Lisa Watkins
netID: watkinsl
IS 452AO
2 Credit Hour
Week 17
12/21/17
Final Project Narrative

The American Music Awards (AMAs) is an awards show held annually. Unlike other award shows, the winners of the AMAs are voted for by fans. In order to be nominated for an AMA, an artist's excellence is determined by a criterion made up of various factors such as: album and digital song sales and, what this concept focuses on, social activity and touring ("About | American Music Awards," 2017). Nowadays, artists bear a lot of responsibility to interact with their fans on multiple platforms. Gone are the days of simply releasing an album and going on tour to interact with fans. Today, fans expect to see their favorite artists on a regular basis and this is achieved through the use of social media.

As far as social media platforms go, Twitter and Instagram are considered two of the most popular platforms fans can connect directly with their favorite artists. Both of these platforms are conducive to candid interactions and revealing behind-the-scenes images of their daily life. If the AMAs were interested in the social activity of the artists they are considering to nominate for one of their awards, Twitter and Instagram would be a great place to start.

Even though fans have the ability to interact with their favorite artists at their fingertips, there is nothing like seeing your favorite act live. To satisfy the touring aspect of the AMAs criterion, I look to Ticketmaster. As an avid music fan myself, I have relied on Ticketmaster to purchase all of my concert tickets. Ticketmaster is also probably the most well-known legitimate ticketing platform. Due to its stake in providing a platform that supports touring artists, Ticketmaster would be a great resource to retrieve touring information.

To aid in the AMAs process of considering artists for nominations, I propose a Python-oriented program that would automatically scrape information from websites that would provide insight on an artist's social and touring activity. In this proof of concept, I will be looking at this year's "Artist of the Year" award nominees and attempt to gather information on them by scraping web pages provided by Ticketmaster, Instagram, and Twitter. This would provide a good baseline as to whether this program is a viable solution for a potentially extensive process of determining the excellence of each artist's social and touring activity.

My initial research on web scraping–a concept not covered in class–led me to a Python library called Beautiful Soup. Beautiful Soup provides "screen-scraping" methods that allows the program developer to parse a web document with less code, which in turn leads to shorter turnaround times for development (Richardson, 2017). This was appealing since web scraping is a new concept for me. To test the Beautiful Soup library, I attempted to extract the Ticketmaster rating information of a single artist. In every example I've seen utilizing Beautiful Soup, the websites are generally simple in their construct. Dealing with a more robust website like Ticketmaster was bound to encounter some issues. As a result, the program would crash every time.

In my investigation of alternatives to scraping Ticketmaster information, I looked to their application programming interface or API. However, Ticketmaster's API was very limited in what kind of information was provided and it wasn't what I was looking for specifically. In the end, there was no choice other than saving out each web page individually and refer to them locally on my computer. I ended up doing this with every web page I intended to scrape including Instagram and Twitter. As you can imagine, this process would be extensive if this project were scaled up to encompass each artist being considered as nominees. This approach brings along additional implications, which will be addressed later. However, for the purpose of this project and the five artists that I am concentrating on, I moved forward.

Once I had each web page for every artist saved, I started to think about my end product–a CSV file. Before I did any scraping, I wanted to determine what my CSV file will include and how it will be organized. I decided on a layout of six rows and seven columns. Five of the rows will be dedicated to each artist with a sixth row containing headers for each of the seven columns. To focus on particular pieces of information, there is a column for the artist's name, Ticketmaster rating, number of Ticketmaster reviews, number of Instagram posts, number of Instagram followers, number of tweets, and number of Twitter followers. In designing my CSV file prior to any development, I was able to scrape each web page intentionally.

The process of scraping each web page was definitely a challenge. Ticketmaster's terms and conditions explicitly state that their website shouldn't be mined for data, which is probably why my program kept crashing initially when I would try to scrape the web page live. This is an important consideration to keep in mind if the AMAs want to utilize Ticketmaster's information as a legitimate source for determining an artist's excellence in touring. I didn't look in to the terms and conditions of Instagram and Twitter, but that will have to be done if this program is scaled up and used for official purposes.

In pursuit of developing a program that can run successfully, I dug through each web page's robust code and identified the pieces of information I needed to extract. The HTML structure and attributes weren't intuitive whatsoever, so it took a careful eye to recognize the pattern of each page's structure. The most difficult piece of information to extract was the number of Instagram followers. I had to revert to utilizing XPath because the information regarding Instagram followers was wrapped in a variable attribute. This particular attribute that distinguished the number of followers from other types of Instagram information reflected the actual number of followers an artist had. Through the use of XPath I was able to access and extract this information. This might be another consideration to keep in mind, whether or not to commit to Beautiful Soup or utilize XPath. In this example, I was successful with using both at the same time, but conflicts could potentially arise through this method.

Once I was able to extract the necessary information needed from each web page, a few tasks remained: cleaning up the output, convert datatypes, and organize the information in a way that will be written correctly to the CSV file. Utilizing a few string methods, I was able to clean up the data by removing commas while also converting the datatype from string to either an integer or float. Other pieces of information required extra assistance such as the number of tweets and Twitter followers. These numbers were abbreviated using a letter (i.e. 39M for 39,000,000 or 35.9K for 35,900). Therefore, I created a function that would evaluate these numbers and convert them to actual integer numbers. At first, I only had a solution for instances where millions were abbreviated, however, Ed Sheeran threw a curveball with his tens of thousands of tweets. He had more tweets than any other artist, which is saying something since he took a year off of social media. So, I had to also take this into account and create a solution to convert thousands represented as “K” into integers. Not all numbers that pass through the function will contain a letter, so a solution was necessary for those instances as well. This solution included a cleanup of the data and the conversion of the string value into an integer. I feel that crafting this function took a little creativity and I really enjoyed figuring it out.

Lastly, I had to figure out how to organize the information so that when it is written to the CSV output file, each piece of information will be placed in its appropriate row and column. I initially had all of the files within a single list, but thought that it might be better to separate the files by each artist. I fed these list of lists of filenames into a set of loops that will extract the necessary pieces of information according to the corresponding file path (i.e. Instagram information will only be extracted from files with a IG prefix). Once that was established, I needed to mimic the list of list idea to capture the results of the web scraping according to each artist. Within the loop there is a list named, “artist_data” that will capture all of the results from a set of artist’s files. Once every set of artist files is done looping through the program, “artist_data” will then be appended and nested into another list called “artist_rows.” As a result, I have a list of lists of results pertaining to each of the five artists specifically. Once I was able to create this structure of the extracted information, I was ready to write to my CSV file. This was achieved successfully as seen in the CSV file included in my GitHub repository and also when you run the program yourself.

This program does everything I set out to do. With the CSV output file there are endless amounts of additional operations you can do included creating data visualizations. However, looking at this program alone, there are some implications. Aside from issues previously stated, one big issue is not being able to parse these robust web pages in real time. The need to save them out as web files and then scrape them can be a potential issue since information on these platforms can change day-to-day. It would be more beneficial to scrape these pages live to get the most accurate count. There was also the issue of having a list of filenames for each artist. This list could get overly complicated the more artists are added to the program to be evaluated. So, this part would need to be revisited if this program was scaled up or performed for official uses.

I do believe that this concept should be considered for continued exploration amongst a small team of programmers. I believe that award shows such as the AMAs would find this beneficial as they aggregate this type of information and analyze it as part of their criterion for nominations. I hope that you see the value in this project as well and all of the potential opportunities that this data can unearth during a time where artists have to not only be strategic about their in-person presence, but also in their social media presence.

Sources:

(2017). “About | American Music Awards.” Dick Clark Productions, Inc. Retrieved from https://www.theamas.com/about/

Richardson, L. (2017). “Beautiful Soup.” Crummy. Retrieved from https://www.crummy.com/software/BeautifulSoup/