Lyrical Distinction

Jesse Bartola
jbartola@

Phillip Michalowski
pmichalowski@

Nischal Tamang
ntamang@

Max Berezin
mberezin@

1 Introduction

Music is an important part of our everyday lives. Technological advances in modern society have made it easier than ever for new artists to distribute their songs over the Internet. Unfortunately, the massive volume of songs that are regularly released into the public can make it difficult to discern which song is made by which artist. Moreover, a more important question arises: Are the most popular artists the ones with the most distinct music?

Aside from sound, the words an artist uses in a song are perhaps the most important determining factor in how it will appeal to the public. Lyrics contain semantic relevance, which allows for the possibility of implicit analysis of a song's cultural relevance. Some of the greatest artists of all time are renown for their unique use of lyricism. Our project will aim to determine if (a) Given a set of lyrics, we can predict with a high degree of certainty who the artist is, and (b) Given our trained prediction model, we can rank artists by their relative lyrical uniqueness and use that to predict their popularity.

2 Related Work

There exists many research projects on the topic of artist classification using lyrics. Stephanie and Scott (2013) try to predict hip hop artists from hip hop lyrics. They focus their research on 4 specific artists: Eminem, Nicki Minaj, Kanye West and Nas. They acquired 30-35 songs for each artist. Their approaches consist of Multinomial Naive Bayes, SVM and Decision Trees. For each of these models, they utilize 2 different feature vectors. The first is a simple bag of words. They selected the 16 most frequent words for each artist. The second feature vector consists of 16 most significant words for each artist, handpicked by Guo and Khamphoune themselves. The hand selected features utilized with the Naive Bayes had a percent error of just 20%, so this is a process that works quite well. Both SVMs and Decision Trees with the hand selected features had a percent error of 35%. Then Naive Bayes and Decision Trees with the objective features performed at a percent error of a little below 30%. The SVM in comparison had a percent error of 35% when utilized with the objective features. However, the research investigates only 4 artists. These 4 artists were hand picked due to their distinguishable differences in lyrical content. For example, Eminem raps about his negligent mother while Nicki Minaj raps about feminism and her conflicts with other female rappers. Classification with a wider range of artists could result in a reduction in accuracy.

A related idea to artist classification is Genre Classification. Each artist is associated with different lyrics in a similar way to how each genre contains certain lyrics more often. Alexandros (2017) attempts to classify music genres from lyrics using Hierarchical Attention Networks (HAN). HAN outperforms N-gram, SVM, KNN and Naive Bayes by utilizing the property that words combine to make lines, lines combine to make segments, and segments combine to make the entire lyrical content. It recognizes that the semantics of lyrics are embedded in the ordering of content within these levels. HAN works by extracting these layers and learning the importance of words, lines and segments. To represent the words being fed into the HAN, Tsaptsinos uses 100 dimensional GloVe embeddings. GloVe outperforms similar word2vec and SVM based models. A benefit of this approach is we do not have to hand pick features. A negative aspect of this approach is that we do not possess familiarity with the processes, and there would be a learning curve for using both HAN and GloVe. This approach performs with an accuracy of 49.50% whereas a higher value would be ideal.

3 Our Approach

We will begin by assembling labeled training data mapping artist names to their respective song lyrics. To start, we'll create a baseline Bayes Net model to predict the most likely artist given a string of lyrics. This will be a more simple implementation to prototype the process. Afterwards, we'll be moving on to more advanced models such as RNN's, where we will aim for better accuracy and prediction of which artist is most likely to have written a given song. This section will take the most time, as we will be not using existing methods directly, but will modify existing approaches for this specific task.

When our model is developed, we will analyze the results by projecting the parameters for each artist onto a 2-dimensional space, highlighting the degree of similarity between lyrics used by all artists in our training data. We will do so using a data visualization website that utilizes the model that we've developed. This will help us better understand whether popular artists are truly the ones with unique lyricism.

3.1 Milestones & Schedule

We will be working on each individual task as a group, making github commits over time to implement the various features that we will have defined.

1. Acquire Lyric & Artist training data from existing Data Sets (1 week) - October 27

2. Build prototype using Bayes Net (1 week) - November 3

3. Build Neural Network model (1.5 weeks) - November 15

4. Progress Report (due Nov. 16)

5. Data Visualization Website (0.5 weeks) - November 19

6. Analyze model results and plot in visualization (2 weeks) - December 3

7. Final report and presentation preparation (2 weeks) - Presentation Day

4 Data

There are many online datasets available containing song lyrics with their respective artists. A simple search on Kaggle returns several datasets of lyrics labeled with artist and song name. For example, Mousehead (2017) contains a repository of over 50,000 songs of a large variety of artists and genres, including ABBA and Aerosmith. This dataset will provide us with the large variety of lyrical data we need to train our model. If more information is needed, we can find larger sets of lyrics on websites such as MetroLyrics and Lyricfreak. Some of the data outside of this primary dataset may even be annotated by genre if necessary.

5 Tools

Many existing Open Source libraries and technologies exist for the tasks we will be pursuing, as Music Information Retrieval and Deep Learning are currently popular fields in which companies and research groups are investing heavily. As we will be dividing the song lyrics into their constituent levels, we will be using the nltk package in python (**?**) for the task of dividing the text into its tokenized parts. There are two libraries at the forefront in the process of building the Neural Network: Tensor Flow (**?**) and Pytorch (**?**). Both are extremely well maintained by their respective parent companies, Google and Facebook, and have great documentation for beginners. That being said, there are some important differences to consider. Most individuals in research seem to prefer Pytorch, as they say it's more 'pythonic' and doesn't hide away functionality of various methods. They also create their computational graphs in different ways. Pytorch has dynamic graphs, as opposed to the static graphs of TensorFlow. These dynamic graphs are apparently helpful for using variable length inputs in RNNs. Based on the fact that the community seems to agree that Pytorch is better for smaller scale Deep Learning projects, along with how it accomplishes certain tasks more efficiently than TensorFlow, Pytorch will be our primary library of choice. We will also be using numpy, which is a library that implements a variety of multi-dimensional array operations. As of now, we are not aware of whether or not we will need GPU servicing, however if we do we will investigate services such as Colaboratory by Google (**?**).

References

Alexandros, T. (2017). *Lyrics Based Music Genre Classification Using a Hierarchical Attention Network*.

Mousehead (2017). 55000+ song lyrics — kaggle.

Stephanie, G. and Scott, K. (2013). *I'm different, yeah I'm different: Classifying RapLyrics by Artist*.