Analysis and Background Overview

This analysis was done with MongoDB and utilized data from Open Street Map.



The locations of interest are Superior, WI and Duluth, MN which are adjacent to each other and are only split by state and lake lines.

Problems in the Analysis

- Naming Inconsistencies
- Odd Characters
- Numerical to Text Conversions for Street Names
- Inaccurate Listings/Duplicates

```
In [26]:

# Find problems with tag names
import tags as tags_processor
tag_problems = tags_processor.process_map(OSMFILE)
print "The number of keys in each of the 'problem' categories:"
print tag_problems['counts']
unique_key_names = tags_processor.unique_tag_keys(OSMFILE)
print "There are {} unique tag key names in the data set.".format(len(unique_key_names))

The number of keys in each of the 'problem' categories:
{'problemchars': 1, 'upper': 169, 'lower': 53080, 'upper_colon': 441, 'numbers': 1904,
'multiple_colons': 131, 'lower_colon': 38315, 'other': 12}
There are 609 unique tag key names in the data set.
```

Restaurant and Food Analysis

Analyzing the food availability in the area pointed to a preference for American/Western Cuisine

At the same time it was interesting to find that open street map was at times inaccurately labeling the data at hand

```
Cuisine counts for restaurant nodes:
Counter({'pizza': 5, 'american': 2, 'italian': 2, 'sandwich': 1, 'mexican': 1, 'fish': 1,
'regional': 1, 'burger': 1, 'snack': 1, 'chicken': 1, 'italian_pizza': 1})


Cuisine counts for cafe nodes:
Counter({'ice_cream': 1, 'coffee_shop': 1})


Cuisine counts for fast_food nodes:
Counter({'burger': 8, 'sandwich': 4, 'mexican': 1, 'pizza': 1})
```

```
from difflib import SequenceMatcher

def similarity_by_name(a, b):
    if 'name' in a and 'name' in b:
        a = a['name'].replace('the', '').lower()
        b = b['name'].replace('the', '').lower()
        return SequenceMatcher(None, a, b).ratio()
    else:
        return 0

subject = food_nodes_without_cuisine[0]

processed_nodes = []
food_nodes_with_same_amenity = [n for n in food_nodes_with_cuisine_and_amenity if n['amenity'] == s
ubject['amenity']]
for node in food_nodes_with_same_amenity:
    if 'name' in node:
        processed_nodes.append({'similarity': similarity_by_name(subject, node), 'node': node})

print "Subject name: {}   Subject amenity: {} \n".format(subject['name'], subject['amenity'])
sorted_results = sorted(processed_nodes, key=lambda k: k['similarity'], reverse=True)
for result in sorted_results[:5]:
    node = result['node']
    score = '%.3f' % result['similarity']
    print "Similarity score: {}   Name: {}   Cuisine: {}".format(score, node['name'], node['cuisine
'])
```

```
Subject name: Red Mug Coffee-Cafe   Subject amenity: cafe

Similarity score: 0.474   Name: Northern Shores Coffee   Cuisine: coffee_shop
```

Top Tens Analysis

I thought it would also be interesting to explore what the most common amenities/facilities were in the area.

It was unsurprising to find that a majority of them were public facilities like schools and parking lots.

```
{u'count': 596, u'_id': u'parking'}
{u'count': 46, u'_id': u'school'}
{u'count': 37, u'_id': u'restaurant'}
{u'count': 34, u'_id': u'fuel'}
{u'count': 30, u'_id': u'place_of_worship'}
{u'count': 26, u'_id': u'fast_food'}
{u'count': 13, u'_id': u'bank'}
{u'count': 11, u'_id': u'grave_yard'}
{u'count': 10, u'_id': u'theatre'}
{u'count': 10, u'_id': u'bar'}
```

Future Steps

This analysis is largely meant to simple give an overall idea of the makeup and general layout of these two cities.

In the future I would like to add comparisons between Superior and Duluth as well as more indepth analysis of individual locations with the consideration that there may be many labeling mistakes in the data.

Taking the time to more carefully look through the data may prove to be fruitful.

Thanks!