Mood Classification of Song Lyrics Using Deep Learning

Cyprian Gascoigne, Jack Workman, Yuchen Zhang
MIDS W266 (Instructor: Daniel Cer)
University of California, Berkeley, 2018

Abstract

In this paper, we analyze mood classification based on song lyrics. Mood classification of lyrics can help in the creation of automatic playlists, a music search engine, labeling for digital music libraries, and other recommendation systems.

We compare the accuracy of deep learning vs traditional machine learning approaches for classifying the mood of songs, using lyrics as features. We will use mood categories derived from Russell's model of affect (from psychology, where mood is represented by vector in 2D valence-arousal space) and also calculate a valence-happiness rating.

We will test the extensibility of our deep learning model using the Million Song Dataset (MSD), a freely available contemporary music track dataset. From MSD, we will use the Last.fm and musiXmatch datasets for song tags and lyrics. We will also use language detection to focus on English lyrics.

1 Introduction

Music recommendation engines are becoming more and more expected norms for listeners and music lovers today. Since mood and emotional classification is crucial to the entertainment value of songs, music recommendation algorithms can take into account mood classification. Expert, and even layman, human assessment is still often superior to machine learning methodology, however as technology improves, appropriately classified mood labels can greatly ease this process, and can even be used as metadata in digital music libraries and repositories.

Song lyrics are different from ordinary text in that they often use more stylistic qualities like rhyming, and other forms, often contributing to the emotional value (Lee et al., 2010). They are also shorter and often have smaller vocabularies than other text documents. This, combined with their more poetic style can cause more ambiguity when it comes to mood identification (Cano et al., 2017).

Much research on music classification is based on audio analysis versus lyric analysis. (Corona et al., 2015). However, classifiers that incorporate textual features can outperform audio-only classifiers (Fell et al., 2014). In this paper, we focus on the analysis of lyrics for classifying mood categories. We use mood and emotion as interchangeable concepts. Classifying mood (or "sentiment") using textual features has been studied less than musical features. One reason may be in obtaining a large dataset legally (as lyrics are copyrighted material).

We use the Million Song Dataset (MSD), a large freely-available dataset, and supplement it with

lyrics scraping we conducted separately. In this paper, we measure the effectiveness and accuracy of different machine learning models on mood prediction, in particular, deep learning, neural network models.

This paper is organized as follows: Section 2 reviews related work on lyrics based mood classification. Section 3 outlines the methods (design and implementation) used in our analysis. Section 4 discusses results, including a detailed analysis of obtained results. Conclusion and recommendations are summarized in Section 5.

2 Background and Literature Review

Previous work reached contradictory conclusions and employed smaller datasets and simpler methodology like td-idf weighted bag-of-words to derive unigrams, fuzzy clustering, and linear regression. Our proposed neural network model approach should have better accuracy.

There has been past research focused on using joint audio-lyric models (Mihalcea et al., 2012). Using SVM, classification accuracy between lyrics only, audio only, and joint lyric and audio classifiers, this study found that joint is most accurate. Similar results were found in other research using SVM implementations (Chen et al., 2009), although not all mood categories saw combined feature sets outperforming lyric or audio only features.

Our focus on lyrics only is akin to that of music classification research performed by Corona and O'Mahony which showed that lyrics alone can be used to classify music mood achieving an accuracy of 70% with the MSD (Corona et al., 2015). In this study, SVM classification algorithms were used, and some moods (e.g., anger) were found to be easier to detect. Our choice of neural networks is based on research demonstrating that neural network based systems can often be transferable to other languages more easily (Becker et al., 2017).

Though our focus is on English language lyrics, there have been studies conducted on lyrics in other languages, such as Chinese (Chen et al., 2009) and Hindi (Bandyopadhyay et al., 2015), that also used mood affects predicted based on combinations of text stylistic features and features based on n-grams (e.g., term frequency-inverse document frequency scores of trigrams).

Genre classification based on lyrics has been studied using shallower textual features like bags-of-words. One study used n-gram modeling to predict one of 8 genres, by honing in on the topic of the text, examining features like length, use of pronouns, past tense, repetitive structures, rhyme features, etc. (Fell et al., 2014). Certain genres, like rap, were found to have more unique properties (e.g., complex rhyme, long lyrics, distinctive vocab), that make it easier to classify than more difficult and frequently confused genres like Folk, Blues, and Country. These often confused genres share similar topics and are stylistically and structurally similar. In these instances, audio and musical properties may be better than lyrics at accurate prediction.

A recent study on sentiment analysis of lyrics showed that music corresponding to happy moods can be predicted with greater accuracies (Raschka, 2014). This study used a Naive Bayes classifier trained on lyrics alone, also pulled from the MSD. Specifically, their best performing model was a multinomial Naive Bayes classifier with unigram tf-idf feature representation, better than the Bernouilli Naive Bayes model. Increasing the n-gram range and other tuning such as parameter smoothing had little effect on classification performance.

3 Methodology

Data Acquisition

The MSD provides mood mapping at a song but not lyric level. A component of the MSD includes Last.fm, which maps songs to user “tags” based on Last.fm user input. These tags can be anything from a human emotion to genre to animals. Another is the MusiXmatch dataset, which is a collection of song lyrics mapped to song in a bag-of-words format. The Last.fm

dataset contains song-level tags for more than 500,000 songs. The mood categories are derived using the social tags found in this dataset.

One initial challenge was in acquiring corresponding lyrics to song data. To obtain the 35,000+ lyrics in our dataset, we used the python package lyricsgenius for retrieving lyrics. The package interfaces with the www.genius.com API for lyric access, and iterates through the artist-song pairs in the Musixmatch file.

After scraping and downloading lyrics, we next index the files and perform basic checks on the validity of each. We drop all songs that are non-English, do not have lyrics available, and do not have a matched mood as classifying across languages is out of scope of this project and no classification can be done on a song without lyrics or without a matched mood.

Word Embeddings

To form our word embeddings, we make use of the word2vec model (Mikolov et al., 2013) and the implementation provided by TensorFlow. Word embeddings are built from full set of song lyrics (including those without labels).

The script we use contains a lyrics2vec python class that saves its embeddings and data as python pickle files. The steps we follow for our analysis are: We first build a list containing all words in the dataset. We then extract the top most common words from the vocabulary to include in our embedding vector. We gather together all the unique words and index them with a unique integer value to create an equivalent one-hot type input for the word. We loop through every word in the vocabulary dataset and assign it to the unique integer word identified to allow easy lookup / processing of the word data stream. Contractions and slang words (e.g., “wanna”) are split because these entities are grammatically speaking, two separate words.

Mood Annotation

We use the popular Russell’s model of affect and focus on 18 mood classes similar to previous studies (Chen et al., 2009). In Russell’s model, mood is represented by a vector in the 2-D valence-arousal space where mood is an element of (valence, arousal):

$$m \, \varepsilon \, M \; = \; (v, \; a)$$

Valence measures the good vs bad dimension (pleasant / unpleasant). Arousal measures the active vs passive dimension of sentiment (aroused / sleepy). Mood in our model refers to categories to be learned in the classification problem. We group mood into 18 categories and match tags to moods.

Mapping Labels

Once we have the index built, we can easily match the lyrics to the mood tags from the Last.fm dataset. We explore the tags available in the Last.fm dataset with a special focus on the moods targeted in our project. The dataset is available in several different forms including individual json files for each track as well as an sqlite db. Iterating over the json files is cumbersome, so we make use of the sqlite db. We use sql commands to expose its table structure in order to gain an understanding of the database schema, which we find out to be quite simple. We iterate over each row of the index, query the sqlite Last.fm database for all associated tags, then attempt to match tags against our Mood Categories.

For each mood, we query all tracks that match exactly with the mood and its siblings. We then query all tracks that match with LIKE the mood and its siblings with some small modifications if necessary (e.g., instead of an exact match on 'meditation', we query for a LIKE match on 'meditat' to allow for different conjugations). Indeed, the LIKE search uncovered a lot of missing tags. We then manually review the returned tags and build a set of filters to remove tags that match but are not appropriate.

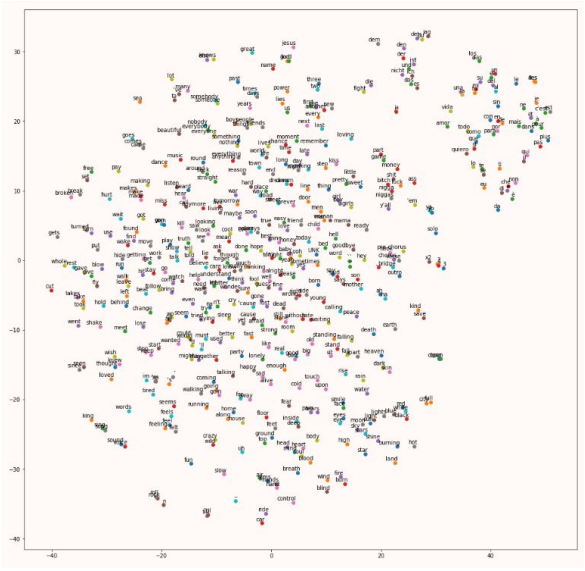
For genre classification, we use the MSD Allmusic Genre Dataset (MAGD), which contains all genre labels collected from Allmusic.com. This includes generic genres like Religious or Christmas, non-musical content like Comedy/Spoken and significantly small genres (Schreiber et al., 2015).

Model Training and Tuning

Once the vocabulary is constructed, we can build and train a word2vec model. We produced embeddings for 35,000+ songs from our full set of 250,000+ lyrics.

Our training methodology is as follows: Given a specific word in the middle of a sentence (the input word), we examine the words nearby and pick one at random. The network then tells us the probability for every word in our vocabulary of being the “nearby word” that we chose. We train the neural network to do this by feeding it word pairs.

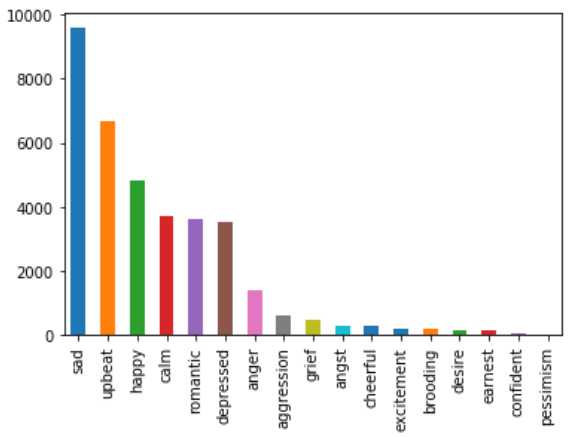
Once we begin training, we can feed our metadata variable for visualizing the graph in TensorBoard. The average loss computed is an estimate of the loss over the last 2000 batches. With our final embeddings, we can write corresponding labels for the embeddings. We can then create a configuration for visualizing embeddings with the labels in TensorBoard. Below is the visualization of distance between embeddings.



Interestingly "rain" and "water" are close, "leave" and "left" are close, "yeah" and "yes" are close. Training time for the embeddings on all 250,000+ songs (unfiltered) was < 10 minutes.

Baseline Mood Classification

From our embeddings model based on word2vec, we can produce real mood classification results with a neural network. The remaining lyrics are then randomly divided into a training set, dev set, and test set, each with similar mood distributions (21,501; 7,167; 7,167 lyrics respectively). Then we establish a baseline classification using simple classifiers. Finally, we can use our neural network architecture for modeling mood classification. The dataset consists of a large number of text files where each file represents a different song. Our index contains song lyrics with matched moods. We then create a categorical data column for moods before reading in the lyrics of each song. Below are the 35,835 songs with their respectively matched moods:



The Naive-Bayes classification requires the actual lyrical text, thus, we begin by reading into memory the text for each song in our dataset. Afterwards, we use the python sklearn package to vectorize and process the lyrical text, fit the Naive Bayes Classifier, and compute the accuracy. We also use SVMs for comparison.

Neural Network Classification

Our first neural network model is CNN. We perform the following data processing steps on all lyrics: Truncate/extend all songs to the 75% word count percentile; Tokenize lyrics with nltk's word_tokenize function; Remove all stopwords that match from within nltk's stopwords corpus; Remove punctuation. All songs will then be limited to 282 words.

With our normalized lyrics, we can then perform our CNN for text classification, which uses an embedding layer, followed by a convolutional, max-pooling and softmax layer. We calculate mean cross-entropy loss and accuracy in our training, with lyrics converted into 2D numpy arrays.

4 Results and Discussion

For our baseline classification, we found the most common case for each dataset split is the mood category: "sad." The accuracy of the most common case classifier is roughly 26-27% for each split (Train: 27.02%; Dev: 26.11%; Test: 26.34%).

For Naive Bayes, we convert lyrics to counts and term-frequencies. The accuracies we get with Naive Bayes Classifier are: 29.64% (Dev accuracy) and 30.25% (Test accuracy).

We then compare Naive Bayes results with SVM, and find that SVMs (more widely used in prior studies) provide increased accuracy. The accuracies we get with SVM Classifier are: 41.61% (Dev accuracy) and 42.47% (Test accuracy).

Our CNN results…

(First stab at a CNN for mood classification is yielding an accuracy rate of ~48% with some rather egregious overfitting. That's 6% better than our SVM.)

5 Conclusion and Recommendations

Thus, we compared classifiers for common case, Naive Bayes, SVM, and neural networks (CNN). The results indicated that the classifier that performed best (by achieving an accuracy of...

Lyrics based mood classification is just beginning. As more music moves online, intelligent and accurate music recommendation systems will be greatly enhanced by the use of lyrics to aid prediction. One downside is that the dataset used still could use improvement as the smaller scale imposes limitations on the methodology. As larger datasets become more readily available, the real-life applicability of these algorithms will become easier to vet. Also in the future, we may want to do some analysis on a few songs that looks at the verse by verse mood and then relate that to the overall mood (e.g., sad songs mostly sad all the time or they tend to have on average X% happy verses).

Another angle is looking at happiness (emotional states) additionally. Studies have

found that the happiness of song lyrics has been trending downward since the 60s, though sometimes stable within genres (Danforth et al., 2009).

References

Bandyopadhyay, Sivaji, Das, Dipankar, and Patra, Braja Gopal. (2015). Mood Classification of Hindi Songs based on Lyrics.

Becker, Maria, Frank, Anette, Nastase, Vivi, Palmer, Alexis, and Staniek, Michael. (2017). Classifying Semantic Clause Types: Modeling Context and Genre Characteristics with Recurrent Neural Networks and Attention.

Bertin-Mahieux, T, Ellis, D, Lamere, P, and Whitman, Brian. (2011). "The Million Song Dataset."

Cano, Erion and Morisio, Maurizio. (2017). "MoodyLyrics: A Sentiment Annotated Lyrics Dataset."

Chen, Xiaoou, Hu, Yajie and Yang, Deshun. (2009). "Lyric-based Song Emotion Detection with Affective Lexicon and Fuzzy Clustering Method."

Corona, Humberto and O'Mahony, Michael. (2015). An Exploration of Mood Classification in the Million Songs Dataset.

Danforth, Christopher M. and Dodds, Peter Sheridan. (2009). Measuring the Happiness of Large-Scale Written Expression: Songs, Blogs, and Presidents

Downie, J. Stephen, Ehmann, Andreas F., and Hu, Xiao. (2009). Lyric Text Mining in Music Mood Classification.

Fell, Michael and Sporleder, Caroline. (2014). Lyrics-based Analysis and Classification of Music.

Lee, Won-Sook and Yang, Dan. (2010). Music Emotion Identification from Lyrics.

Manning, Christopher D. and Wang, Sida. (2012). Baselines and Bigrams: Simple, Good Sentiment and Topic Classification.

Mihalcea, Rada and Strapparava, Carlo. (2012). Lyrics, Music, and Emotions.

Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Greg, and Dean, Jeffrey. (2013). Distributed Representations of Words and Phrases and Their Compositionality.

Raschka, Sebastian. (2014). MusicMood: Predicting the Mood of Music from Song Lyrics Using Machine Learning.

Russell, J. A. (1980). "A Circumplex Model of Affect," Journal of Personality and Social Psychology, vol. 39, no. 6.

Schreiber, Hendrik. (2015). Improving Genre Annotations for the Million Song Dataset.