**Introduction/Business Problem**

A patron of classical arts wants to open a theater near Boston City Hall. Of course, as this will be a new theater, the patron needs to attract customers to develop a customer base. To do so, he contacts a marketing agency to find out which places in Boston would be best to advertise the new theater. These places would constitute an optimal set of venues for both physical and virtual advertising. To carry out the project, the marketing agency will use the Foursquare API to collect information about venues in Boston and construct a system to rank which venues would be best for advertising the new theater. Foursquare is a data and intelligence company that provides location data for venues around the world. For example, one could use Foursquare as a simple search engine or for a more detailed description of venues. Developers may use the Foursquare API to freely gather limited location data or have access to more location data for a fee. Unfortunately, due to lack of funding, the data gathered from the Foursquare API will only be the free data accessible to developers. The marketing agency hopes that this project would be able to be further improved and models further generalized for a wider audience of advertisers.

**Data**

The marketing agency will come up with a ranking system to determine the best venues in Boston for advertising the theater near Boston City Hall. The data to be used in this ranking system will come exclusively from the Foursquare API.

First, the marketing agency will gather data for the top 100 venues around Boston City Hall (using latitude and longitude geographic coordinates) based on the Foursquare API “explore” query. The “explore” endpoint shows the top venues around a location ranked by Foursquare. Location data for these venues are then available. Boston City Hall has geographic coordinates (42.360100, -71.058900) for latitude and longitude, respectively. An example of the available location data results shown by this “explore” endpoint for Boston City Hall is shown in Table 1 for the top 5 venues from this search.

**Table 1**. The top 5 venues around Boston City Hall according to the Foursquare API explore endpoint. The name, category (type of venue), and latitude and longitude location for each venue are given. In addition, the distance from Boston City Hall in meters for each venue are shown.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Category | Latitude | Longitude | Distance (meters) |
| Boston Athenaeum | Library | 42.35748 | -71.0618 | 378 |
| North End Park | Park | 42.36249 | -71.0565 | 332 |
| Sam LaGrassa's | Sandwich Place | 42.35687 | -71.06 | 369 |
| Faneuil Hall Marketplace | Historic Site | 42.35998 | -71.0564 | 205 |
| Boston Public Market | Market | 42.36195 | -71.0575 | 237 |

After using the “explore” endpoint to find the top 100 venues around Boston City Hall, the marketing agency will use a methodology to determine the most appropriate venues for advertising. This methodology will be applied to the top 100 venues around Boston. The methodology is based on the idea that venues from which people then went to theaters would be good sites for theater advertising. In other words, if people went to a theater after visiting a particular venue, then that venue is a good place to advertise for a theater. This is a reasonable assumption since people who go to theaters are more likely to go to a theater again than people who don’t go to theaters. With this observation, a filtering process will be used based on the “nextvenues” endpoint from the Foursquare API. This endpoint shows the top 5 venues people visited from a given venue. Table 2 shows a sample of the “nextvenues” results for the top 5 venues from the “explore” endpoint for Boston City Hall shown in Table 1. The data that will be used from the “nextvenues” endpoint will be the category for each of the 5 next venues.

Table 2. The 5 next venues and their category that people visited from the venues listed in Table 1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Next Venue 1/  Category | Next Venue 2/  Category | Next Venue 3/  Category | Next Venue 4/  Category | Next Venue 5/  Category |
| Boston Athenaeum | Boston Common/  Park | Scollay Square/  American Restaurant | Emmet's Irish Pub/  Irish Pub | Faneuil Hall Marketplace/  Historic Site | 21st Amendment/  Pub |
| North End Park | Mike's Pastry/  Pastry Shop | Paul Revere House/  Historic Site | Quincy Market/  Historic Site | Faneuil Hall Marketplace/  Historic Site | Neptune Oyster/  Seafood Restaurant |
| Sam LaGrassa's | Boston Common/  Park | Quincy Market/  Historic Site | Faneuil Hall Marketplace/  Historic Site | Thinking Cup/  Café | Samuel Adams Brewery/  Brewery |
| Faneuil Hall Marketplace | Quincy Market/  Historic Site | Cheers/Bar | Mike's Pastry/  Pastry Shop | Newbury Comics/  Record Shop | Paul Revere House/  Historic Site |
| Boston Public Market | Quincy Market/  Historic Site | Faneuil Hall Marketplace/  Historic Site | George Howell Coffee/Coffee Shop | Mike's Pastry/Pastry Shop | Union Oyster House/  Seafood Restaurant |

In addition to the “nextvenues” category data for the top 100 venues near Boston City Hall, specific attributes about each of these venues will be used: “rating”, “likes”, and “distance” from Boston City Hall. The “rating” attribute is the average rating given to a particular venue by Foursquare users, the “like” attribute is the total number of Foursquare users that liked a particular venue, and the “distance” attribute in this case is just the distance of a particular venue from Boston City Hall in meters. Thus, the relevant location data to be used in the ranking system are given below.

**Foursquare Location Data to be Used**:

* Top 100 venues from “explore” endpoint for Boston City Hall
* “Rating”, “likes”, and “distance” (from Boston City Hall) attributes for each of these top 100 venues.
* The “category” of each venue in the “nextvenues” endpoint for each of these top 100 venues.

**Methodology**

Now, each venue has a uniquely named “category” attribute that describes the type of venue. Foursquare gives a list of all possible categories. There are 10 main categories: “Arts & Entertainment”, “College & University”, “Event”, “Food”, “Nightlife Spot”, “Outdoors & Recreation”, “Professional & Other Places”, “Residence”, “Shop & Service”, “Travel & Transport”. For each main category, there is a tree of subcategories. These trees are main category trees. Each node in these trees corresponds to a possible venue category and may have a parent node and children nodes. Therefore, each possible category corresponds to a specific node in the collection of main category trees. The unique path for each category in its corresponding main category tree can be found from following the parent nodes starting at the node for the particular category. A list of these parent nodes can be constructed to give the path for the category. For example, the “Theater” category has “Performing Arts Venue” as its parent node, which has “Arts & Entertainment” as its parent node (this node is also a main category so the path for the “Theater” starts at “Arts & Entertainment”). Therefore, the path for the “Theater” category can be written as [“Arts & Entertainment”, “Performing Arts Venue”, “Theater”].

|  |
| --- |
| **Theater Path**: [“Arts & Entertainment”, “Performing Arts Venue”, “Theater”] |
| **Dance Studio Path**: [“Arts & Entertainment”, “Performing Arts Venue”, “Dance Studio”] |

Given a path list, each category in the path has a corresponding *depth* which is simply its position within the path. For example, for the “Theater” path above, “Arts & Entertainment” has a depth of 1 since it is the first category in this path. Similarly, “Performing Arts Venue” has a depth of 2 and “Theater” has a depth of 3. Note that these depth values are actually attributes for each category since every path that contains these categories will have these categories at the same depths described above. Therefore, the depth of a category is unique.

Given two categories, we would like to know how related they are using their unique paths. We can compare these paths by determining if they have any nodes in common. The common node with the largest depth determines the similarity of the two categories. We define the similarity index to be the depth of the maximum depth common node for two categories (determined by their paths). If there are no common nodes for the two paths, then the similarity index is 0. As an example, the path for the “Dance Studio” category is [“Arts & Entertainment”, “Performing Arts Venue”, “Dance Studio”]. Comparing this path to the “Theater” category path, we see that the two categories have a similarity index of 2 through the “Performing Arts Venue” of depth 2. On the other hand, if we compare the “Theater” category path to the “American Restaurant” path [“Food”, “American Restaurant”], we see that the “American Restaurant” and “Theater” categories have a similarity index of 0. Naturally, the similarity index therefore is a measure of how similar two categories are.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Depth 1 | Depth 2 | Depth 3 | Similarity Index |
| Theater | “Arts & Entertainment” | “Performing Arts Venue” | “Theater” | 2 |
| Dance Studio | “Arts & Entertainment” | “Performing Arts Venue” | “Dance Studio” |

For each venue in the “nextvenues” list, we calculate the similarity index of their respective category to the “Theater” category. The similarity indices calculated for the “nextvenues” endpoint allow us to measure the likelihood a “Theater” category would be visited. For a given venue, if the venues in the “nextvenues” list have similarity indices close to 3 for the “Theater” category, then we would expect the given venue to be a good candidate for advertising as the venues visited from the given venue are similar to “Theater” venues. Two quantities based on the similarity indices for “nextvenues” searches will be used. First, the maximum similarity index of the “nextvenues” list of a given venue will be used. This quantity will be called the “max next venue similarity index”. Second, the average of the similarity indices of the “nextvenues” list of a given venue will also be used. This quantity will be called the “average next venue similarity index”. Any missing “nextvenues” entries will be ignored for the max next venue similarity index and the average next venue similarity index.

The classification will break down venues based on the max next venue similarity index and average next venue similarity index. The max next venue similarity index is given the highest priority and the average next venue similarity index is given the next highest priority. In other words, venues will be sorted first based on the max next venue similarity index. Then, each venue with the same max next venue similarity index will be sorted based on the average similarity index. The result of this sorting procedure is equivalent to the sorting done on a Pandas DataFrame by pandas.DataFrame.sort\_valules(by=[“Max Next Venue Similarity Index”, “Average Next Venue Similarity Index”]).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Name | Next Venue 1 Similarity Index | Next Venue 2 Similarity Index | Next Venue 3 Similarity Index | Next Venue 4 Similarity Index | Next Venue 5 Similarity Index | Max Similarity Index | Next Venues Average Similarity Index |
| Boston Athenaeum | 0 | 0 | 0 | 1 | 0 | 1 | 0.2 |
| North End Park | 0 | 1 | 1 | 1 | 0 | 1 | 0.6 |
| Sam LaGrassa's | 0 | 1 | 1 | 0 | 0 | 1 | 0.4 |
| Faneuil Hall Marketplace | 1 | 0 | 0 | 0 | 1 | 1 | 0.4 |
| Boston Public Market | 1 | 1 | 0 | 0 | 0 | 1 | 0.4 |

After this sorting procedure is applied, further sorting may be needed since some venues may have the same max next venue similarity index and average next venue similarity index. Since the “nextvenues” data have already been used for this sorting procedure, we use one last quantity for the final sorting. This quantity should capture attributes related to the popularity and quality of a venue. The Foursquare API venue attributes to be included in the sorting model are the “Rating”, “Likes”, and “Distance” from Boston City Hall. These venue attributes require Premium API calls and can be found through a “details” endpoint call. We want to find those venues with a large rating, large likes, and small distance since we would like to advertise to venues close to the Boston City Hall. To this end, we will calculate an unweighted Minkowski length based on these attributes. First, we normalize these attributes by their max values within each “Max Next Venue Similarity Index” equivalence class. For the distance, we use 1 minus the normalized distance in calculating the Minkowski length. Then, we calculate the unweighted Minkowski length according to

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Rating | Likes | Distance | Minkowski Length |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Since Each category lies in a particular tree of other sumarks a node in a tree of subcategories. So, if a venue with category “Theater” appears in the “nextvenues” response for a given venue, the given venue is a good candidate for theater advertising. Venues that are good candidates for theater advertising based on this algorithm will be termed “theater-prone” and all other venues will be described as not “theater-prone”. Therefore, the model will produce a binary classification of “theater-prone” and not “theater-prone” venues applied to the top 100 venues around Boston.

In addition to the “theater-prone” algorithm detailed above, the model will use a specific set of features for each of the top 100 venues around Boston. The features should reflect possible correlations between attributes of a venue and being “theater-prone” while also having a preference for more popular venues since these venues should generate a larger audience for the theater. The venue attributes to be included in the model are the category, rating, likes, and distance from Boston. Naturally, it is expected that certain venue categories would be more likely to be “theater-prone” than others. For example, a restaurant might be “theater-prone” as an entertaining visit to a theater is often prefaced by a meal. On the other hand, we generally wouldn’t expect automobile shops to be “theater-prone”, for example. As this marketing agency is using the free data from the Foursquare API, the “checkins” data for the total number of visitors at a venue in a given period of time are inaccessible. Hence, to capture the popularity of a venue, the rating, likes, and distance from Boston attributes of a venue will be used as features.

**Approach**:

Find venues in Boston that have theaters as the next venue visited. In other words, find all those venues from which people visited a theater. Call these venues “theater-prone” venues. Separate all venues in Boston based on whether or not they are “theater-prone”. We will then have a binary classification of “theater-prone” and not “theater-prone” venues.

**Venue classification**:

* 1 for “theater-prone”
* 0 for not “theater-prone”

**Extract the following features for venues**:

* Category, Rating, Distance from city center, and Number of Likes.

Take relevant data of Boston venues (venue classifications and features) and analyze data based on SVM or logistic regression.

For predicting classification of venues as “theater-prone” or not “theater-prone”, a logistic regression analysis will be used. The venues with the highest probability will be chosen as candidates for advertising.

To create the logistic regression model, the top 100 venues around Boston will be cross-validated. The output probabilities of the logistic regression will be averaged over all the cross-validation bins. A grid search of the best possible parameters in the logistic regression will also be used.

Fit data with different testing and sampling data subsets using logistic regression.