BEYOND THE BEAT: LEVERAGING LYRIC CONTENT AND SENTIMENT FOR SONG GENRE CLASSIFICATION

Courtney Maynard

Department of Data Science, College of William and Mary

DATA 340: Natural Language Processing

Dr. James Tucker

December 19th, 2023

AUTHOR NOTE

All code and documentation be found at the following GitHub:

github.com/courtney-maynard/DATA340NLP-SemesterProject

ABSTRACT

Music recommendation systems utilize audio features to recommend songs for users but neglect the lyrical features of songs such as lyric content, structure, and sentiment. Natural Language Processing (NLP) techniques can give insight into whether song lyrical information can be used to categorize songs into genres to enable better music recommendation systems. Previous literature reveals gaps in multi-class genre classification that utilize non-bag-of-words datasets, machine learning architectures that preserve lyric structure, and combining lyric sentiment analysis with lyric word content. Three ideas were tested regarding their ability to classify songs into different genres: song lyric content in an intact song structure format, song sentiment at the song versus lyric level, and combining the content and sentiment performance in an ensemble manner. Using a dataset of ten musical genres and comparing Logistic Regression (LR), Bi-Directional Long Short Term Memory (Bi-LSTM), and Hierarchical Attention Network (HAN) machine learning techniques on lyric content found the HAN model outperforms other methods. Comparing lyric sentiment at the song level and lyric level saw a nearly two-fold increase in accuracy for lyric-level models. Combining the LR and Bi-LSTM content models with song-level and lyric-level sentiment models, Decision Tree Classifier and Recurrent Neural Network (RNN) respectively, decreased overall ensemble performance. The highest performing ensemble was the HAN content with lyric-level RNN, with an accuracy of 43.07%. Matching the data structure with a model that attends to the nuances of a data form, such as the attention levels of the HAN paired with structural song breakdown, can increase performance and the ability to solve song genre classification and other NLP tasks. Future work can fine-tune the ensemble models for higher accuracy.

1. INTRODUCTION

The popularity of using music recommendation systems to discover new music has increased in recent years, spurring many questions regarding how music is classified into genres or interests. With an increase in interest in music categorization, there has been more scholarly attention on music genre classification, particularly regarding using auditory signals. Though many studies have indicated that auditory features such as bass, timbre, and length of song can be indicative of song genre, there is less research regarding song lyrics as a genre indicator. This report seeks to investigate whether lyric content and sentiment with preserved lyrical structure can be used to classify music into various genres. While some previous works have used machine learning techniques to classify songs using lyric content and structure, there is a lack of research into whether the perceived emotional content or sentiment of a song is indicative of its genre.

The logic and techniques behind popular music recommendation systems remain largely unknown. While it is known that audio features are used by music recommendation systems, such as Spotify, it is unknown whether individual users are receiving songs that may appeal to them lyrically. Discovering whether lyrics can be used to classify songs into different genres leads to the potential to build more personally tailored music recommendation systems for users who prefer to listen to music with a focus on lyrics, rather than on audio or production.

2. LITERATURE REVIEW

Many studies have focused on the audio features of songs, with the exploration of the usefulness of lyrics for song genre classification emerging primarily within the past five years. Within the context of this study, relevant literature falls under the categories of dataset evaluation, emotional and sentiment analysis techniques, embedding techniques, and model architecture variations.

2.1 Dataset Evaluation

One hindrance to large-scale lyrical analysis is the availability of intact lyric datasets (Tsaptsinos, 2017). A majority of research on song genre classification utilizes bag-of-word lyrics or subsets of full lyric corpi due to copyright restrictions or incomplete datasets (Tsaptsinos, 2017). In some studies, researchers have collected personal lyrical data and connected it with existing song genre data sets, such as the Million Song Data Set or various crowd-sourced datasets on Kaggle, as a way of circumventing the bag of words problem for lyric analysis (Liang, Gu, & O'Connor, 2011) (Kumar, Rajpal, & Rathore, 2018) (Dammann & Haugh, 2017). Genre classification is a difficult task due to the large number of genres and 'micro-genres' that songs are classified into; Spotify, for example, has over 6,000 micro-genres, such as 'East Coast Hip Hop' or 'South African Pop Dance.' Due to the proliferation of these micro-genres, all previous studies approach the classification problem in regards to larger, more expansive genres such as 'Indie' or 'Rock' with a note that classification for micro-genres is an area of further exploration. Most previous studies attempted to generalize the songs into fewer than five genres as additional genres weren't available in their chosen dataset. This is problematic for genre classification algorithm application to a wider variety of songs and genres. One previous study classified songs into 20 genres and 117 genres in separate instances to determine if their implementation performed better for an increased number of genres, and both variations of their implementation architecture did not (Tsaptsinos, 2017). This decrease in accuracy may have been because there were more genres for the model to select from, so overall performance decreased.

2.2 Emotional and Sentiment Analysis

Though several studies combined lyrics with other song aspects, such as album artwork or audio features, there was only one study that combined lyrical content with lyric sentiments. The study found that using lyric sentiments, as derived from the emotional valence score of Affective Norms for English Words (ANEW) word lists, provided no signal as to the genre of the song (Liang, Gu, & O'Connor, 2011). Investigation into the ANEW word lists for sentiment analysis indicates that the technique is commonly used in poetry classification and poetry generation, with varying levels of success, since the dataset of emotional words is only 2500 words (Satrio Utomo, Sarno & Suhariyanto, 2018). Since lyrical data contains one hundred times more words, it is likely that many of the songs, and their sentiment, can not be accurately represented through those words. Other poetry generation studies have used a technique and tool called SentiStrength, which is a Java class that estimates the negative and positive sentiment values of a particular segment of text by evaluating the valence scores of words within the text segment and returning the strength of a segment's negative, positive, or neutral sentiment (Misztal-Radecka & Indurkhya, 2014).

2.3 Embedding Techniques

The way that words are embedded into vectors before a machine learning technique is applied can have a significant impact on the model's performance and is a variable in many studies regarding song classification with lyrics. Several studies used custom or pre-trained GloVe embedding techniques (Boonyanit & Dahl) (Tsaptsinos, 2017), while others utilized Word2Vec (Kumar, Rajpal, & Rathore, 2018).

The study that utilized Word2Vec used many different machine learning techniques and compared whether a simple average Word2Vec implementation or Word2Vec with a TFIDF vectorizer obtained a higher accuracy. They determined that the TFIDF outperformed the simple Word2Vec embedding with an accuracy of 74%, as compared to 65% on their best machine learning technique for a three-layer deep learning model (Kumar, Rajpal, & Rathore, 2018). The authors applied Word2Vec at a song level with Continuous Bag of Words rather than SkipGram. Other studies utilized a bag-of-words approach, with variations of processing techniques, such as stemming and not removing the stopwords (Mayer, Neumayer, & Rauber, 2008) (Liang, Gu, & O'Connor, 2011) (Dammann & Haugh, 2017). One limitation of the bag of words method is that it does not grasp the structural information of the songs, rather it may grasp only the topical content of the songs (Mayer, Neumayer, & Rauber, 2008).

2.4 Machine Learning Architectures

A majority of genre classification studies experiment with different machine learning methods and model architectures to determine which techniques can best predict what genre a song belongs to. Studies have utilized Naive Bayes, K-Nearest Neighbor, Support Vector Machines, Decision Tree Classifiers, Random Forest, eXtreme Gradient Boosting, Logistic Regression, and Deep Neural Networks, to varying degrees of success (Mayer, Neumayer, & Rauber, 2008) (Dammann & Haugh, 2017) (Kumar, Rajpal, & Rathore, 2018) (Liang, Gu, & O'Connor, 2011). Two different studies used a Long Short-Term Memory model and a Hierarchical Attention Network, respectively, to capture the word order and structure of the songs, not just the word content. The study which utilized the LSTM and bi-directional LSTM models classified songs into three genres and had a maximum accuracy of 68% (Boonyanit & Dahl). The study that pioneered the use of a HAN for this task, and is the only study to do so, classified 20 genres and determined that the LSTM outperformed the HAN, which had two layers, one for attention and the word level and the other for attention at the line/segment level. The LSTM had an accuracy of 49.77%, while the HAN had an accuracy of 49.50%, so the difference was not great (Tsaptsinos, 2017).

2.5 Research Gaps

There are significant opportunities for increased research, which culminated in this study. There is a lack of studies utilizing a non-bag-of-words approach to lyric classification, specifically regarding the analysis of the emotional content and sentiments of lyrics. The study that found that the sentiments of the lyrics did not improve the model's ability to classify a song into a genre used an un-intact bag of words corpus for their analysis (Liang, Gu, & O'Connor, 2011). More meaning sentimentally and content-wise can be derived from lyrics that are analyzed in regards to their structure, whether that be in song segments or song lines, rather than in a bag of words context. There is an opportunity to investigate whether sentiment analysis at a structural level of a song, such as a line level, could indicate song genre. Additionally, there is only one study regarding Hierarchical Attention Networks applied to song lyric context, and this study utilized GloVe embeddings. More can be discovered regarding the use of different types of embeddings, such as Word2Vec, in tandem with Long Short Term Memory models and Hierarchical Attention Networks. Lastly, there are no studies that combine the use of a model that analyzes/preserves word order and/or structure with the sentiment analysis of a song. This gap presents itself keenly for an investigation into using the LSTM model and HAN machine learning techniques on Word2Vec embeddings performed at the lyric level, and then in an ensemble model with the sentiment analysis of a song at the song and lyric levels to classify song genres. Will analyzing song lyrics at a line structural level by sentiment and content improve the ability to classify songs by genre as opposed to using general representations of song content and sentiment?

3. METHODOLOGY

3.1 Dataset

The dataset chosen for this project is open-source on the popular data-hosting platform, Kaggle. I chose this dataset due to its size and full lyrics, allowing the models to potentially learn from the structure of the lyrics[1]. The original dataset consisted of over two hundred thousand songs across ten genres: Rock, Metal, Pop, Indie, Folk, Electronic, R&B, Jazz, Hip-Hop, and Country. Other variables, Artist Name, Song Title, and Song Language, were not utilized in genre predictions.

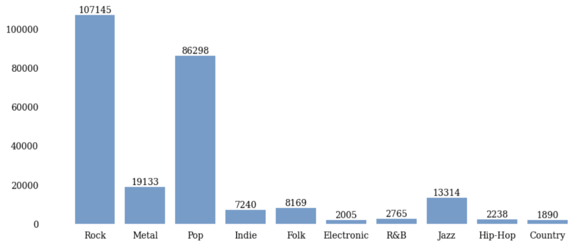


Figure One: Dataset Distribution of Song Genres

The number of songs per genre was disproportionate, as evidenced by Figure One above. Some data-cleaning procedures were undertaken before creating a training set of songs in which all genres were evenly represented. Firstly, only English songs were chosen. Several criteria led to the removal of portions of the raw song lyric data, including:

- Lyrical structure markers: [Verse 1:]
- Special characters and symbols that were substitutions for a quotation mark
- Introductory or post-song information: 'Lyrics By:', 'Music By:'
- Artist name or song title preceding the song
- Links to the source of lyrics

Additionally, songs with repetitive sections, such as repeated 'ohs' or 'ahs' followed by a number indicating how many times that particular lyric or word was repeated, were removed from the corpus entirely. Similarly, other songs had lyrical structure placements without lyrics in place, such as marking 'Chorus' where the lines for the chorus occur. The lack of lyrics in place of the structural or repetition markers posed a potential problem for classification, as the lyrics are not preserved in full for those songs. Lastly, one of each set of duplicate songs was removed.

Since the data was disproportionated per genre, I chose 1500 random songs from each genre to use in the project. Selecting a fraction of the total songs allowed for consistent representation of each category to ensure there were no classification biases based upon oversampling or undersampling, and the smaller set helped with decreasing the amount of time to run models later in the project.

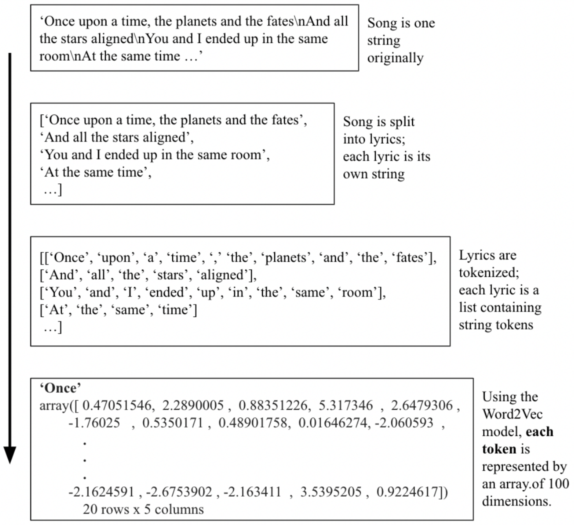
3.2 Embeddings

One consideration when utilizing word embeddings in machine learning models is the data that the embedding model has been trained upon. I chose to create a custom Word2Vec model on the entire song lyric corpus from the dataset because existing models are trained on data that may not include song lyrics. Since lyrical data is different in structure and content than other data, such as legal texts or Wikipedia pages, creating a custom Word2Vec model will likely provide a better understanding and representation of the lyrical content of the songs. The Word2Vec model allows words to be represented in a multidimensional space using

[1] https://www.kaggle.com/datasets/mateibejan/multilingual-lyrics-for-genre-classification

vectors that indicate how words are related to each other across the corpus.

To construct the Word2Vec model, I split each set of lyrics from its original state as a string into many lists, where each list is a sentence corresponding to a line. This line representation will be referred to throughout the paper as line, lyric, line-level, or lyric-level, depending on the context. Each line was tokenized. I chose to keep potential stop words and punctuation since punctuation could be indicative of genre, or of sentiment, such as with repeated exclamation points or other punctuation symbols. I trained and saved the custom Word2Vec embeddings, which have a standard vector dimension of 100 and a window size of ten (the average length of a lyric), to capture the words before and after each predicted word. These embeddings are numerical representations of words in relation to other words in the corpus, which in this context is the set of all the songs in the original dataset.



```
'Once upon a time, the planets and the fates\nAnd all
the stars aligned\nYou and I ended up in the same
room\nAt the same time …'
```

Song is one string originally

```
['Once upon a time, the planets and the fates',
'And all the stars aligned',
'You and I ended up in the same room',
'At the same time',
 …]
```

Song is split into lyrics; each lyric is its own string

```
[['Once', 'upon', 'a', 'time', ',' 'the', 'planets', 'and', 'the', 'fates'],
['And', 'all', 'the', 'stars', 'aligned'],
['You', 'and', 'I', 'ended', 'up', 'in', 'the', 'same', 'room'],
['At', 'the', 'same', 'time']
 …]
```

Lyrics are tokenized; each lyric is a list containing string tokens

```
'Once'
array([ 0.47051546,  2.2890005 ,  0.88351226,  5.317346  ,  2.6479306 ,
      -1.76025   ,  0.5350171 ,  0.48901758,  0.01646274, -2.060593  ,
        .
        .
        .
      -2.1624591 , -2.6753902 , -2.163411  ,  3.5395205 ,  0.9224617])
        20 rows x 5 columns
```

Using the Word2Vec model, **each token** is represented by an array.of 100 dimensions.

Figure Two: Flow Diagram of The First Verse of Taylor Swift's 'Mastermind'

I applied these embeddings to the data before being used in each of the content and ensemble models. Each song is split into individual lyrics, then tokens, and then embedding representations, as explained in Figure Two, above.

3.3 Lyric Sentiment

Song sentiment was analyzed in two ways to then be fed into models that used the sentiment to classify the songs into genres. To determine the sentiment of each song, I utilized the SentiStrength tool popular in poetry generation studies; the version I used was available through a Python wrapper class I downloaded from GitHub[2]. The SentiStrength function takes segments of text as input and assigns numerical representation to the emotionally charged words in the text, such as 'love' or 'family'. To allow for the greatest range of sentiments to be expressed, I utilized the 'scale' option, which meant that each word could receive a value ranging from -4, strongly negative, to 4, strongly positive. These values are then averaged across the text segment to determine the overall sentiment of the text segment, which is the output value of the function.

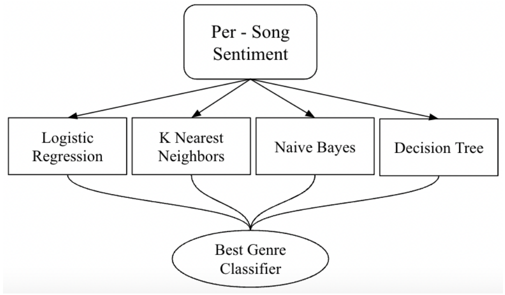


Figure Three: Per-Song Lyric Sentiment Classification Construction

At the song level, the entire song as one string was passed into the SentiStrength classifier, and a singular number was returned representing the sentiment of the overall song. Four models were used with this sentiment classification to predict song genre: Logistic Regression, K-Nearest Neighbors, Naive Bayes

[2] https://github.com/zhunhung/Python-SentiStrength

Classifier, and Decision Tree Classifier, as seen in Figure Three, above. These models were chosen because they are relatively simple models and the model input was a singular feature. It is likely that additional layers, such as in a Recurrent Neural Network, would not be beneficial due to the simplicity of the input for the song-level sentiment models.

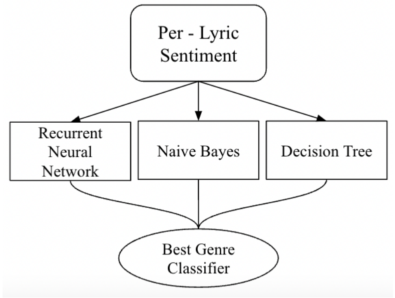


Figure Four: Per-Lyric Lyric Sentiment Classification Construction

At the lyric level, each line of the song was fed into the SentiStrength classifier and received a score. Then, the scores from each line were combined into a list representing the entire song while maintaining the order of the lines. Because some songs have fewer lines than others, the ends of many song-representative lists were padded with zeros up to the maximum length of all songs. Three models were chosen due to their ability to work with more complex features for lyric-level classification: Recurrent Neural Network, Naive Bayes Classifier, and Decision Tree Classifier.

3.4 Lyric Content

To determine whether lyric content may indicate genre, I trained three different models on song representations of line-level embeddings.

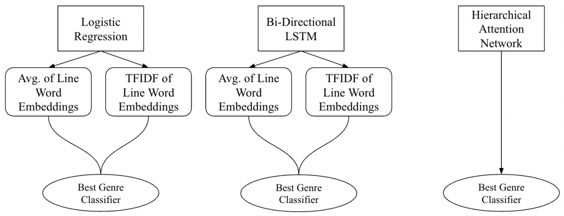
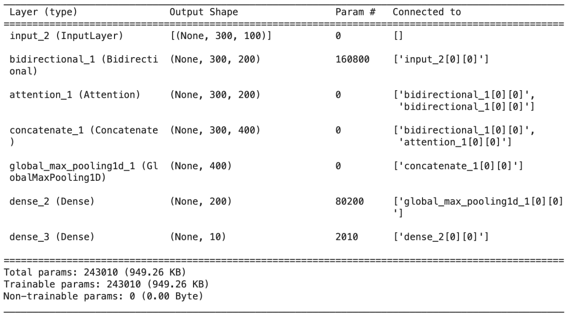

Logistic Regression
Avg. of Line Word Embeddings
TFIDF of Line Word Embeddings
Best Genre Classifier
Bi-Directional LSTM
Avg. of Line Word Embeddings
TFIDF of Line Word Embeddings
Best Genre Classifier
Hierarchical Attention Network
Best Genre Classifier

Figure Five: Lyric Content Classification Construction

I applied the custom Word2Vec model to each tokenized line to create the line-level word embeddings. To get a representation of the song for use in the classification models, the embeddings were into a song-level embeddings vector. Figure Five, above, indicates the flow of model creation; for the Logistic Regression (LR) and Bi-Directional Long Short Term Memory (Bi-LSTM) models, I compared the average of the sentence-level embeddings with the Term Frequency- Inverse Document Frequency (TF-IDF) weightings of the sentence-level embeddings. TF-IDF works by comparing the occurrence of a word in a singular song to how often it occurs across the entire song dataset.



```
 Layer (type)                Output Shape                 Param #   Connected to
==================================================================================================
 input_2 (InputLayer)        [(None, 300, 100)]           0         []

 bidirectional_1 (Bidirecti  (None, 300, 200)             160800    ['input_2[0][0]']
 onal)

 attention_1 (Attention)     (None, 300, 200)             0         ['bidirectional_1[0][0]',
                                                                     'bidirectional_1[0][0]']

 concatenate_1 (Concatenate  (None, 300, 400)             0         ['bidirectional_1[0][0]',
 )                                                                   'attention_1[0][0]']

 global_max_pooling1d_1 (Gl  (None, 400)                  0         ['concatenate_1[0][0]']
 obalMaxPooling1D)

 dense_2 (Dense)             (None, 200)                  80200     ['global_max_pooling1d_1[0][0]
                                                                    ']

 dense_3 (Dense)             (None, 10)                   2010      ['dense_2[0][0]']

==================================================================================================
Total params: 243010 (949.26 KB)
Trainable params: 243010 (949.26 KB)
Non-trainable params: 0 (0.00 Byte)
__________________________________________________________________________________________________
```

Figure Six: HAN Model Architecture

The three models analyzed were Logistic Regression, Bi-Directional LSTM, and Hierarchical Attention Network. Logistic Regression was chosen as a baseline performance metric, while Bi-LSTM and HAN were evaluated due to their ability to preserve the structure of lines and documents in various capacities, and were the main research gaps being addressed. Since the Bi-LSTM model can attend to words before and after the word that is currently being predicted, it has the potential to learn patterns in the structure of a song. The HAN model, being a hierarchical model, distinguishes various levels of documents, starting with the word, sentence, line, and then document level, and also has the potential to learn whether lyrical structure indicates song genre.

All of the models were trained using mini-batches of the data, splitting the training set of data into smaller segments to train and validate, until they achieved an optimal minimal validation loss. This ensured that the model was not overfitting to the training data and would still have strong predictions on the test data. The Logistic Regression models were trained for only two epochs, and no other hyperparameters were modified. The Bi-LSTM had an ADAM optimizer, categorical cross-entropy loss, and softmax activation for genre prediction, and was trained with a batch size of sixteen for fifty epochs. The HAN had an RMSprop optimizer, and categorical cross-entropy loss, and was trained with a batch size of sixteen and three epochs. The HAN accepted sentences with a maximum length of three hundred words, and the architecture is shown in Figure Six. Since HAN's internal architecture already provides attention to the sentence and then document level of a song, it was not necessary to use an average or TF-IDF weighting of the word embeddings to represent each song (Yang et al.).

3.5 Ensemble Models

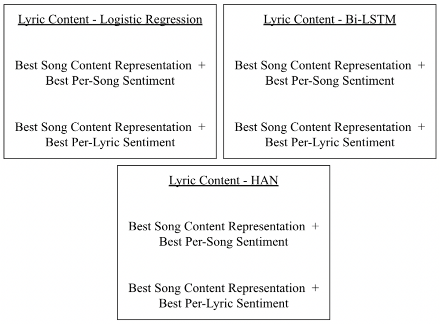

Lyric Content - Logistic Regression
Best Song Content Representation + Best Per-Song Sentiment
Best Song Content Representation + Best Per-Lyric Sentiment
Lyric Content - Bi-LSTM
Best Song Content Representation + Best Per-Song Sentiment
Best Song Content Representation + Best Per-Lyric Sentiment
Lyric Content - HAN
Best Song Content Representation + Best Per-Song Sentiment
Best Song Content Representation + Best Per-Lyric Sentiment

Figure Seven: Ensemble Models Construction

The goal is to determine if song content and song sentiment, which both preserve the structure of the song in their classification, can be used to classify songs into genres. I created six ensemble models that combined the best genre-classifying models from each of the base lyric content models (LR, Bi-LSTM, and HAN) with the best song-level and lyric-level sentiment models.

For reproducibility and consistency, the same 80/20 train-test split was used throughout all of the created independent and ensemble models, and the fitted models from the independent experiments were saved for use in the ensemble models. The ensemble models voted based on the weighted average of predicted class probabilities from both models, with equal consideration of the predictions of the sentiment and content models.

4. RESULTS

4.1 Independent Lyric Sentiment Models

The data was evenly sampled; there were 1200 songs per genre in the training set and 300 in the testing set. All visualizations depict the count of the songs predicted by the model for each genre.

The ensemble models are evaluated on two metrics: accuracy and confidence. Confidence is a custom metric on a scale of 0/10 to 10/10, where each point corresponds to how confident a model is in a particular genre. The model receives a point for a genre if the number of songs that it correctly classifies in that genre is greater than the number of songs it misclassifies as any other genre.

4.1.1 Song-Level Lyric Sentiment

All of the song-level lyric sentiment models performed similarly poorly, only slightly above how they would perform by random chance. The models had low accuracy and low confidence in the predictions.

Model Type	**Accuracies**	**Confidence**
Logistic Regression	15.20%	2/10
K- Nearest Neighbors	15.30%	2/10
Naive Bayes	15.10%	2/10
Decision Tree	15.30%	2/10

Table One: Lyric Sentiment, Song-Level Model Performance

The best model was the Decision Tree Classifier, with an accuracy of 15.3%, however, it only predicted five out of the ten genres.

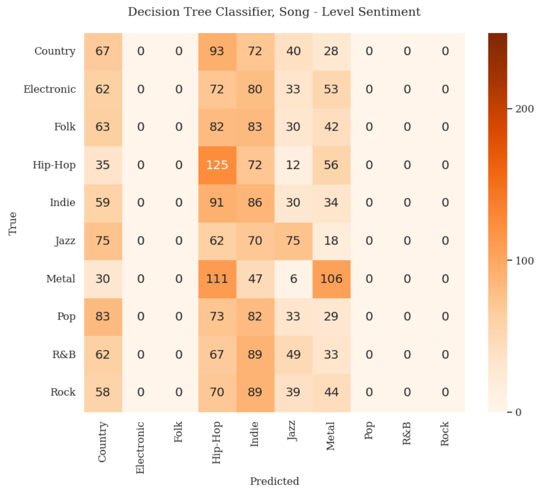


Figure Eight: Lyric Sentiment Confusion Matrix for Best Song-Level Sentiment Model

4.1.1 Lyric-Level Lyric Sentiment

Model Type	**Accuracies**	**Confidence**
Recurrent Neural Network	28.53%	4/10
Naive Bayes	25.13%	5/10
Decision Tree	24.53%	4/10

Table Two: Lyric Sentiment, Lyric-Level, Model Performance

The lyric-level sentiment models performed significantly better than the song-level models, with nearly double the accuracy for the best performing Recurrent Neural Network, though the confidence of the sentiment models remained low.

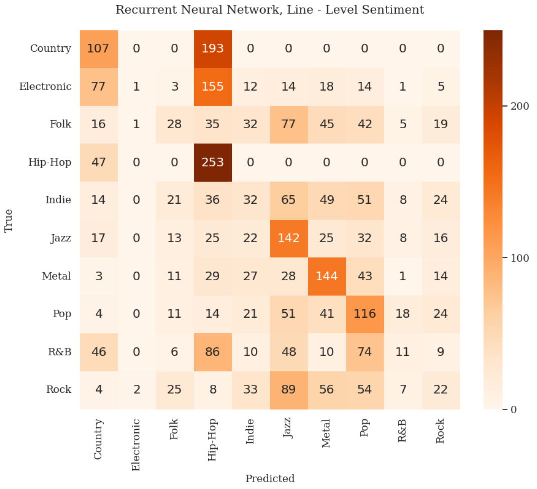


Figure Nine: Lyric Sentiment Confusion Matrix for Best Lyric-Level Sentiment Model

The RNN performed best in the Hip-Hop genre, like the previous sentiment models, with an 84.33% accuracy in correctly classifying Hip-Hop songs. It performed extremely poorly in the Electronic and R&B categories.

4.2 Independent Lyric Content Models

Content Model	Average of Line Embeddings	TF-IDF of Line Embeddings
Logistic Regression	12.90%	36.33%
Bi-Directional LSTM	31.60%	24.83%
Hierarchical Attention Network	38.60%	

Table Three: Lyric Content Model Performance, Accuracy

Content Model	Average of Line Embeddings	TF-IDF of Line Embeddings
Logistic Regression	2/10	8/10
Bi-Directional LSTM	7/10	5/10
Hierarchical Attention Network	7/10	

Table Four: Lyric Content Model Performance, Confidence

The highest-performing content model in terms of accuracy was the Hierarchical Attention Network, with an accuracy of 38.6 percent. The model with the highest confidence in its predictions, correctly predicting a majority for eight out of ten genres, was the Logistic Regression with TF-IDF weighted embeddings. This model was able to obtain more correct classifications for Folk than misclassifications of any other genre, which no other content model achieved. For the Logistic Regression model, using TF-IDF weightings of the line-level embeddings led to improved classification over a simple average, while for the Bi-Directional LSTM, the average was a better classifier than the TF-IDF weightings in both accuracy and confidence.

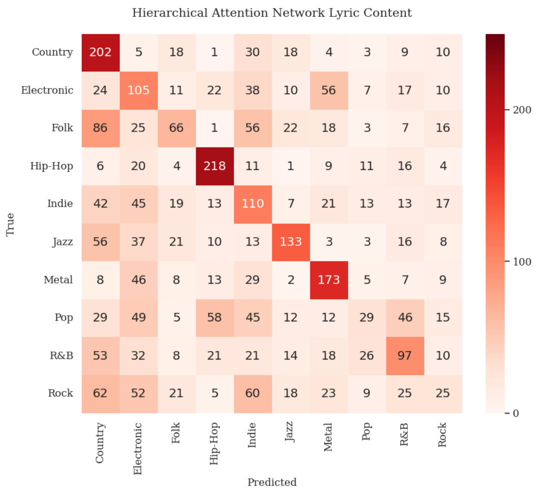


Figure Ten: Hierarchical Attention Network Confusion Matrix For Best Content Model

Across all five models, the genre with the highest accuracy was Hip-Hop, and the genre with the lowest accuracy was Rock The best content model for each model type also performed moderately well in classifying country, metal, and jazz songs. The confusion matrices for the best content models for Logistic Regression (TF-IDF weighting of line embeddings) and Bi-LSTM (Average of line embeddings) can be found in the appendix.

4.3 Ensemble Models

Using the weaker song-level sentiment model in combination with the content models did not improve the ability to classify song genre.

Model Type	Best Content Only	Ensemble w/ Song - Level Sentiment	Ensemble w/ Line - Level Sentiment
Logistic Regression	36.33%	25.46%	25.46%
Bi-LSTM	31.60%	27.57%	33.23%
HAN	38.60%	38.60%	43.07%

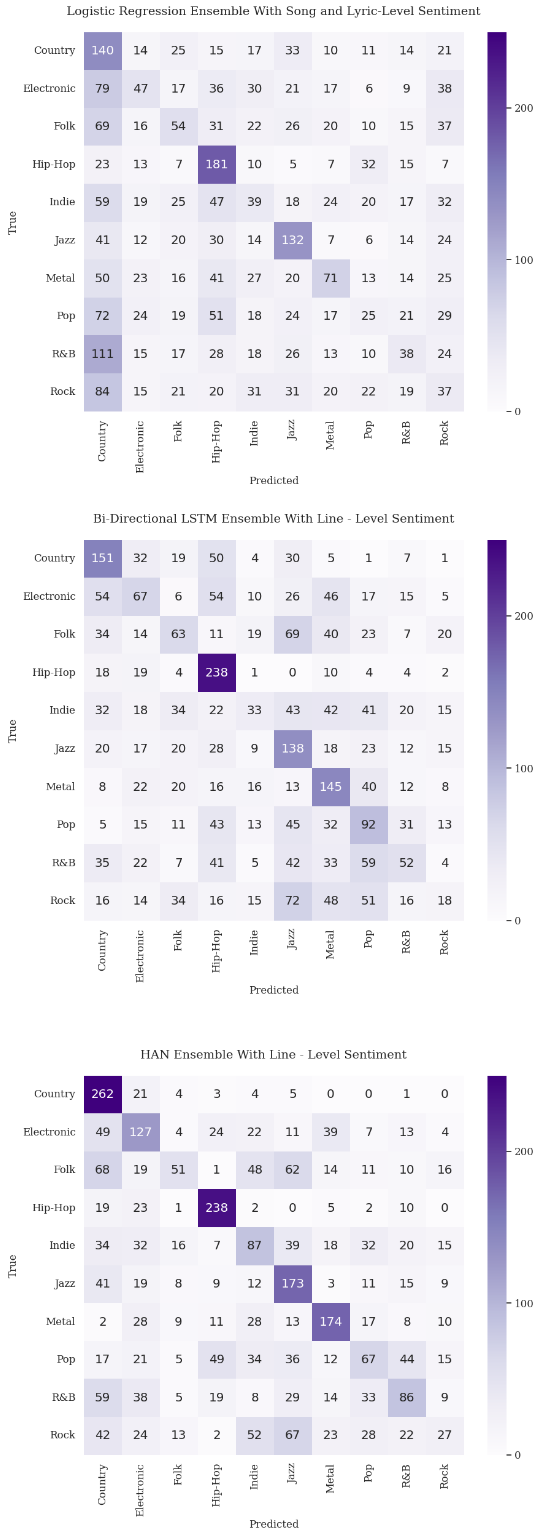
Table Five: Ensemble Models' Performance, Accuracy

Model Type	Best Content Only	Ensemble w/ Song - Level Sentiment	Ensemble w/ Line - Level Sentiment
Logistic Regression	8/10	4/10	4/10
Bi-LSTM	7/10	7/10	6/10
HAN	7/10	7/10	8/10

Table Six: Ensemble Models' Performance, Confidence

For the Logistic Regression models, using the sentiment classification decreased the accuracy of the ensemble model, while the ensemble of Bi-LSTM and line-level RNN saw a slight improvement. The HAN with line-level RNN together had a significant improvement and was the best-performing model across all independent and ensemble models, indicating that combining sentiment analysis with content analysis of lyrics can improve genre classification.

Notably, the HAN and RNN ensemble model was the only model to correctly predict more Country songs than any other genre, with an 87.33% accuracy in predicting Country songs, the highest from any genre in any model. It also performed moderately well on some of the more difficult-to-classify genres, based on the poor accuracy of many models, such as Electronic and Indie.



Logistic Regression Ensemble With Song and Lyric-Level Sentiment

True \ Predicted	Country	Electronic	Folk	Hip-Hop	Indie	Jazz	Metal	Pop	R&B	Rock
Country	140	14	25	15	17	33	10	11	14	21
Electronic	79	47	17	36	30	21	17	6	9	38
Folk	69	16	54	31	22	26	20	10	15	37
Hip-Hop	23	13	7	181	10	5	7	32	15	7
Indie	59	19	25	47	39	18	24	20	17	32
Jazz	41	12	20	30	14	132	7	6	14	24
Metal	50	23	16	41	27	20	71	13	14	25
Pop	72	24	19	51	18	24	17	25	21	29
R&B	111	15	17	28	18	26	13	10	38	24
Rock	84	15	21	20	31	31	20	22	19	37

Bi-Directional LSTM Ensemble With Line - Level Sentiment

True \ Predicted	Country	Electronic	Folk	Hip-Hop	Indie	Jazz	Metal	Pop	R&B	Rock
Country	151	32	19	50	4	30	5	1	7	1
Electronic	54	67	6	54	10	26	46	17	15	5
Folk	34	14	63	11	19	69	40	23	7	20
Hip-Hop	18	19	4	238	1	0	10	4	4	2
Indie	32	18	34	22	33	43	42	41	20	15
Jazz	20	17	20	28	9	138	18	23	12	15
Metal	8	22	20	16	16	13	145	40	12	8
Pop	5	15	11	43	13	45	32	92	31	13
R&B	35	22	7	41	5	42	33	59	52	4
Rock	16	14	34	16	15	72	48	51	16	18

HAN Ensemble With Line - Level Sentiment

True \ Predicted	Country	Electronic	Folk	Hip-Hop	Indie	Jazz	Metal	Pop	R&B	Rock
Country	262	21	4	3	4	5	0	0	1	0
Electronic	49	127	4	24	22	11	39	7	13	4
Folk	68	19	51	1	48	62	14	11	10	16
Hip-Hop	19	23	1	238	2	0	5	2	10	0
Indie	34	32	16	7	87	39	18	32	20	15
Jazz	41	19	8	9	12	173	3	11	15	9
Metal	2	28	9	11	28	13	174	17	8	10
Pop	17	21	5	49	34	36	12	67	44	15
R&B	59	38	5	19	8	29	14	33	86	9
Rock	42	24	13	2	52	67	23	28	22	27

Figure Eleven: Best Performing Ensemble Model for each Base Content Model Combination

5. DISCUSSION

5.1 Lyric Sentiment

The poor performance, in accuracy and confidence, of the song-level sentiment models indicates that the data form of one number representing the sentiment of each song does not provide enough information to guide a model in identifying a song's genre. The models all performed similarly, with around 15 percent accuracy and a 2/10 confidence level. The model performed only slightly better than guessing and only five of the ten genres were even predicted by the model. Sentiment at the song level, with no further information, is not helpful in categorizing songs into genres.

All of the lyric-level sentiment models saw an increased accuracy, nearly double that of the song-level models. Thus, the sentiment of a song, at a line structural level, can be used to classify songs into genres. The increased accuracy when considering line versus song level sentiment brings to light the possibility of the structure of the song being a factor in its genre classification. The lyric-level models, specifically the most accurate RNN, were biased towards five genres, consistently overpredicting for several genres, including hip-hop. It was unable to generalize and was nearly blind to some genres, such as Electronic. The low confidence score of the RNN model indicates that the model was unable to learn features that indicate more nuanced genres like Folk and Rock, or that those genres do not have considerable sentiment indicators.

5.2 Lyric Content

While with lyric sentiment the accuracy increased to a similar performance across all models when considering the line level, with lyric content, the performance depended on the model and embedding methodology. The HAN model had the highest accuracy and confidence and was able to identify and learn from a majority of genres, likely due to the ability to attend to the word, level, and document-level structure of the data. The lyric content models demonstrate that it is important to consider the word content of the songs in comparison to the larger corpus; using TF-IDF weightings with the Logistic Regression model almost tripled its accuracy as compared to the average embeddings model. It was surprising that the Logistic Regression model performed with the second highest accuracy for the content-only models and with the highest confidence, considering its weak performance in other similar studies. Interestingly, using TF-IDF embeddings with the Bi-LSTM model decreased the accuracy of the model as compared to the average embeddings; the TF-IDF may be providing extraneous information that the forward and backward-looking nature of the Bi-LSTM model does not find helpful in determining song genre.

5.3 Ensemble

Pairing the sentiment models with the Logistic Regression content model led to a decrease in accuracy and confidence for both types of sentiment analysis. This indicates that the poor voiting of the sentiment models actually outweighed the better voting of the Logistic Regression model for many songs. This doesn't mean that using sentiment in tandem with content makes it more difficult to classify songs into genres, rather, it indicates that the ensemble structure is not optimized to exhibit the strengths of both models for a greater overall accuracy.

The Bi-LSTM ensemble models had mixed results, with increased accuracy and decreased confidence for the ensemble with lyric-level sentiment, and a decreased accuracy but increased confidence for the ensemble with song-level sentiment. The lack of consistent poor or good performance was surprising considering it's great performance in previous bag of words studies, even outperforming HAN models.

Combining the HAN model with song-level sentiment led to no improvement in classification accuracy. This is supported by the higher confidence of the HAN content model than the DTC sentiment model. However, the line-level HAN ensemble model had meaningful increases in accuracy and confidence. It was able to classify songs and receive confidence points in the Indie and R&B genres, which previous models struggled on. The performance of the HAN line-level ensemble model clearly indicates the potential of utilizing song lyric data on a structural level to classify songs into genres.

5.4 Implications

The goal of the study was to determine if analyzing song lyrics at a structural level, through sentiment and content, can aid in the classification of songs into genres, and experiment with several different model architectures. The results from the HAN content model, which considers the song content at many levels, and the HAN line-level ensemble model illuminate that researcher smust consider how to optimally pair initial data, embeddings, and model architectures when solving the song genre classification problem, and other NLP-related problems. Since the structure of the data, song lyrics in verses, chorus, and lines, was able to be understood by the HAN model, it was likely the contributing to the greater performance. It is necessary to match the complexity of the model to the complexity of the data.

The results are promising; there is reason to believe that song genre can be determined through lyrical patterns, whether that is lyric content, structure, sentiment, or some combination of several lyrical factors. This opens avenues for increased research into what these structural factors are, as well as practical implementations of song lyrics into music recommendation systems.

Though ultimately many of the models performed well, they consisentely overpredicted for genres like Country and Rap, and performed poorly on more nuanced genres, such as Electronic, Folk, and Pop. Notably, throughout the models, many Folk songs were misclassified as Jazz, but Jazz songs were not as frequently misclassified as Folk. There are many relationships between genres that require additional exploratory analysis, such as visualizing embeddings, to determine possible reasons for model performance on these classifications.

Future research can focus on other model architectures that may be able to understand the structure of song data, as well as increased feature engineering and fine-tuning to increase the performance of the HAN and RNN ensemble models, such as creating a more layered neural network and experimentation with activation and optimization functions. Additionally, there is an exploratory investigation that can be done into which words or phrases are the strongest indicators of song genre and where in the song these words occur, to determine if there are distinct patterns in songs of different genres that the model may be picking up on. Lastly, it is worth delving into why some genres, such as Rock, Indie, and Pop, are harder to classify than others; possible factors could be similar lyric structure or content between those genres.

6. CONCLUSION

Utilizing song lyrics on a structural level represented through lyric-level sentiment and content analysis aids in the classification of songs into genres. Exploring the relationship between words in songs at the word, line, and song level, as well as comparison to the corpus, allowed the best performing HAN model to discover trends and similarities within different genres. The results illuminate the need to ensure the compatibility of models with the data structures they are working with; models must be able to utilize the various layers of information or features of the data.

REFERENCES

Boonyanit, A., & Dahl, A. (n.d.). *Music Genre Classification using Song Lyrics*. Stanford CS224N Course. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1214/reports/final_reports/report003.pdf.

Dammann, T., & Haugh, K. (2017). Genre Classification of Spotify Songs using Lyrics, Audio Previews, and Album Artwork. *Stanford CS229 Course*. https://cs229.stanford.edu/proj2017/final-reports/5242682.pdf.

Kumar, A., Rajpal, A., & Rathore, D. (2018). Genre classification using word embeddings and deep learning. *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. https://doi.org/10.1109/icacci.2018.8554816

Liang, D., Gu, H., & O’Connor, B. (2011). Music Genre Classification with the Million Song Dataset. *Carnegie Mellon 15-826 Course.* https://www.ee.columbia.edu/~dliang/files/FINAL.pdf.

Mayer, R., Neumayer, R., & Rauber, A. (2008). Rhyme and Style Features for Musical Genre Classification by Song Lyrics.

ISMIR 2008 - 9th International Conference on Music Information Retrieval. 337-342.

McVicar, M., Di Giorgi, B., Dundar, B., & Mauch, M. (2021, November 29). Lyric document embeddings for music tagging. https://arxiv.org/pdf/2112.11436.pdf

Misztal-Radecka, J., & Indurkhya, B. (2014). Poetry generation system with an emotional personality. *International Conference on Innovative Computing and Cloud Computing. https://computationalcreativity.net/iccc2014/wp-content/uploads/2014/06/6.3_Misztal.pdf*

Satrio Utomo, T., Sarno R., & Suhariyanto. (2018). Emotion Label from ANEW Dataset for Searching Best Definition from WordNet. *2018 International Seminar on Application for Technology of Information and Communication, Semarang, Indonesia.* 10.1109/ISEMANTIC.2018.8549769.

Tsaptsinos, A. (2017, July 15). Lyrics-based music genre classification using a hierarchical attention network. https://arxiv.org/abs/1707.04678

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (n.d.). Hierarchical Attention Networks for Document Classification. Carnegie Mellon University. https://www.cs.cmu.edu/~./hovy/papers/16HLT-hierarchical-attention-networks.pdf.

APPENDIX

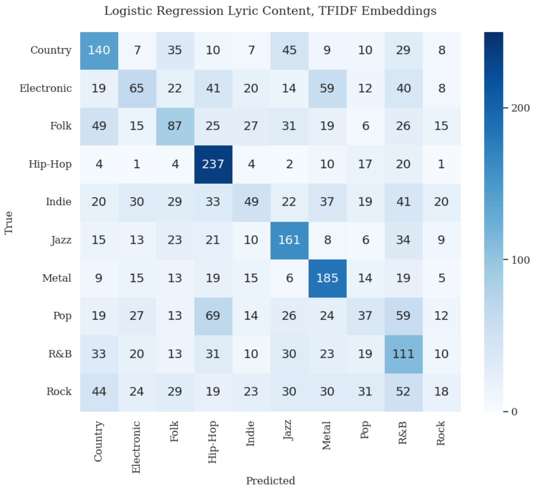


Figure Twelve: Logistic Regression Confusion Matrix For Best Content Model

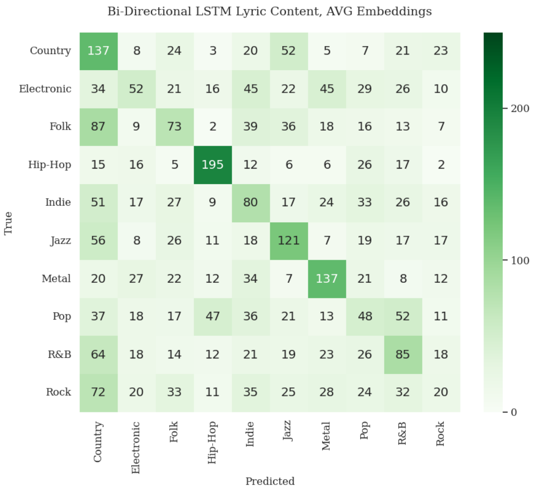


Figure Thirteen: Bi-Directional LSTM Confusion Matrix For Best Content Model

Model Category	**Model**	**Accuracy**	**Confidence**
Song-Level Lyric Sentiment	Logistic Regression	15.20%	2/10
	K- Nearest Neighbors	15.30%	2/10
	Naive Bayes	15.10%	2/10
	Decision Tree	15.30%	2/10
Lyric-Level Lyric Sentiment	Recurrent Neural Network	28.53%	4/10
	Naive Bayes	25.13%	5/10
	Decision Tree	24.53%	4/10
Lyric Content	Logistic Regression: Average of Line Embeddings	12.90%	2/10
	Logistic Regression: TF-IDF of Line Embeddings	36.33%	8/10
	Bi-Directional LSTM: Average of Line Embeddings	31.60%	7/10
	Bi-Directional LSTM: TF-IDF of Line Embeddings	24.83%	5/10
	Hierarchical Attention Network	38.60%	7/10
Ensemble With Song-Level (DTC) Sentiment	Logistic Regression + Sentiment	25.46%	4/10
	Bi-Directional LSTM + Sentiment	27.57%	7/10
	Hierarchical Attention Network + Sentiment	38.60%	7/10
Ensemble With Lyric-Level (RNN) Sentiment	Logistic Regression + Sentiment	25.46%	4/10
	Bi-Directional LSTM + Sentiment	33.23%	6/10
	Hierarchical Attention Network + Sentiment	43.07%	8/10

Table Seven: All Models' Accuracies and Confidences