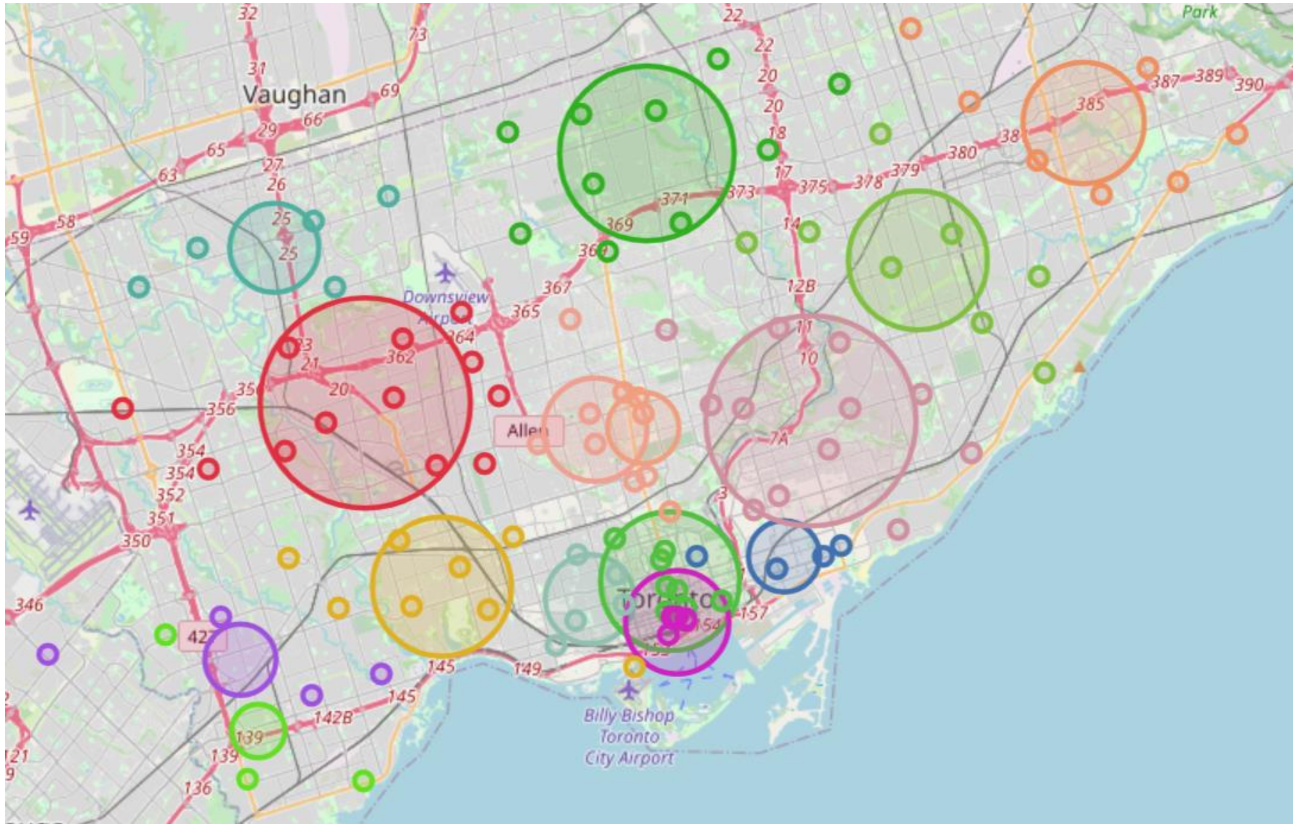
Building a Restaurant Recommendation System with Foursquare

JEREMY NATHAN

MOTIVATION

Given an input restaurant, produce a list of recommended restaurants based on their similarity to the input restaurant's attributes

- Requirements: Fast, scalable, logical
- Initially will focus on a subset of restaurants in Toronto



DATA

- Obtained from Foursquare API
- 2 Endpoints used: Explore, Venue Likes
- Focused on restaurants in Scarborough, Toronto for prototype
- Preprocessing: dropping duplicates, fast-food restaurants
- ~460 restaurant sample size

[8]:

	name	postalCode	category	lat	lng
id					
4b6718c2f964a5203f3a2be3	Harvey's	M1B	Restaurant	43.800020	-79.198307
4b914562f964a520d4ae33e3	Caribbean Wave	M1B	Caribbean	43.798558	-79.195777
4ba6f126f964a520ee7839e3	Pizza Pizza	M1B	Pizza	43.806613	-79.178445
4b02dff3f964a520974a22e3	Swiss Chalet	M1B	Restaurant	43.800236	-79.198366
4ceaa2f0f8653704f906bec4	Mr Jerk	M1B	African	43.801262	-79.199758
...	...	...	...	...	...
5529a011498eddb919b0f2f5	Kori Sushi	M2K	Japanese	43.791613	-79.392267
4bd396d041b9ef3b799c00e6	Sun Star Chinese Cuisine 翠景小炒	M2K	Chinese	43.787914	-79.381234
4b32769df964a5205a0c25e3	Harvey's	M2K	Restaurant	43.792287	-79.393024
4bdc7dd8c79cc9287ecc86e9	Maxim's Cafe and Patisserie	M2K	Café	43.787863	-79.380751
5404d153498ebbc7332f3e4e	Kaga Sushi	M2K	Japanese	43.787758	-79.381090

460 rows × 5 columns

METHODOLOGY

- Create a recommendation matrix R such that R_{ij} yields restaurant j's "recommendability" relative to restaurant i.
- Let R be created by adding four attribute scores together for each pair of restaurants:
  - Proximity: closer restaurants score higher
  - Category: restaurants in the same category score higher
  - Cross-references: the number of users who have liked both restaurants
  - Likes: the number of likes the restaurant has (vector)
- Each attribute receives its own matrix, which is scaled before addition.

$$\boldsymbol{R_{ij}} = \left[\mathbf{log}\left(\frac{\mathbf{1}}{\boldsymbol{dist(i,j)}}\right)\right]_{\boldsymbol{rm}} + [\boldsymbol{cat(i,j)}]_{\boldsymbol{rm}} + [\boldsymbol{ref(i,j)}]_{\boldsymbol{rm}} + [\boldsymbol{likes(j)}]_{\boldsymbol{rm}} \qquad \mathbf{(1)}$$

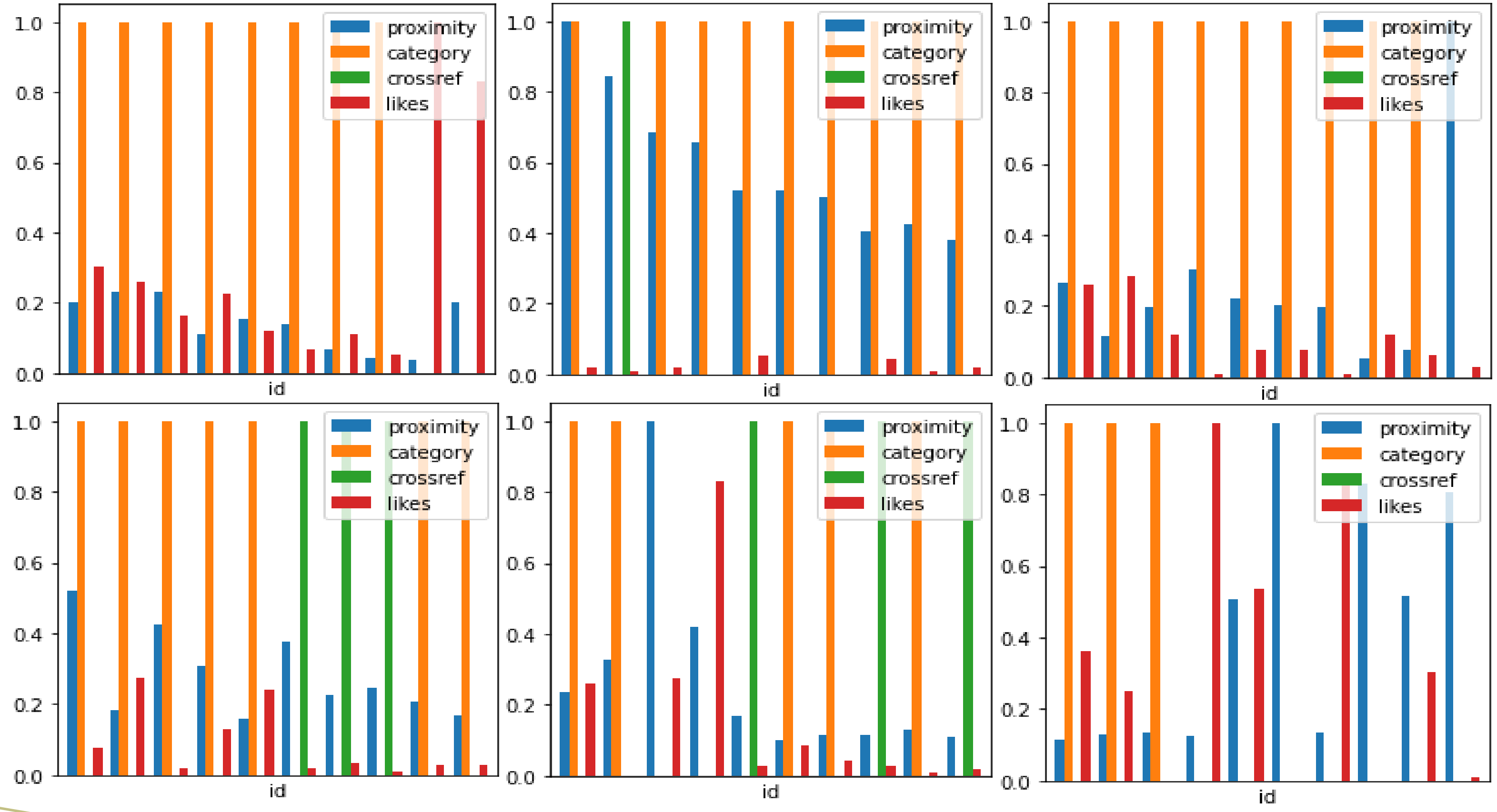
SCALING

- Before attribute matrices can be added, they must be scaled to same range
- Four different scaling combinations tested
- Row-wise min-max scaling selected
  - Min-max scaling guarantees that values will fall between 0 and 1, and that shape of distribution is preserved
  - Scaling of entire row ensures that each row ranges from 0 to 1, as opposed to a subset

Summary of Scaling Methods Examined		Scaling Type	
		Min-Max	Standardization
Subarray Selection	Row-wise	Selected	Rejected
	Triangular	Rejected	Rejected

RESULTS

- Each factor shown to influence recommendation matrix
- Across 6 trial restaurants, restaurants that scored highly in one or more of each attribute are well represented



CONCLUSION

- Solution is fast
  - Pre-computed recommendation matrix ensures instantaneous lookups
- Solution is scalable
  - Matrix operations can be applied to whatever sample size of restaurants provided
- Solution provides logical results
  - Each attribute has an equal opportunity to influence the final recommendation matrix

THANK YOU FOR REVIEWING!