**Music Lyrics Classification and Generation**

**Ivan Amies, Nicholas Grifka, Hang Shang, Alex Turner, Luyao Yuan**

University of Michigan

EECS 445 Final Report Fall 2015

{amiesi, ngrifka, hshangaa, alexlt, yuanluya}@umich.edu

**Abstract**

Music is a form of expression filled with variety. Each song has unique lyrics and a unique feel. In addition, we predicted that each genre, besides having unique musical characteristics also has unique lyrical characteristics. In this paper we show that we can accurately predict, using state-of-the-art machine learning algorithms, a song’s genre using only the lyrical text associated with a song. We compare the performance of several different approaches including support vector machines, naïve bayes, gaussian mixed models, and neural networks. We show that the performance of neural networks on predicting a song’s genre based on the lyrical text vastly out performs previous work done and the performance of other classifiers. In addition to classifying songs into genres based on their lyrics, we propose a method to generate original lyrical text in the style of a given artist. We show that using a recurrent neural network we can get good subjective results.

**1 Problem Statement**

The challenge of classifying songs into genres is a natural task for humans, but a far more challenging one for machines. One way of discovering the different categories of music is to examine lyrics. Certain types of music tend to use the same words frequently. For example, country music might associate with the words “tractor” or “beer.” While pop music might frequently use the words “party” or “dance”. Prior work has shown progress into the analysis of lyrics to categorize songs into music genres. However, none has yet produced high accuracies. Our project replicates this prior work and furthers it to achieve high classification accuracy.

Another problem we address in our project is generating original lyrical text, given a large sample of an artist’s existing lyrics. The challenge of creating a machine that can generate authentic lyrics is not well researched. One of the few prior works found however, the DeepBeat project, attempted to piece together existing rap music lyrical lines in order to produce better rhymes. They did achieve impressive results. Their results show that, given the right algorithms and enough effort, machines are capable of manipulating phrases into a musical form pleasing to the human ear. We attempted to, and succeeded at, generating lyrics at the character level, producing lyrics with more originality than prior works.

**2 Motivation**

Our main motivation for applying a neural network to the lyric classification task is that neural networks (NNs) have been used for other text based classifications and have shown good results8. In addition, we have found no prior work which has applied NNs to song lyrical information. In prior work, we have seen the using of hidden markov models (HMMs), gaussian mixed models (GMMs), support vector machines (SVMs), naïve bayes (NB), k-nearest-neighbors (k-NNs), and k-means. Out of all of these different approaches we found that the best accuracy was 53% over 7 categories4. In other text based classifications we have seen neural networks vastly outperform traditional methods of machine learning.

Music genre classification is a hard problem. It is a problem which, if improved upon, would improve the quality of numerous services. One category of services that stands to be improved is online music streaming services such as Apply Music, Spotify, and Pandora. Each of these services has millions of users and millions of songs. One aspect all of these services could improve upon is better predicting user's tastes in songs. One way to do this is to improve upon the classification of music, providing more specific genres for their algorithms to use to better predict a user’s taste. With the ability to automatically classify songs into accurate genres, these services might be able to classify songs into sub-genres, allowing them to fine-tune their algorithms to a user’s taste. Currently, songs have to be manually sorted into genres. Providing the ability to automatically sort songs into genres automatically would reduce the burden upon these services and improve the quality of suggestions and predictions for users.

Many people find it difficult to write lyrics that are thoughtful and original. Lyrics are a complex form of expression, if writing lyrics is a challenge for people, can machines help write lyrics and even produce lyrics entirely on their own that sounds as good or better than people? We believe machines can at least help people write lyrics. If machines can help people write lyrics, then we might see more originality in songs. Currently, a lot of songs, namely pop/rock songs, are not very original. One such obvious example is the four chords song9. While it exemplifies this problem for musical qualities, this is also true of lyrics. With the help of machines, we believe this might change and our music, and therefore our culture, will be improved for the better.

**3 Approach**

In this section, we discuss the methods and models used to classify music genres by lyrics and our approach to generating music lyrics.

**3.1 Database**

We used the Million Song Dataset (MSD)1 to acquire song tracks. In association with the MSD, we used the musiXmatch dataset2 for our source of song lyrics, and the taugtraum genre annotations dataset3 for mapping our songs to genres. Because of the copyright problem, we can only access lyrics in bag-of-words format. The MSD dataset already extract 5000 vocabulary as features, and divided all the data into training and testing sets.

**3.2 Genre Classification**

For genre classification, we chose to classify the seven most representative genres in the dataset: Pop/Rock, Electronic, Rap, Country, R&B, Latin, and International. We combined Pop/Rock because prior work has shown that their lyrical qualities are extremely similar.4,5 We used six different classifiers, Support Vector Machine (SVM) (with linear and Gaussian kernel), Naïve Bayes (under Multinomial and Gaussian distribution), Logistic Regression, and a Neural Network (NN).

Raw data was transformed into a feature matrix with a Python script. Since the data was already in a bag-of-words format, we didn’t do further preprocessing. Every song within the feature matrix has a 5000-feature vector, with each entry representing the frequency of the corresponding word in the lyrics. The raw term-feature (tf) matrix was fed to the Naïve Bayes classifier using a Multinomial distribution, while a term frequency-inverse document frequency (tf-idf) matrix was used as input for our neural network, SVMs, logistic regression, and Naïve Bayes using a Gaussian distribution classifiers. The tf-idf matrix was transformed from raw tf matrix using the Scikit-Learn machine learning package. NNs usually learn effective features automatically, and provide outstanding accuracy in natural language processing field. However NNs, due to excessively complex structure, are likely to introduce overfitting. In order to address this problem, we use principal component analysis (PCA) to extract useful features from raw data and reduce the complexity of the model. To decide the number of optimal features, we did cross validation across different feature sizes, Figure 1 below.

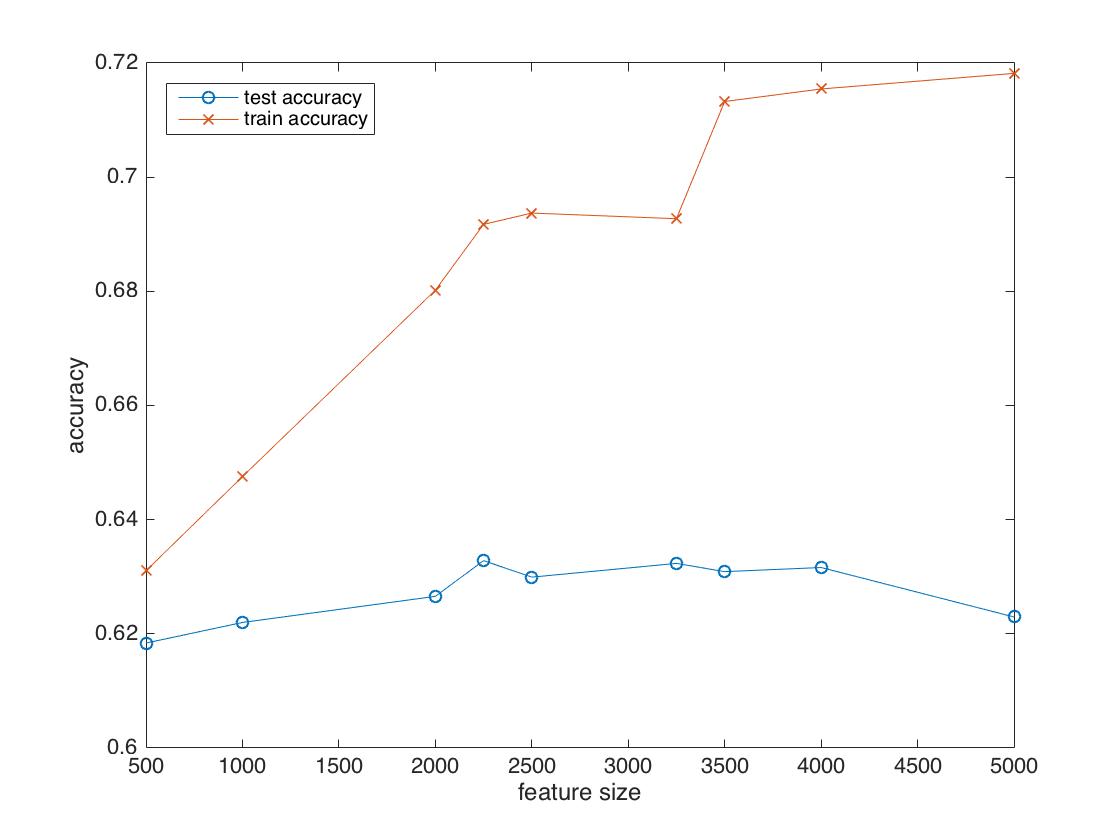


Figure 1: Feature Size and Accuracies.

To achieve a high accuracy, we tried several different structures for our neural network. We started with two hidden layers and initialized the number of neurons in each layer as the average of its input and output sizes. We tuned the number of neurons using a step size of 50 units. We used a mean-squared error function for our loss function. Finally, we optimized our neural network using stochastic gradient descent with a batch size of 10. After extensive testing, we decided on the following neural network, shown in Figure 2, below.

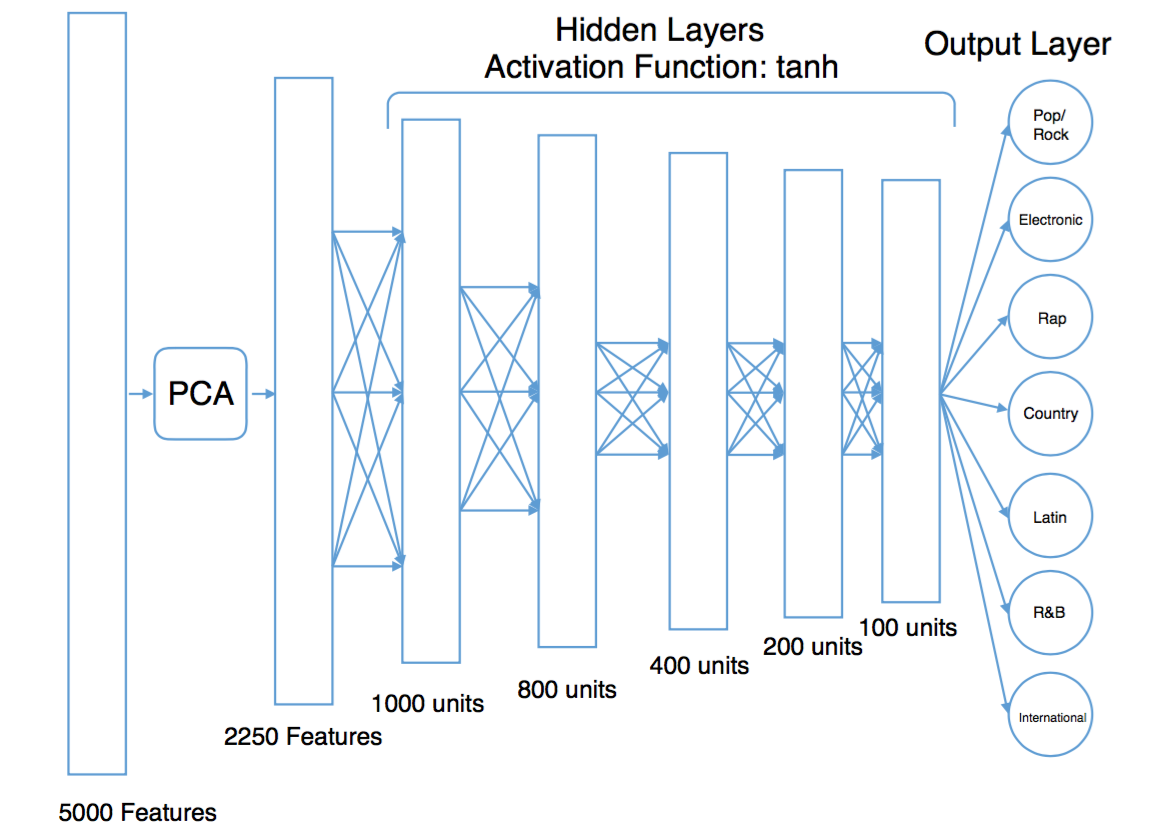


Figure 2: Our neural network structure.

**3.3 Lyric Generation**

Generating lyrics *de-novo* involves capturing many layers of complexity. Recurrent neural networks (RNNs) are one model capable of holding information rich data features. This approach comes recommended by researchers in the field such as Andrej Karpathy6. Here, we present the generation of lyrics in the Rap and Country musical genres using three models. One generates Rap lyrics exclusively, the second Country lyrics exclusively, and the last a mix of both Rap and Country.

Lyrical corpora 500,000 characters long were generated from the artistic portfolios of the singers Johnny Cash and 50 Cent. These artists were chosen in recognition of their contributions to the genres of country music and rap music, respectively. Their songs were assembled in separate textual documents, and arranged consecutively. To create a fusion training set of Johnny Cash and 50 Cent, lyrics for both songs were appended. The character level representation of the feature matrixes extracted in subsequent steps allowed this approach.  
 The lyrics were read through a Python script. This includes all characters presented within the corpora. Upon reading, the data was enumerated into discrete indices and in the process all text was cut into semi-redundant sections. At the end of feature extraction, if a character existed in a sentence, it was placed into a matrix. The data was extracted at the character-level so novel grammar and vernacular slang could be produced at the generation steps.

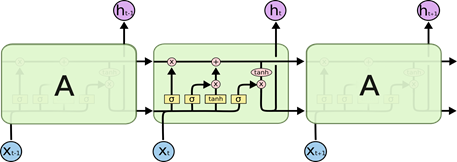


Figure 3: The structure and layout of the LSTM, including layers

To accomplish text generation, a stacked Long Short Term Memory (LSTM) RNN with used. Its structure can be referenced in Figure 3, above. RNN stacking allows greater model complexity, combining synergistically with the memory features of the recurrent neural net. Indeed, the effect is towards finer description of hidden feature details. For the LSTM implementation itself, the Keras library was used in combination with the Theano scientific computing library. Each LSTM layer had 512 hidden nodes. A graphics processor was recruited to accelerate training time using its one gigabyte of processing power.  
 The results were run for fifty epochs of training time each. After which, the models were “unwound,” generating nine songs of two-thousand five-hundred characters each. Sample results can be found in Figure 4, below.







Figure 4: Sample Lyrics Generation Results

**4 Results discussion**

The results, presented in Figure 5 below, show that the neural network outperforms all other methods with an accuracy of 63.3%. Our results also show that softmax and SVM perform similarly to our neural network, with an accuracy of 61.3% and 61.0%, respectively. All other models exhibit poor performance in comparison. Naïve Bayes exhibits poor performance. In addition, our models which assume the data is distributed according to a Gaussian distribution also exhibit poor performance.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | SVM Gaussian kernel | Naive Bayes Gaussian | Naive Bayes Multinomial | SVM linear kernel | Softmax | Neural Network |
| Accuracy | 27.5% | 38.8% | 57.1% | 61.0% | 61.3% | 63.3% |

Figure 5: Accuracies for Lyric Classifier Models

Our neural network achieved the best results with 63.3% classification accuracy. It was also our most complex model. This demonstrates that lyrics exhibit significant overlap between words used in lyrics across genres. To classify lyrics with higher accuracy, we believe one would need to have a model of much greater complexity to successfully classify with better accuracy patterns of words unique to genres. In other words, using a neural network one would need to either increase the number of hidden layers, or increase the number of units per hidden layer, or both. We believe that both would be required. Another way to increase classification would be to have better feature selection. Since our results demonstrate that the lyrics exhibit significant overlap between words used across genres, one might do further experimentation to determine the most significant words per genres and use those as features for classification.

We believe Naïve Bayes exhibits poor performance because of its assumption that frequencies of terms within lyrics are independent of one another. We believe this assumption of independence between frequencies of terms to be false within a phrase as the presence of a term in a document tends to be highly dependent on the terms around it. Articles such as “a” and “the” are highly dependent on the terms around them. However, even if we strip these articles from the dataset, we still have highly dependent terms. One such example is, if one has the term “tractor” in a sentence, it is much more likely to also find “farm” than to find the term “netflix” in the sentence. Therefore, the results of classification using Naïve Bayes will not be as accurate as one might expect. Our results confirm this hypothesis, Naïve Bayes has an accuracy of 57.1% using a Multinomial distribution and 38.8% using a Gaussian distribution.

Models which assume the data is distributed according to a Gaussian distribution also exhibit poor performance. We believe this is because the term frequencies in the lyrics do not follow a Gaussian distribution. Since our models rely on the assumption that the frequency of terms is a gaussian distribution, our models exhibit poor performance. Our results confirm this hypothesis, Naïve Bayes using a Gaussian distribution has an accuracy of 38.8% and SVM using a Gaussian kernel has an accuracy of 27.5%.

**5 Conclusion**

Our results demonstrate the lyrics exhibit some qualities unique to the genre of the song. In addition, using these qualities, machine learning algorithms can successfully classify the genre of lyrics. The method that gave the greatest accuracy was our neural network. It exhibited significantly better performance than any of the other methods we used. Our findings show that while a neural network exhibits better performance when compared to other models, the performance is not very good. At best, we achieved 63.3% classification accuracy using our neural network. This demonstrates that lyrics exhibit significant overlap between words used in lyrics across genres. To classify lyrics with higher accuracy, we believe one would need to have a model of much greater complexity to successfully classify with better accuracy patterns of words unique to genres. One might also determine better features to do classification over. Both would increase the classification accuracy. Finally, we replicated prior research done by students at Stanford,4 Carnegie Mellon,5 and exceeded their results using similar models. We believe this is due to better feature selection. In addition, we have shown that using a neural network, we can outperform all prior research and other models.

Our results also demonstrate that we have successfully generated original lyrics using a recurrent neural network. Furthermore, the lyrics generated can be in the style of a specified artist. This work improves upon the DeepBeat7 project, which generates a set of rap lyrics from combinations of real rap lines. In this project, we generate lyrics character by character, instead of line by line. This greatly increases the originality of the lyrics. Despite a few misspellings of words and some grammatical mistakes, the resulting lyrics are quite remarkable, and may provide a glimpse of how song lyrics might be written in the future.

In the future, one might consider generating lyrics that satisfy constraints other than being in the style of one artist. One constraint could be that the lyrics must exhibit a specific mood, such as happiness or sadness. Another constraint could be that the lyrics must use a given keyword or phrase. By using different constraints and combining multiple constraints, one could further specify what kind of lyrics will be generated from the model.

**References**

[1] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The Million Song Dataset. In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011.

[2] musiXmatch dataset, the official lyrics collection for the Million Song Dataset,   
available at:<http://labrosa.ee.columbia.edu/millionsong/musixmatch>

[3] Hendrik Schreiber. Improving genre annotations for the million song dataset. In Proceedings of the 16th International Conference on Music Information Retrieval (ISMIR), pages 241-247, 2015.

[4] Ahmed Bou-Rabee, Keegan Go, Karanveer Mohan. (2012). Classifying the Subjective: Determining Genre of Music From Lyrics. <http://cs229.stanford.edu/proj2012/BourabeeGoMohan-ClassifyingTheSubjectiveDeterminingGenreOfMusicFromLyrics.pdf>

[5] Dawen Liang, Haijie Gu, Brendan O’Connor. (2011). Music Genre Classification with the Million Song Dataset. <http://www.ee.columbia.edu/~dliang/files/FINAL.pdf>

[6] [Andrej Karpathy](http://karpathy.github.io/). (2015).The Unreasonable Effectiveness of Recurrent Neural Networks.<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

[7] Eric Malmi, Pyry Takala, Hannu Toivonen, Tapani Raiko, Aristides Gionis. (2015). DopeLearning: A Computational Approach to Rap Lyrics Generation. <http://www.deepbeat.org/>

[8] Nan Yang, Shujie Liu, Mu Li, Ming Zhou, Nenghai Yu. (2013). Word Alignment Modeling with Context Dependent Deep Neural Network. <http://www.aclweb.org/anthology/P13-1017.pdf>

[9] The Axis of Awesome. (2011). 4 Chords. YouTube. <https://www.youtube.com/watch?v=oOlDewpCfZQ>

Source Code Available at:

<https://github.com/bluechill/Music-Lyrics-Classification-and-Generation>