Building a Better Live Music Economy

Charles J. Anderson

January 8th, 2021

**1. Introduction**

**1.1 Background**

The music business has completely changed in the last 20 years. While many look at the rise of streaming as a leading factor in this, it is only a symptom of a larger shift towards live performances being the main revenue streams for musicians. As noted by Amy Wang for Rolling Stone, “Live events are quickly shaping up to be the most lucrative space for musicians in the digital-music era, and for good reason: As listeners become inundated with cheap access to music provided by streaming services, dedicated music fans crave more intimate experiences with their favorite artists.” Whereas artists used to tour in promotion of their records, records are now released as advertisements for tours. Much of the planning and promotional effort surrounding music focuses on how to ensure tours are booked appropriately. A large part of this comes down to selecting the proper venue.

**1.2 Problem**

My capstone project will help touring and local artists, signed and independent, to book shows in the Austin, Texas area. This project aims to create a recommendation model for a given musical act based on specific selected features of the act in conjunction with identified and relevant venue factors.

**1.3 Interest**

The stakeholders will be the record labels, managers, and bands that are booking the shows. Given the state of the world in 2020, with the COVID-19 pandemic all but shutting down live performances, labels are looking for more security in booking their shows so that they get a lot more money and do not have to worry about the performance as much. They want to control the variables and avoid booking bad shows. There are several features across paid-for platforms that help labels know what cities to tour in (Bandcamp pro allows you to map the streams and purchases of users to better find your audience, as does spotify). Once they find those cities, the process becomes a little bit more difficult to navigate, which is where this project can come in handy. I am limiting the scope of the project to the Austin area.

**2. Data**

**1.1 Databases and sources**

For this project, I used two databases. First is the Foursquare database, which holds information about venues of all kinds, including relevanat metadata including a venue’s name, user rating and location. I determined that I would also need relevant information that my Foursquare account does not provide, including venue capacity and and associated musical genres. For these key features, I used “Indie on the Move” (IOTM), a database that is manually regulated and updated to reflect more current venue information as well as additional, music venue-specific insights.

**1.2 Data acquisition**

My first step was to download all of the music venues I could get from the Foursquare database. Because of rate limits, I set a limit of 100 venues. The resulting JSON file was converted into a pandas dataframe, and stored locally on my machine. I selected the name, ID, address, latitude, longitude, and rating of a venue as my features of interest. Unfortunately, I was not able to make API calls to IOTM with a free account. Instead, I manually entererd capacity and genre info for the venues that I could find in IOTM’s database using Linux’s LibreOffice Calc. Once I completed this step, I saved the resulting csv file and read it back into a new pandas dataframe in my notebook.

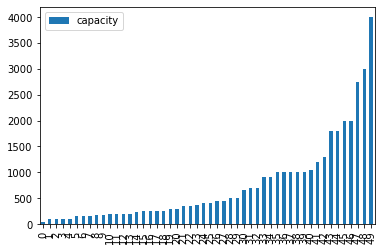
**1.3 Data cleaning**

I was not able to find information regarding capacity and genre for all of the venues that were pulled from Foursquare. Additionally, some of these venues did not show up in IOTM’s database at all. A few cursory google searches showed many of these unlisted venues to be permanently closed. I decided to drop all of the venues from my dataframe with no genre or capacity information so as to avoid accidentally making any recommendations for closed venues.

**3. Methodology**

**3.1 Venue sizes**

Because venues at different sizes operate quite differently in the live music economy, I decided to split the venues into three size bins, “small,” “medium,” and “large.” Small venues typically include bars and cafes where the venue does not exist solely as a music venue. Medium venues are typically local staples that host small to medium sized touring acts and local bands with a decent ability to draw a crowd. Large venues are typically reserved for major label touring acts and special events. In order to sort venues into these categories, I sorted the existed dataframe by capacity, and created a table mapping capacity against table index:



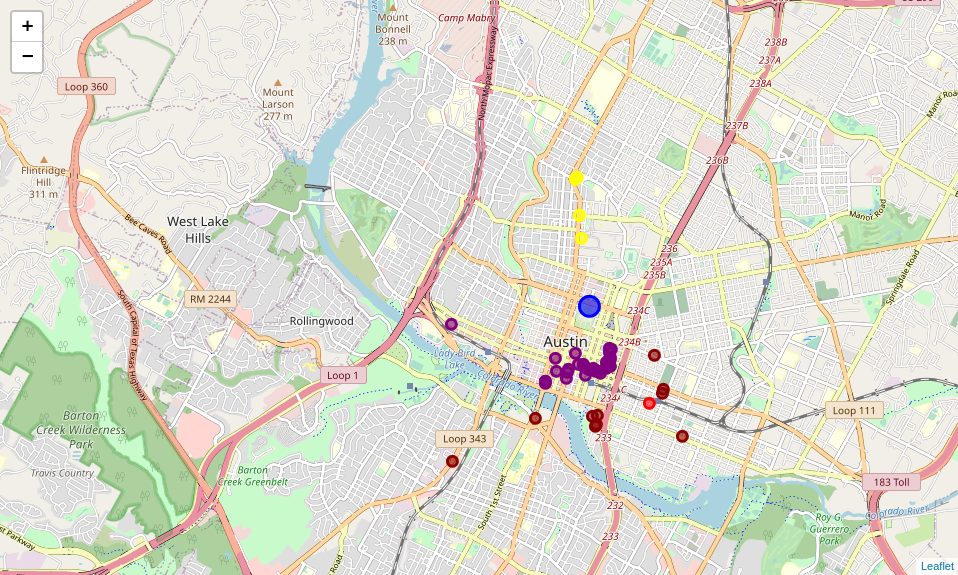
The shape from index 0 to index 29 looked like a good set for the “small” venue size. This includes all venues with a capacity ranging from 30 to 500. The “medium” category included venues from index 30 to index 39, indicating venues with a capacity between 500 and 1000 (inclusive). As someone that has performed shows at many of these venues, I found the bins to be of relatively good fit. I then added a column to the dataframe to correspond to each venue’s size.

**3.2 Proximity score and small venues**

As defined above, small venues typically include bars and cafes, where live music is a draw for the business, but perhaps not the main event. As such, there may not be much promotional effort put into the night by the venue. Artists at this level typically have a very limited promotional budget, if any, meaning that they must rely on random people, or “walk-ins,” to make up their crowd. I hypthesized that the likelihood of walk-ins increases for venues located in closer proximity to other venues, bars, and restaurants. For the purposes of this project, I used the venues in my existing dataframe to generate a “proximity score,” which is the average distance of the 3 closest neighboring venues. A lower proximity score may indicate a higher likelihood of walk-ins, and was used as the dominant feature for recommending a small sized venue. In order to calculate the distance between venues, for which I had the geospatial coordinates, I used the haversine distance formula. Python has a haversine library built in, so I installed it and used a zipped list of each venue’s latitude and longitude to create an array of lists, for which each element in a list reflected the distance from that venue to each other venue in the dataframe. I generated another array of lists, giving the indices of the three closest venues for each venue in the dataframe. With these lists, I was able to implement a simple loop to create yet another list of the proximity score corresponding to each venue in the dataframe, in miles. I then added the proximity score as a new column to the dataframe. After doing this, I created a separate dataframe containing only the small venues for later data handling.

**3.3 Mapping proximity score and location**

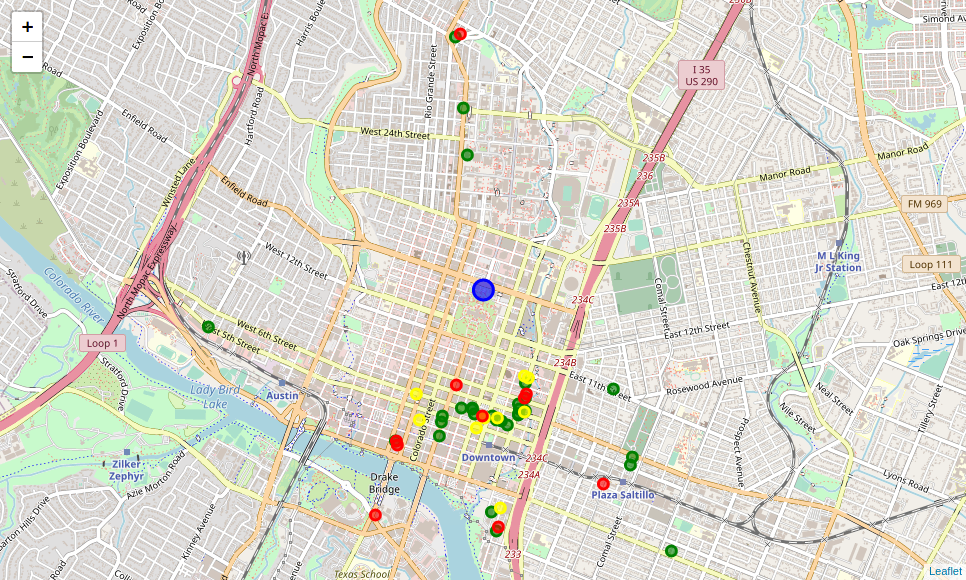
Once I had the proximity score for all venues added to the dataframe, I used the folium library to create a map of Austin with all of the venues in the dataframe marked and color coded by proximity score. The blue marker indicates the capitol building of Austin, a commonly used reference point for the center of the city.

****

A few clusters appear right away. The purple markers indicate venues with the best proximity score, followed by dark red, light red, and yellow. This means that small venues in the areas of 6th Street and Red River Street are easier to recommend for getting walk-ins. Sure enough, as any weekend on the town will show, these areas are very busy with bar-hopping tourists and locals. Other notable clusters include the venues on Rainey Street, just north of the Colorado River and west of I-35, and the venues on East 7th street.

**3.4 Mapping venue size and location**

I created a map using folium to show the venues as binned by size. Green venues are “small,” yellow venues are “medium,” and red venues are “large.”



Comparing the map above with the map from section 3.3, we can see that many of the venues in the Red River and 6th Street areas are “small” venues with high proximity scores. This suggests a correlation between the two, potentially indicating proximity as a key feature of small venues’ business models. Furthermore, large venues appear to have slightly lower proximity scores on average, typically falling into the “dark red” proximity category, just below “purple.”

**3.5 Medium sized venues and ratings**

The following comments are all assumptions that I am making for my recommendation engine based solely on my own experience in the music industry. Medium sized venues are more likely to be dedicated music venues than small sized venues are. As such, more promotional effort goes into each show that is booked there, and music takes on a much larger portion of a venue’s income. Venues are more likely to have door charges and sell tickets online, making walk-ins a much less significant portion of the customer base. As such, when choosing a venue to book at, an artist may find other features of a venue to be more relevant than the previously calculated proximity score. I created a dataframe for the medium sized venues, ranked by user rating, for later data handling.

**3.6 Large size venues and capacity**

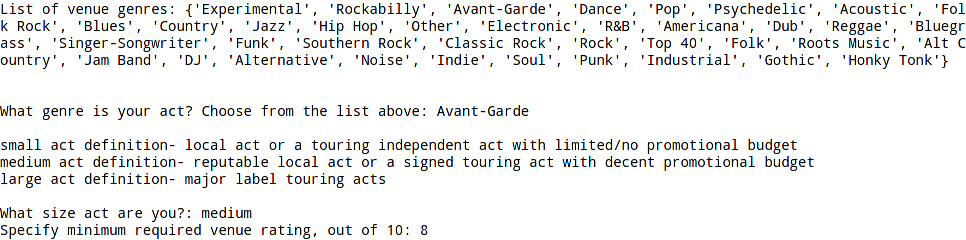
The following comments are, again, assumptions that I am making based on my own experience in the music industry. Large size venues, such as the ones in this dataframe, typically work with larger touring acts. Often times, these artists are represented by booking agencies who are looking at the capacity of a venue as one of its most relevant features. I created a dataframe for the large sized venues, ranked by capacity, for later data handling.

**3.7 Recommendation engine**

My ultimate goal for this project was to create a basic recommendation engine for musical artists to use in booking shows in the Austin, Texas area. I identified the popularity (“size”) and genre of an act to be key features for recommendation. I wanted to give the artists the option of setting a rating threshold, so that they only receive recommendations for venues that meet their own personal standards. The recommendation engine lists all of the genres in the dataframe, as well as the option to list one’s genre as “other.” The recommender uses this input to determine what sized venues to recommend, and based on that, recommends venues that match the band’s genre as ranked by that size bracket’s selected feature. Small venues are recommended by proximity score, medium venues by rating, and large venues by capacity.

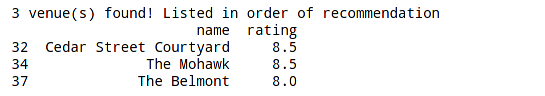
**4. Results**

An example prompt of the recommendation engine looks as follows:



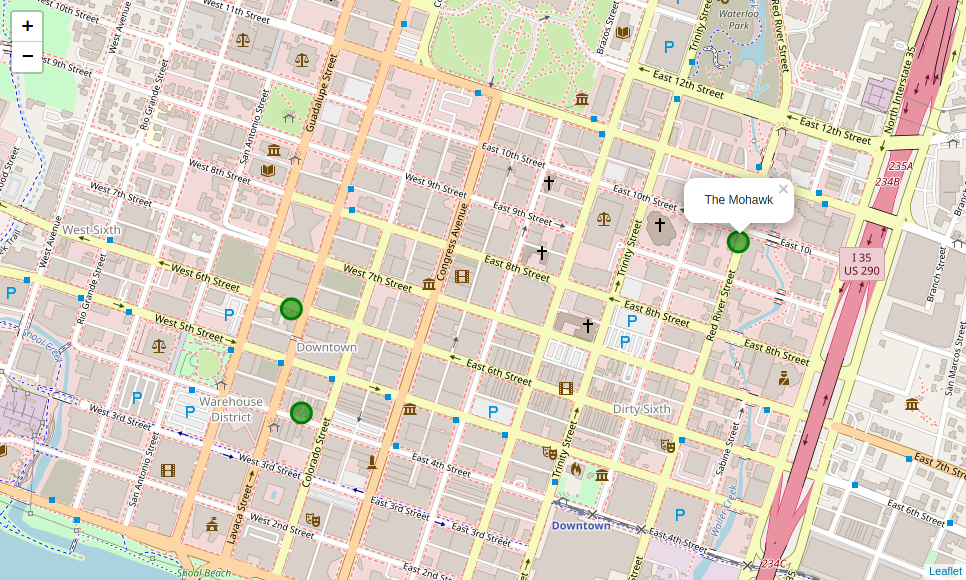
The program lists all of the genres for an artist to choose from before prompting them to enter their genre. Then, it defintes the small, medium, and large artist categories before prompting the artist to enter the size of their act. Finally, the engine allows the artist to specify a desired minimum venue rating. In this example, I said that I was an Avant-Garde act of medium size, and that I was only interested in venues with a user rating of 8/10 or higher.

The result for my example query:



The recommendation engine found and returned three venues that match my criteria! Not bad for being a choosey Avant-Garde act.

Furthermore, the engine shows these venues on a map of Austin:



As someone who has played at all of these venues, I find the results to be generally acceptable and accurate given various inputs to the engine.

**5. Discussion**

**5.1 Further research and development**

The final deliverable of this project, a music venue recommendation engine for the city of Austin, Texas, represents a rudimentary framework. While I am pleased with the end results, there are many elements of the process that would require deeper research and analysis before taking such an engine to market. First, the final list of 47 venues is hardly exhaustive, even for just the city of Austin. If I were able to make more API calls for venue data, mesures such as the proximity score and venue size would also have to be recalculated to offer meaningful variance in the data, and to achieve plots similar to what I found in my visualizations from sections 3.3 and 3.4. Additionally, calculating the proximity score based solely on neighboring music venues is restrictive at best. I would like to calculate this value based on many types of venues, including bars and restaurants that are not counted as music venues. If I had access to the Foursquare database without fear of rate limiting, I would have done so. Providing the artist with booking contact information for each venue would be beneficial as well.

**5.2 Scaling the service**

While Foursquare has international venue data, IOTM only offers venue data for the United States. If I wanted to scale this recommendation engine to an international scale, I would have to find foreign venue databases for capacity and genre information. Some of the earlier steps in the notebook would also be automated for user input, including reading in the the city of interest from the user. I would want to build the model so that instead of calling Foursquare’s API every time a user wanted to make a query, they would interface with the dataframe that I had cleaned, created and stored remotely. There are legal ramifications for operating off of databases such as Foursquare that I would need to look into. Finding a way to integrate “temporary” venues, such as festivals, would be a unique and interesting challenge to tackle as well.

**5.3 Integration**

This recommendation engine could be used as a module in a full-feature booking marketplace application. This application could handle the venue search, booking, and payment aspects of any given musical event. While this would require bookers, venues, and artists to sign on with the service, it could potentially create a democratized and direct marketplace that removes many middle-men and simplifies the long imperfect process of tour booking.

**6. Conclusion**

In this project, I analyzed music venues in the Austin, Texas area, and identified key features by which to categorize and rate them. These included location, proximity to other venues, capacity, and musical genre. I built a basic recommendation engine to help artists find the best fit venues for their acts when booking shows in the area. I hope to conduct further research to make the model more robust and helpful. I plan to meet with bookers and artists to see what features they would like included in such a platform. There is room in the market for a centralized, democratized booking platform: I hope this project can serve as the beginning of its development.