Executive Summary – Good Composers Borrow, Great One's Steal

We developed a model to analyze the similarity between songs by directly analyzing the audio. Using audio analysis techniques such as calculating log-mel spectrograms, audio augmentation, and transfer learning we established a way to compare the similarity between audio files for the purpose of detecting potential plagiarism. Our work focused on minimizing the *triplet-loss,* which aims to maximize the distance between embeddings that are different while simultaneously minimizing the distance between similar embeddings. These embeddings are referred to as the anchor, positive, and negative.

Throughout music history, composers and songwriters have borrowed musical elements from each other. Similar chord progressions, rhythms, and melodies can often be found spanning different musical styles, genres, and eras. Our interest in the topic was sparked by many famous examples of composers throughout history borrowing musical structures and motifs from each other. One can hear obvious similarities in the works of Bach and Vivaldi from the baroque era and Mozart/Hayden from the classical era. In the music for Star Wars in 1977, John Williams incorporated similar leitmotifs as composer Igor Stravinsky used in his famous 1913 ballet, *The Rite of Spring*. In modern music, courts have adjudicated disputes over similarities between songs. In 2015, for example, artists Robin Thicke and Pharrell Williams were sued for allegedly plagiarizing Marvin Gaye's song "Got to Give it Up" when writing their song "Blurred Lines." Marvin Gaye's estate ultimately won the $7.4 million case. These are just a few examples of audible similarities between musical works. Our project aimed to use deep learning to assess the similarity between music to potentially establish a more robust way to potentially detect music plagiarism.

Stakeholders for which our results may be relevant include: artists looking to ensure their work is not plagiarized by others, record companies and courts looking to have an objective measure of song similarity, and companies looking to better classify and recommend wider arrays of music to listeners. KPIs include that the model minimizes the triplet-loss and beats a baseline of calculating the triplet-loss without feeding audio files into the model. We also wanted to investigate famous plagiarism cases and how such songs may have commonalities detectable by AI models.

Our data primarily came from the Million Song Dataset MSD) as well as several Kaggle datasets where authors paired known songs with links to previews of the songs from various APIs. We also wrote scripts to find song previews where we did not already have them. We ultimately focused on 10s-30s clips of songs from a 50k song subset of the MSD from Kaggle where links to audio were provided. We explored a wide array of models, including building a CNN from scratch and fine-tuning a variety of pre-trained models such as ResNet-18 and various transformer models such as Audio Spectrogram Transformer. The best performing model was the ResNet-18 fine tuned on a dataset of 10k triplets of songs. It beat the baseline by about 84.83%, doing a much better job of separating the song embeddings in the 128-dimensional embeddings space. Positive songs were generated using augmentations of the anchor songs and lower layers of the ResNet were frozen, leaving only the last residual block and fully connected layer. Ultimately, we feel that we achieved our KPIs and created a model that shows promise going forward.

We faced a number of key roadblocks, particularly lack of compute and storage. Audio data sets are complex and extremely large, which limits us to training smaller datasets for fewer epochs. We deployed our model on a different set of songs and covers of those songs with limited success, but the model showed promise for having such limited data and training. In addition to increased compute/storage, improving data quality is a top priority going forward. Other priorities include aggregating different recordings of the same song to create more difficult triplets for the model to learn, studying embeddings using different audio processing techniques, and experimenting with additional model architectures. We also want to train a plagiarism classifier head on top of our existing embeddings model.