Program 7 Reflection

Group Work Statement

The work comprised in this submission was accomplished by the following people.

- Van Spezzapria
- Darius Lakas
- Matthew Van Antwerp
- Justin Schmid

Below each group member should write a few words about what they contributed to the project.

- Van contributed to problem 3, the csv, writing reflection page, error handling
- Darius contributed: Part 3, fixing dataset, writing reflection
- Matthew Van Antwerp contributed Helped vent data and do reflection
- Justin contributed…

Data Pre-processing

As part of your reflection document for this program, include the following. Please respond to each with a paragraph response.

1. If you removed or altered any entries in the dataset, please give a detailed description as to why, include the data that was removed. Discuss what implications might exist from removing or changing it and how you dealt with those implications in your project.

Many entries were altered due to having commas that interfered with processing the comma-separated values file. These commas were replaced by a vertical bar---this character was not in the dataset (and is a common substitution for csv files) so this alteration made sense to ease implementation details. No entries were removed.

2. Detail your chosen list of genres. Discuss how this list was agreed upon and what implications it might have on your recommendation system. What genres might be missing from your list? Why are they missing?

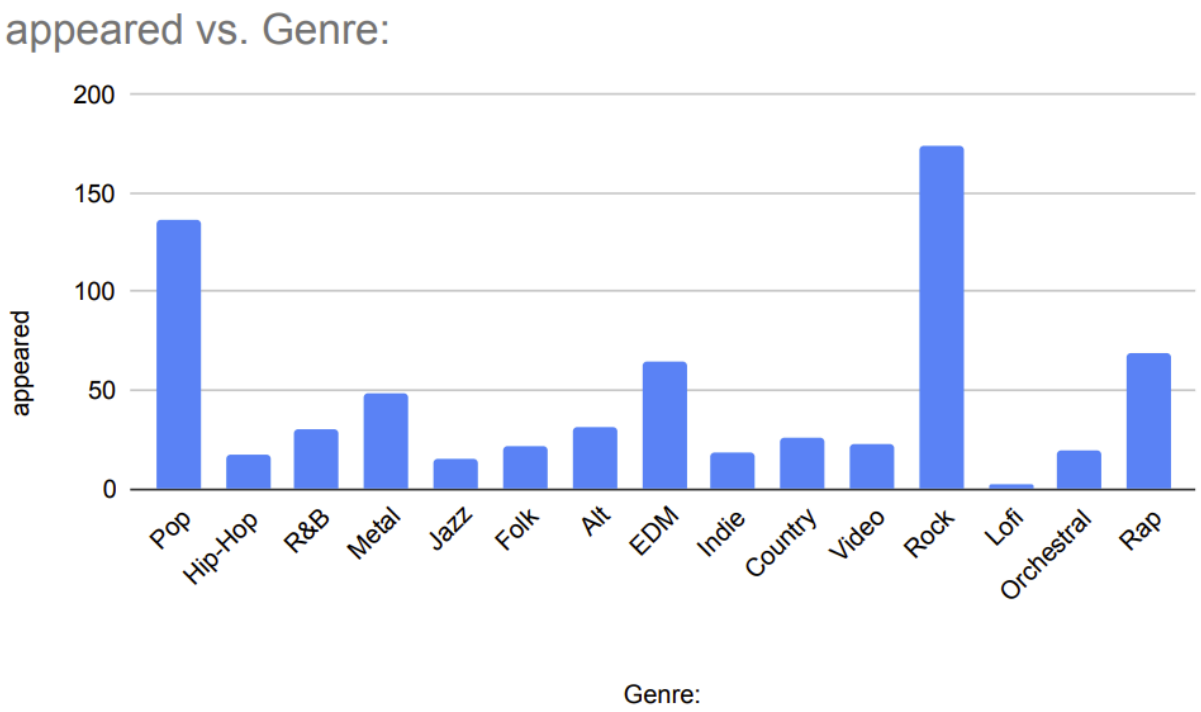
The chosen list of genres:

1) Pop
2) Rock
3) Hip-Hop
4) Rap
5) R&B
6) Metal
7) Jazz

8) Folk
9) Alt
10) EDM
11) Indie
12) Country
13) Video Soundtrack
14) Lofi
15) Orchestral
16) Christmas

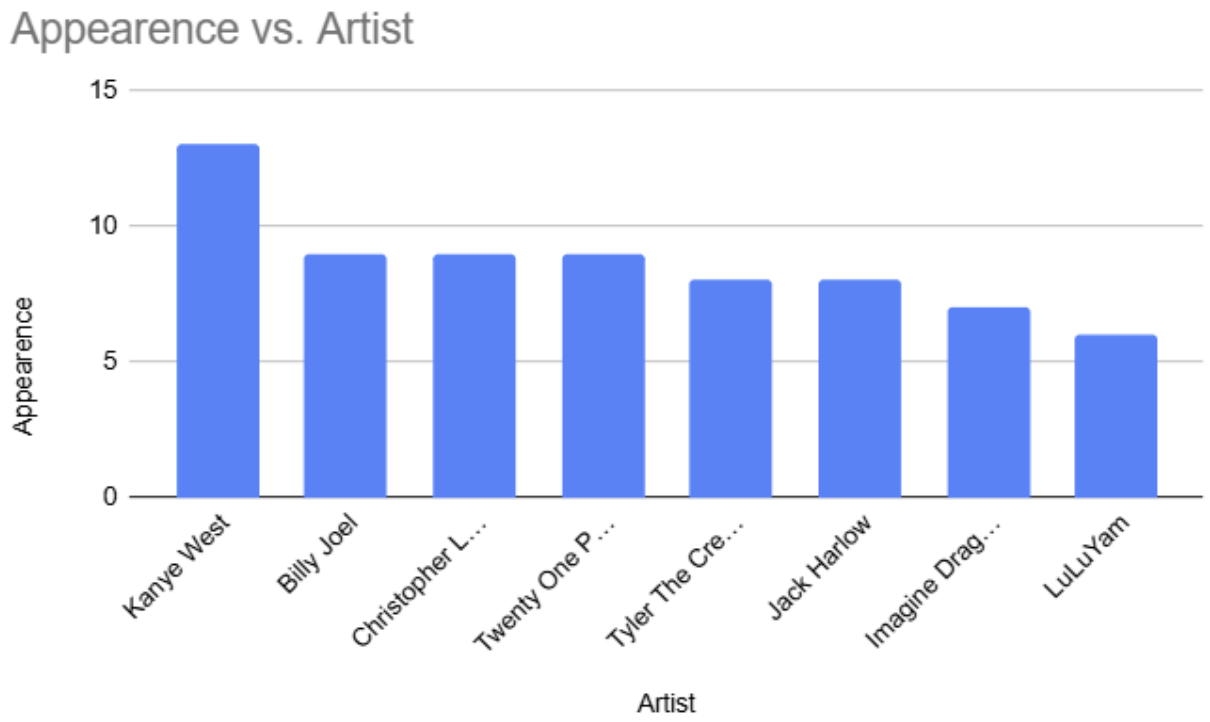
This list was agreed upon by small-group discussion and consensus. Several heuristics were identified and used, such as “pop trumps all,” “there is only pop and metal,” “shorter genres take priority" and “Mariah Carey’s All I Want For Christmas is a biblical grade memetic virus.” Of course, this excludes many genres such as dwarf metal, pirate metal, frog metal, Ned Flanders metal, doom metal, experimental industrial heavy metal, anti-christian war-pagan reindeer-slaughtering industrial death metal. These all fit under the metal genre. A similar approach was taken to other broad genres such as pop.

3. Include at least two tables or graphs that examine a particular aspect of the dataset. For example, a bar graph of included genres, table of popular artists, or list of the most played songs. You should explain your choice of graph.



This first chart shows the frequencies of genres in the data set to see what genres are most popular as you can see we really like Rock music in addition we really like Pop. We did take some creative liberties with what counts as pop and we took the approach of when in doubt it’s pop so things like K Pop. J Pop, Rock

Pop, Alt Pop, all of these were amalgamated into the frankenstein monster that is pop and it's pretty surprising to see Pop is actually not the most popular genre and it is in fact Rock.



I made this super cool graph that shows the most popular artists in our class and how frequently they were in our class's top 10 songs. To do this I wrote a sorting algorithm in google sheets that was quite the time I looked up a youtube tutorial and adapted the code to work on our data set. The results are phenomenal. Kanye is at the top as he should be followed by a 3 way tie between Billy Joel Christopher Larken and Twenty One Pilots. This all makes sense so far as kanye is one of the most decorated artists out there Billy Joel is older than dirt and twice as old as air so everybody knows him Christopher made the music for silk song and that game

4. What short-comings or flaws do you see in the way in which we collected data. If you had to collect data like this again, what would you change, and why.

I would have made more restrictions on what people are allowed to enter to reduce the amount of time it would take to vet the data. Some of the restrictions I would include would be only english characters and only characters that are on the ascii table so we can sort the inputs easily into strings. We would also have preferred an input file format that does not use commas as delimiters, and support for lists within fields—songs can have multiple artists.

Reflection Paper

Below write a paper that analyzes your project and its results. Your paper should cover the following topics (but is not limited to):

- How do the Euclidean and Pearson similarity methods compare to each other, which produces better results?
- An analysis of the playlists results. Are the tracks listed good matches for users?
- How might you improve the relevance of the generated playlist tracks?
- How might you modify the program to compare tracks across multiple fields?
- An analysis of the efficiency of your code. How might the speed, memory footprint, or accuracy be improved?
- What challenges did you face developing the program and how did you overcome them?

Our program implemented both the Euclidean and Pearson similarity methods to judge whether two users had similar music tastes based solely on how the genres of music compared between the two users. Both methods use different algorithms to get to the same goal, and we found that the Euclidean algorithm results seemed to make more sense, though this could be a result of the lack of data used for making a recommendation in the first place. When analyzing the results of the playlist created, the quality of the matches made leaves a lot to be desired unfortunately. Since the similarity scores only looked for if genres were the same, and we did a lot of work to condense the amount of genres down to about 15 total, there was a lot of wiggle room for songs that weren't exactly similar to the users original songs. For example, there are many different subgenres of "pop" that were all changed to pop in the original data file for the sake of simplicity, but not all songs labeled pop are actually the same sort of pop, just what we deemed to be close enough. If we had a much larger dataset with thousands upon thousands of songs, we may have been able to create a much better recommendation system that would allow for niche genre songs to actually have matches. Without a larger dataset, we still could've implemented a subgenre system which could have worked by acknowledging that, for example, hyperpop is a type of pop and scoring it against other pop genres or subgenres about half as heavily as it would be scored if it was scored against another hyperpop song. We could have also improved the accuracy by having the program look at more than just genre. One easy implementation for similarity would be to check if an author or album was the same as another song, which would lead to having a high similarity score regardless of if the genres were exactly the same. One inherent issue with our code is that none of the hashmaps are stored long term, meaning that a new (and likely duplicate) hashmap is made every time the code is run. We could improve the speed of our program, especially if a large data set was used, by finding a way to implement our hash as permanent storage on the user's disk. I believe the speed of our algorithms are quite efficient and get done extremely fast with our relatively small dataset. There could be some more work to be done if a larger dataset was going to be used, but it still works

nearly instantaneous and although some algorithms may not have the best time complexity in theory, but in practice everything works extremely fast with our provided dataset, so we believe our code is good enough for our purposes. The biggest challenge I found in developing this program was the collaboration aspect of it, as it was a bit challenging to communicate without being in the same space together, and after divvying up a lot of the work between each other, we found that a lot of the work we had to do would build off of something that another person had to do first, meaning we couldn't really all work on the project at once. This led to a lot of parts of this program not getting done until closer to the deadline than we otherwise would have liked. Overall, all of us in this group are happy with how our program turned out and are proud of what we were able to achieve working together.

Extra Credit

Below write a list of the extra credit that you attempted or state that you did not attempt any extra credit.

We did not attempt any extra credit.

- Extra credit 1
- Extra credit 2

If you attempted any extra credit, please create a new subheading and explain the required information for that extra credit. Extra credit will not be graded if not explained well enough for me to assess.