Data

The lyrical data we will be analysing has been sourced by using the Spotify and Genius APIs using code that we constructed (not from scratch, sources in appendix) to scrape song details and lyrics for any Spotify playlist. Each observation contains the song name, album, artist, release date, length, popularity (according to Spotify) and their lyrics, which are in the form of a single string, containing metacharacters and sometimes, non-alphanumeric characters.

The playlists we chose to scrape were the “Top Hits for [Year]” playlists, compiled by Spotify themselves (we assume with reference to historical charts). Unfortunately, lyrics weren’t found for each of these songs (and if no lyrics were found, they weren’t included in our data) but this process still left us with approximately 5500 songs to analyse spanning from the 1950s to now. Note that we used other popular playlists not compiled by Spotify as part of our data for the 1950s and 1960s as Spotify didn’t have “Top Hits for [Year]” playlists for these decades.

The choice of these playlists was in order to accurately reflect trends in popular music in western culture as trends in music that never gained traction is of little importance. We have also got some genre specific data from similar playlists compiled by Spotify (e.g. best of rap, party and rock genres) for potential further analysis of lyrical trends in these other popular genres. We chose not to add these to the overall dataset as the language used tends to be quite genre specific and could bias our results. Although it is worth noting that many of these songs would also be in our dataset of focus due to their overall popularity.

Experimental Design

The main way we plan to evaluate our classifier is by using accuracy scores. As we will likely start off our classifier as a binary classifier (which century as opposed to which decade), we can use the accuracy, error rate, precision, recall and F1-score to assess our model. However, as we add complexity to our classifier to the point where it is able to classify which decade, precision and recall will be defined for each individual class/decade. We can also use the per-class F1-score as an evaluation metric, despite the classifier becoming multi-class. Normally, the F1-score is contentious because it gives equal weigh to precision and recall but in our case, that’s acceptable as the cost of misclassifying a song from decade x as being from decade y has equal cost for every decade x and y.

In terms of evaluating our topic model, there is no sound mathematical evaluation similar to our metrics for classifiers. We do however, aim to manually assess the accuracy of our topic model by taking a random sample of 100 songs, determining the topic by inspection and comparing this to topic the model gave us, thus assessing the model’s accuracy. Beyond that, we will compare if the trends we found match up with those found in Christenson et al 2019.

Furthermore, we can compare any individual word trends to this study, by manually matching individual tokens to their likely topics and seeing if similar trends occur here too. Another evaluation we can conduct on our token trends is to compare the prevalence of these tokens in lyrics with their prevalence in Google trends, hence general language and interest. We somewhat expect some tokens to be extremely topical for a song’s given release date which should correspond with peaked interest in Google trends data. Notably, Google trends only has data dating back until 2004 so this evaluation can only be done with songs released since then.

Finally, looking at term frequencies for documents and the bag of words we can observe simple trends such as if repetition and simplicity is becoming more popular in lyrics as found in the study by Varnum et al 2021. If the maximum term frequency for each document (song) increases over time, this suggests more repetition while if the average word length for each document decreases over time, this suggests lyrics are becoming simpler.