Neer Jariwala
William Lucca
10 November, 2019

Objective

Our objective was to create a song lyric generator. The generator would take a bunch of song lyrics and use a procedural generation method to create a brand new song. Our goal with this experience was to create an entertaining and funny generator while also learning more about procedural generation methods. Initially, we hoped to achieve a fully functioning generator that was able to create songs with a chorus, pre-chorus, verses, etc. and separated by artist. We planned on using java and the n-gram procedural generation method.

End Result

Our final product was a Java application that allowed you to generate a random song with the click of a button. The program used n-grams and a database of 12,116 songs to generate from. The database was compiled using the JSoup library, the application was built using a JFrame, and n-grams were stored in a List of Lists. Compared to our initial hope, our generator is not capable of creating songs by artist and with a chorus, separate verses, etc. Instead, it chooses a random number of stanzas, lines per stanza, and words per line from a small range of possibilities, that still appears to have the structure of a song.

Approach

We chose to use the n-gram generator method as it allowed us to create more readable lyrics than if we used something like a Markov Chain, which would end up making a random jumble of words. While the n-gram is able to make somewhat readable lyrics, it is not perfect. There are times where the generator creates a totally random line that makes no sense at all. However, this is part of our generator's charm of making a totally wacky and funny to read song. Some of the things we would change include how our songs is formatted upon generation, change the place our database is built from, and improve our choice in generator. We could also test words against an English dictionary to prevent foreign language songs in the database from interfering. These changes would all be towards the goal of creating more readable lyrics that are more properly formatted to look like an actual song.

Things we learned

One of the biggest things we learned from this experience is how many options are available when creating a generator. While building our own generator, we discussed various

ideas and switched tracks multiple times. When deciding on an appropriate value for “n” for creating our n-grams, we noticed that low values would create incoherent output, but high values would echo songs it read while collecting input. However, even with all of these possibilities, it is still quite difficult to create a generator with all of the features you want. There are many program bugs and areas where you lack experience in that can show up when creating a procedural generator.

Community participation

We chose to participate in the community through the Discord server. Getting to see the various creations that were shared each day was quite interesting in that it allowed us to see even more ways a procedural generator could be used. One we thought was very creative was “No Destination” by BigBread, which generates a landscape in realtime framed as a train ride with no destination that the player can passively observe from inside. We were also able to follow along with the development process for each of these generators and see how each individual or team chose to tackle the problems that they encountered.