Popular Words Extraction for US Hit Song Lyrics

Zhixiao Xiong, Zhiyu Xu, Cenhao Zhu

Abstract

Pop culture is an indispensable part of modern life. This research seeks to answer the problem of what makes a particular era distinctive and which words encapsulate that period. An analysis is conducted on the top 100 most popular songs in the US, as ranked by Billboard, from 1959 to 2022. The study employs interpretable classification, specifically logistic regression, for popular words extraction. Feature engineering is conducted to aid classification, and the validity of bag of words assumption is confirmed by stronger classifiers. Various algorithms (such as ETM and BERTopic) are performed to divide time into time periods, but it turns out unsuccessful. Then, the study resorts to decades division that is supported by several clustering learning algorithms as well as one-against-one logistic regression result. The finding indicates that Cold War is the most prominent milestone in pop culture, and each decade is marked by milestones like war and movements. The paper concludes on potential further improvement and refinement of the project.

1 Introduction

Pop culture is an indispensable part of modern life. Slogans are on billboards, TVs and everywhere. Concerning the development of pop culture over time, some argue that "fashion trends are cyclical", while others believe each era possesses distinctive features. Furthermore, if some distinguished features mark a certain time, we want to investigate into what plays the most important role as the distinguished features, and what words specifically mark this time. Hence, song lyrics, a very important form of pop culture, is adopted. The goal of this project is interpretable classification of lyrics into a certain time period, which enables the extraction of popular words.

As a form of natural language, song lyrics are intrinsically hard to classify, especially when interpretability is a vital aspect. The main challenges are as follows:

1. Feasibility of lyrics for time period classification.
2. Proper division of time periods.
3. Selection of relevant features for classification and identification of additional factors for improving prediction accuracy.
4. Suitability of the Bag of Words (BoW) assumption.

The four points will be investigated respectively in the following sections.

Accurate classification relies on suitable time period division. Lacking the expertise required for pop culture time period division, unsupervised learning methods that consider temporal trends are favored. For natural language processing, topic models, such as probabilistic Latent Semantic Indexing (pLSI) [Hof], Latent Dirichlet Allocation (LDA) [BNJ], are developed for clustering documents into groups with similar topics. Based on LDA, Dynamic Topic Models (DTM) [BL] is able to observe how topics

change over time. Topic-Score is a recently-developed topic modeling method, which utilizes pre-SVD and pro-SVD to achieve identifiabiity and uniqueness of the final result[KW22]. Beyond BoW, Embedded Topic Models (ETM) [DRBb] incorporates context information via word embedding models like Word2Vec [MCCD] and by projecting topics into the embedding space, it has the ability to take into account of correlation between topics. Its dynamic version has been proposed as well [DRBa]. Topic models also incorporate BERT (Bidirectional Encoder Representations from Transformers) to have stronger language processing ability, which also has a dynamic variant[Gro22]. Due to the large number of features and their unbalance, it is difficult to yield ideal period division. By performing regression with each song's year as label via neural network, information can be compressed into low dimension balance representation. Clustering based on this low dimension representation may yield reasonable period division.

However, it is possible that no time period division is satisfactory. In this case, an intuitive approach is to directly use decades as period division rule. Still, how to combine decades is another issue. In order to establish reasonable time period division based on decades, several unsupervised learning algorithms can be helpful. Within the scope of BoW, clustering methods are well-known for its ability to discovering patterns, relationships, and structures in data. Under the umbrella of BoW assumption, various clustering methods: Principal Component Analysis (PCA) [F.R01], t-Distributed Stochastic Neighbor Embedding (t-SNE) [vdMH08], Uniform Manifold Approximation and Projection (UMAP) [MHM20] and Locally Linear Embedding (LLE) [GGKC20], etc. can be introduced to discover previously unknown inherent patterns or relationships, offering insight for time division, feature selections and future supervised learning steps.

Previous research on lyrics classification mainly focuses on sentiment analysis and related tasks. [Joa] introduces text classification via support vector machines (SVM). Transductive Support Vector Machine (TSVM) is suggested for unbalanced and small datasets. Regarding the unique characteristics of lyrics, [XWW08] proposes sentiment vector space model (s-VSM) to address ad hoc characteristics of lyrics, including focus on sentiment words, negations, and modifiers. [DS] suggests a weighting strategy for features. [Bah] presents music genre classification using audio through deep neural networks. The structure of lyrics resembles poems. [JMN12] suggests extracting features of number of lines in the document, number of words in the document, average number of words per line, number of paragraphs, number of words per paragraph etc. Similarly, [TSD08] suggests poems in pattern recognition should recognize features like rhyme, meter, shape, meaning. [KM] suggests using gain ratio to select features.

For identification of popular words, logistic regression with penalty is an intuitive approach. The coefficient of each word, after scaling, directly indicates its importance. However, note that the label is ordered, where ordered logistic regression with penalty fails. Therefore, another perspective is to treat it as a multi-label problem with label correlation. CorrLog [BXT12] has been proposed for this specific scenario.

The project is structured as follows: Section 2 introduces data collection, cleaning and feature selection. This section also involves training classifiers to demonstrate the predictive capability of the features. Section 3 performs EDA. Section 4 outlines our efforts to divide time periods through unsupervised learning and neural networks. Section 5 employs logistic regression to extract popular words of each period. Finally, the paper concludes with a discussion of the limitations of the current project and proposes future directions for improvement.

2 Data Preparation and Feature Selection

2.1 Data Collection and Cleaning

To ensure the popularity of the selected songs, the songs are chosen from Billboard Top 100 Year-End List in the US from 1959 to 2022. The title of each song is obtained on corresponding Wikipedia pages, as well as information about the artists. Lyrics are collected from Genius, a popular lyrics website. From literature review, previous researches have demonstrated the effectiveness of classifying artists and genres.This data collection pipeline avoids this undesirable property.

Following the standard approach of cleaning natural language data, we begin by removing stop words and non-English words. (Here we use the stop-words dictionary in packages "snowball" and "tidytext" in R, as well as python pacakge "nltk"). Additionally, we eliminate words that appear fewer than 20 times across all songs. In consideration of the unique characteristics of lyrics, we further remove words with fewer than three letters, as these short words (such as "to," "ha," and "is") hold little semantic meaning. To capture contextual information, we combine words into bi-grams using the "nltk" package in Python. Finally, we perform TF-IDF (Term Frequency-Inverse Document Frequency) analysis to assess the importance of words or phrases within the lyrics' dataset.

For the purpose of this research, specific cleaning procedures are implemented to accommodate the unique characteristics of song lyrics. First, lemmatization is not performed, as people's usage of a particular word strongly reflects the time period. To ensure interpretability, non-interpretable words such as prepositions, conjunctions, and common names are removed. Regarding common names, we compile a dataset from the US Statistics Bureau containing the frequencies of names given to newborn babies, and select the top 1000 popular names. Furthermore, to avoid mistakenly deleting essential feature words and to ensure thorough cleaning, we conduct a manual screening of all 6000+ words. For example, meaningless interjections like 'hahaha' and 'balabala' that are not included in any dictionaries and were not removed during previous steps are eliminated after a manual check. However, certain common names like Heaven, Hope, etc., which hold other important meanings, are added back in this final step.

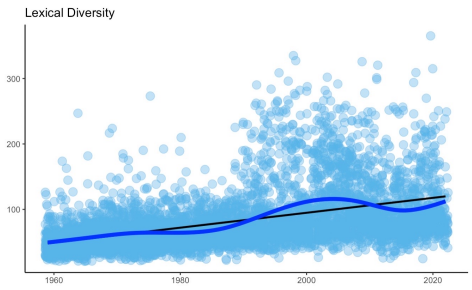
The data is then split into training set and test set randomly. Dictionary is built based on training set. The following table shows balance in training set and test set after splitting. After data cleaning, there are 5085 samples left with more than 3560 unique words.

	60s	70s	80s	90s	00s	10s
Training Set	844	778	792	731	725	789
Test Set	81	85	69	78	68	85

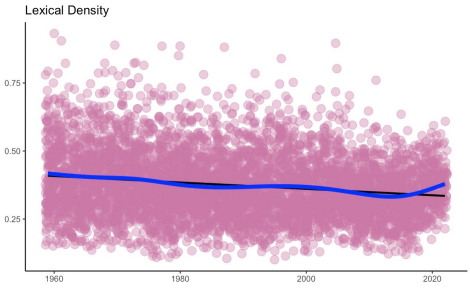
Table 1: Balance of labels in training and test sets

2.2 Feature Engineering

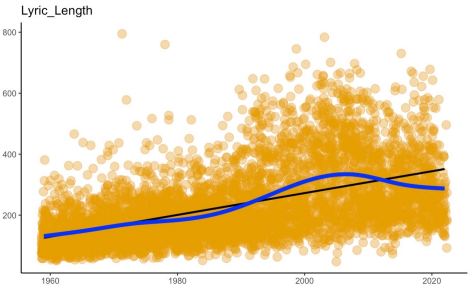
Upon observation, we identify 4 potential features that might be useful: average length of words in a song (average word length), length of a song (lyric length), number of unique words in a song (lexical diversity), ratio of unique words to total number of words (lexical density). In order to visualize the relationship between time and these features, plots are presented in Figure 1.



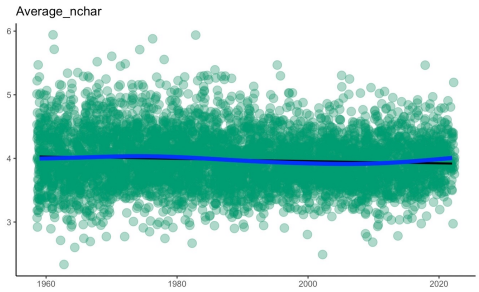
(a) Number of unique words



(b) Number of unique words averaged over number of words



(c) Number of words



(d) Average length of words

Figure 1: Four features from observation

From the results of the scatter plot in Figure 1, it is evident that the lyric length and lexical diversity have consistently increased over the past six decades. This indicates a trend of songs becoming longer over time. Conversely, the lexical density shows a declining pattern, suggesting that more repetitive words or paragraphs are being used in recent lyrics. Consequently, we have incorporated these three features into our model, while excluding average word length, as it remains constant throughout the entire time period. Additionally, we have included other relevant factors based on a literature review, such as the artist's active years, number of lines, number of paragraphs, average words per line, and average words per paragraph. Furthermore, we have selected 13 genres that appear more than 200 times as dummy variables: Alternative, Country, Dance, Disco, Folk, Funk, Hip-hop, New wave, Pop, R&B, Rap, Rock, and Soul. These features, along with the 3560+ unique words, serve as covariates in our model.

2.3 Feasibility of Features

In this section, we show that these features are competent enough for classification purpose. This section sets error rate as metric. Period division of 6, 4, 2 classes will be illustrated in the following sections.

SVM with Gaussian kernel has poor accuracy. To overcome the BoW constraint, different types of word embedding, including FastText and Word2Vec, are performed and combined with SVM, without substantial improvement. Embedding dimensions ranging from 10 to 2000 have been tested. It is observed that mild improvements take place with the increase of embedding dimension, but still not far from satisfactory. The result for 500 embedding dimensions is displayed in the following table. XGBoost shows satisfactory prediction accuracy.

Accuracy	SVM	Embedding SVM	XGBoost
6 classes	41.63%	42.70%	69.52%
4 classes	59.01%	62.45%	85.41%
2 classes	82.18%	82.40%	95.71%

Table 2: Result of three classifiers

Poor performance of SVM may be due to the sparse structure. Combination of features performed by kernel is meaningless, and often yield 0. Therefore, as can be seen later, its performance is similar to logistic regression. The failure of embedding may be due to the lack of logic in lyrics. The classification result of XGBoost shows that BoW assumption is reasonable and satisfactory classification results can be achieved.

3 EDA

Before applying clustering and classification methods to analyze the lyrics, we conduct Exploratory Data Analysis (EDA) to gain a comprehensive understanding of the lyrics. Initially, we use the cleaned data set to explore what words remain popular over times. Figure 2 shows the word cloud of words used in all songs. Notably, commonly heard words such as "love," "time," "world," "heart," and "baby," among others, constitute a significant portion of the word cloud. This observation provides us with a holistic overview and reinforces the effectiveness and rationality of our data cleaning process.

Figure 2: Word-Cloud of all songs

We are also curious about the change of frequent words over time. So we set an arbitrary time division of decades here, and generate a bar graph illustrating the most frequent 10 words in each of the 6 decades, as depicted in Figure 3. Remarkably, enduring words such as "love," "time," and "baby," among others, consistently rank among the top 10 words in each decade. Additionally, we have observed a fascinating trend: leading up to the 2010s, the top words predominantly consisted of everyday language. However, in the 2010s, the inclusion of the word "shit" as the ninth most frequently

used word suggests a significant surge in the use of vulgar language in recent years.

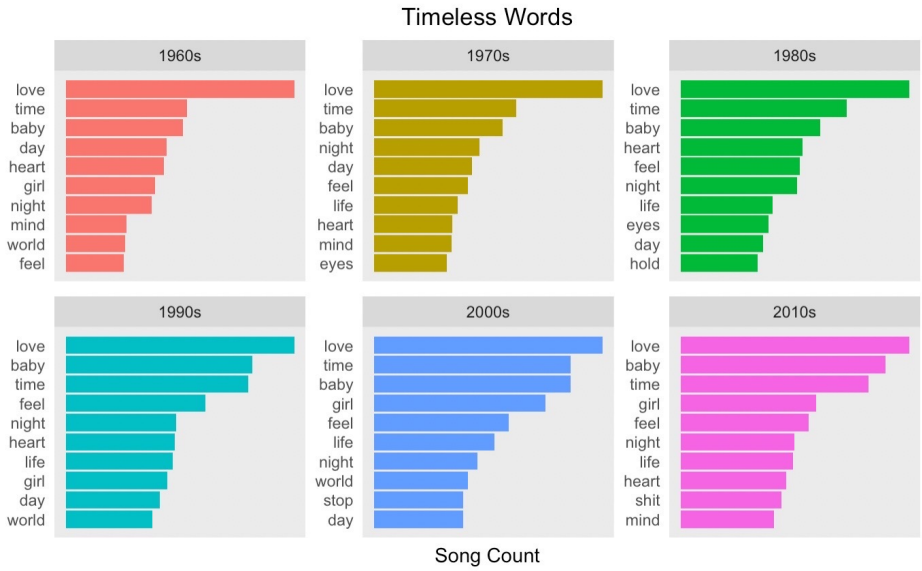


Figure 3: the most frequent 10 words in each decade

Due to the prevalence of timeless words, which overshadow changes in word usage patterns, we employ TF-IDF conversion based on decades to analyze the count data. The converted results are depicted in Figure 4, illustrating a noticeable transformation. Prior to the 1990s, lyrics predominantly consisted of formal and written language, whereas after this period, there is a noticeable prominence of vulgar and colloquial expressions. This shift can be attributed to the historical context, particularly the end of the Cold War, which likely played a significant role in this dramatic change.

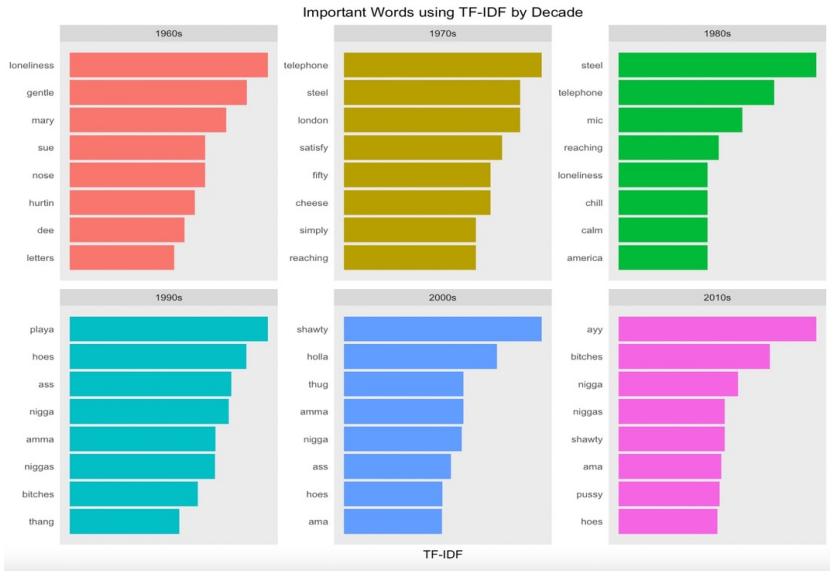


Figure 4: the most representative 10 words in each decade (TF-IDF)

4 Time Period Division

4.1 Flexible Time Division Attempts

By tuning parameters, it is shown that LDA and DTM are very unstable, thus abandoned. Currently, ETM cannot be implemented together with TF-IDF. Therefore, it has the tendency that all the topics are composite of the most frequent words, as shown below. In this case, this section introduces the results of stronger models, including BERTopic, BERT, Multi-Layer Perceptrons (MLP).

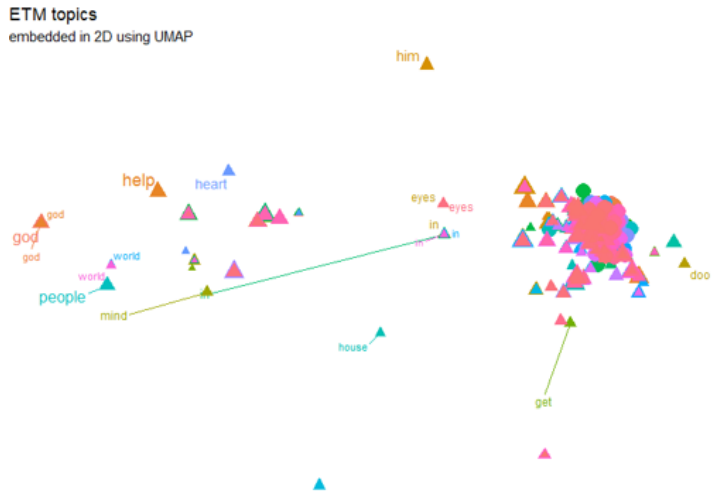


Figure 5: The result of ETM

4.1.1 BERTopic Model

BERTopic [Gro22] is adopted as an integrated framework for analyzing embedded topics and their trends over time in lyrics. BERTopic is a topic modeling framework that leverages BERT embeddings and c-TF-IDF to create dense clusters, allowing for easily interpretable topics whilst keeping important words in the topic descriptions. It is also scalable, as it leverages HDBSCAN to reduce the number of candidate topic words with each iteration. The structure of BERTopic is shown in 6.

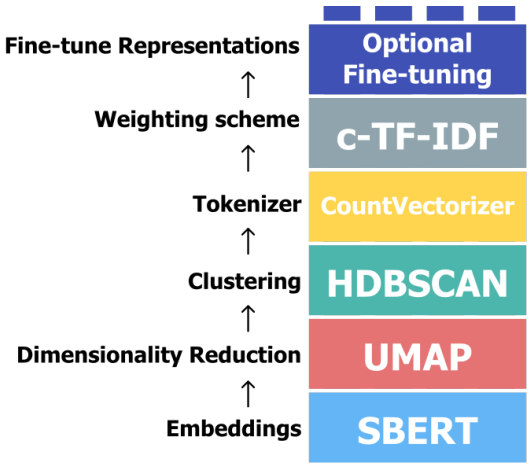


Figure 6: BERTopic Structure

We have experimented with 486 combinations of hyperparameters and discovered that the result is highly sensitive to the hyperparameters. Due to limited time and computation resources, we have not been able to find the best hyperparameters. The following hyperparameters are chosen. The result of BERTopic is shown in Figure 7 and Figure 8.

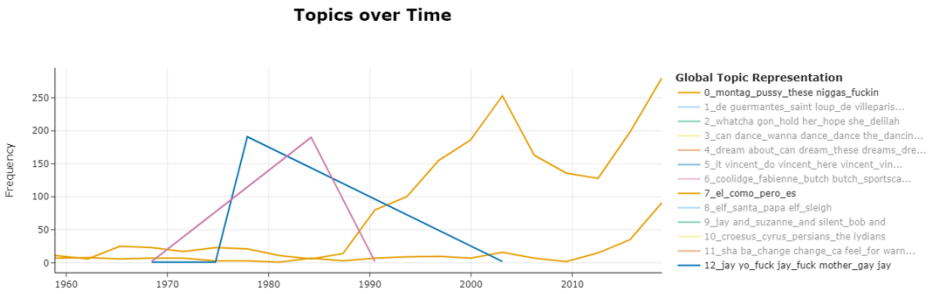


Figure 7: BERTopic result

As we can see, a dominating trend starting from the late 1980s is the more frequent use of languages used more often by African Americans, which is associated with the rise of hip-hop music. In comparison, the 1970s to 1980s witnessed the upsurge of languages used more often by the white. It may have correlation with Punk. The purple line consists of words urging for peace, showing people's wishes during the Vietnam War. The recent years' trend of Spanish music is also captured by BERTopic. Despite the seeming interpretability, its classification accuracy is not satisfactory. It is easily understood since BERTopic is not a very stable algorithm as well.

Part	Algorithm	Parameter name	Parameter value
Dimensionality reduction	UMAP	n_neighbors	15
Dimensionality reduction	UMAP	n_components	15
Dimensionality reduction	UMAP	metric	cosine
Clustering	HDBSCAN	min_cluster_size	5
Representation	MaximalMarginalRelevance	diversity	0.3
Tokenizer	CountVectorizer	n_gram_range	(1,1)

Table 3: BERTopic hyperparameters

4.1.2 BERT for Classification

We have experimented with BERT to directly classify the lyrics. We adopt the pretrained model BertForSequenceClassification developed by HuggingFace. In order to reduce the overfitting effect, data is first divided into training_and_validation and test sets. At the beginning of each epoch, we randomly split the training_and_validation set into training and validation sets. We then used the training set to optimize the model parameters. During each epoch, we evaluate the model on both validation dataset and test dataset to better understand the overfitting effect. The training process is shown in 8.

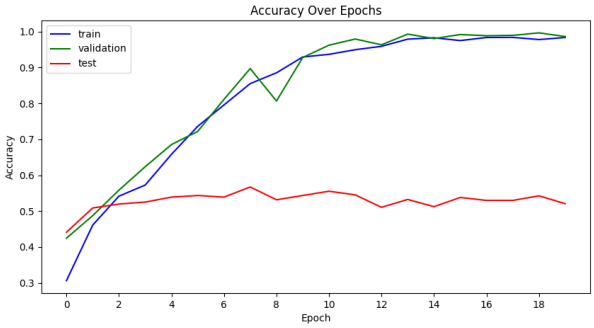


Figure 8: The BERT training process

Based on the above results, we had the following 2 observations:

1. Although the train and validation accuracies both increase to nearly 1, the test accuracy has no significant improvement with the number of epochs. This indicates that the model overfits the training data, yet it does not learn the desired patterns of the lyrics.

2. Due to its millions of parameters, BERT actually memorizes the lyrics rather than learning the patterns of the lyrics. Therefore, we have decided not to use large language models like BERT in this task.

4.1.3 Neural Network Classifier

Recall that for lyrics, word count is more competent than word embedding. Therefore, word count is utilized as the input of the neural network. Specifically, 2 MLP layers are adopted to perform classification. The training process is as follows,

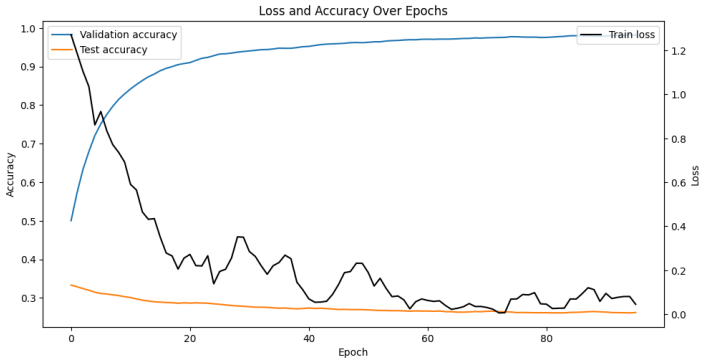


Figure 9: Our primitive neural network classifier

From 9, it is observed that the network also suffers from overfitting in that it reaches an accuracy of 98% in the training process, yet its accuracy on the test set barely changes. It is because of the high dimension (51693) of word count as input. Despite only 2 fully connected layers, parameters of the model are more than 10 times the number of lyrics, which makes it possible for the model to memorize all the training data instead of generalize to unseen ones.

The following 3 approaches have been adopted to tackle overfitting,

1. **Dropout**: adding a dropout layer between the 2 fully connected layers
2. **L1 penalty**: adding a L1 penalty in the loss function to prevent extreme weights
3. **L2 penalty**: adding weight decay to the optimizer

We evaluate the results by training the model for **100** epochs and then compare the accuracy of the training set and the validation set. It is difficult to find good hyperparameters for L_1 and L_2 penalty. We have made our best effort to find them. The following table shows the comparison results.

	Final validation Acc	Final Test Acc
No Dropout	0.9819	0.2559
Dropout	0.8156	0.2941
L1 penalty	0.9231	0.2853
L2 penalty	0.9780	0.2694

Table 4: Classification results comparison

From 4, we can see that all three techniques slightly improved the performance of the neural network on testing data. However, the test accuracy has no significant increases during the training process as is the same in 9, which indicates that none of the above techniques can significantly reduce the overfitting effect and the model lacks the ability to generalize. Therefore, we believe that the neural network is not a good choice for the classification of lyrics.

4.2 Decade Division

4.2.1 Unsupervised Clustering Methods

As demonstrated in the previous section, no compelling evidence suggests the existence of an inherent time division criterion. In other words, there are no abrupt shifts in lyrical styles between consecutive years within specific periods. Consequently, we have chosen an arbitrary threshold to divide the 65-year duration into six decades, as mentioned in EDA Part. These decades are as follows: 1960s (1959-1970), 1970s (1971-1980), 1980s (1981-1990), 1990s (1991-2000), 2000s (2001-2010), and 2010s (2011-2022).

Based on the pre-processed data, we construct a corpus, denoted as matrix A, with dimensions of 5085×3560. This matrix records the word counts for each song. The columns of matrix A represent the different words used in the lyrics, while the rows represent individual songs. Each element A_{ij} in matrix A corresponds to the number of times word j appears in song i. Notably, matrix A is sparse.

To begin our analysis, we standardize matrix A based on each song. Specifically, we utilize the $L1$ norm to standardize each row.

$$\tilde{A}_{ij} = \frac{A_{ij}}{\sum_j A_{ij}}$$

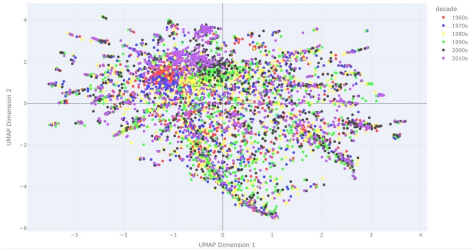
Then we perform clustering methods including Principal Component Analysis (PCA), Locally Linear Embedding (LLE), t-SNE and UMAP on matrix $\tilde{A}$. Even after numerous parameter tuning efforts, we still fail to identify the clustering of any decade. This is not surprising, since standardization is based on lyric length. However, for those timeless words, their entries are quite large, and are still dominant in each row (songs), covering the effect of these less frequent but more featured words.

So we try instead to use the same tactic as in EDA Part — TF-IDF, this time, however, based on each song. We convert the original matrix A into matrix A'.

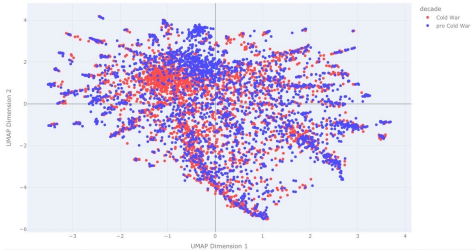
$$A'_{ij} = A_{ij} * \log \frac{5085}{\sum_i I(A_{ij} \neq 0)}$$

In this analysis, four clustering algorithms are applied to the matrix A'. PCA, being a basic linear algorithm, does not yield satisfactory results, which is within our expectation, since is not designed to capture complex non-linear relationships. On the other hand, t-SNE, LLE, and UMAP are able to produce better results after parameter tuning. LLE, being a manifold learning algorithm, focuses on preserving the local structure of the data. However, it tends to struggle when dealing with large datasets, as in this case. t-SNE assumes that the data follows a t-distribution, which is not the case for the sparse matrix A'. As a result, t-SNE also does not perform optimally in this scenario. UMAP, on the other hand, is based on topological structures and constructs a low-dimensional representation that preserves both local and global structure. This makes UMAP more robust to a wider range of parameters. Interestingly, a wide range of parameters are found to perform well with UMAP in this analysis.

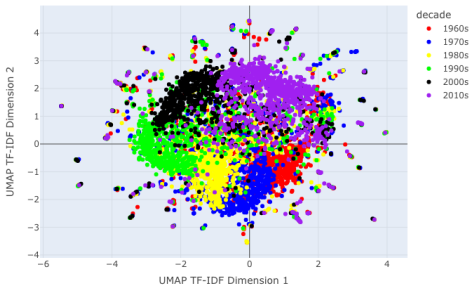
Overall, UMAP outperforms other algorithms due to its ability to capture both local and global structures of the data. Therefore, we present the optimal UMAP result on the TF-IDF matrix denoted as A', using the following parameters for visualization: components=2, number of neighbors=15, minimum distance=0.1, and random state=15. Additionally, we provide the UMAP result on the standardized matrix in Figure 10. The plot demonstrates the classification of songs from six decades, with the clustering becoming more apparent when we set the year 1990 (the end of the Cold War) as a threshold to divide the songs into two groups.



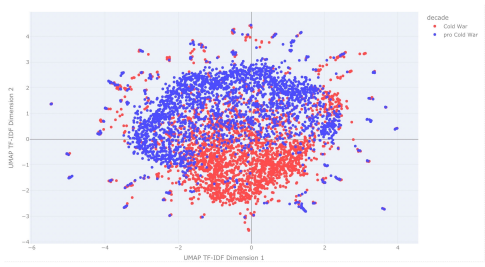
(a) UMAP on standardized matrix (six decades)



(b) UMAP on standardized matrix (pre and pro Cold War)



(c) UMAP on TF-IDF matrix (six decades)



(d) UMAP on TF-IDF matrix (pre and pro Cold War)

Figure 10: Four features from observation

4.2.2 Heat Map

Although UMAP clustering offers some sort of insight into the relationship between songs from different decades, we aim to obtain a more specific and quantified way. To achieve this, we conduct one-to-one logistic regression for each pair of decades and record their error rate. Figure 11 depicts a heatmap based on the error rate, where darker colors indicate lower error rates and higher correlation. The heatmap reveals significant differences between the three decades preceding the Cold War and the three decades following it. Within each time period, the three decades before the Cold War exhibit a strong correlation, while the three decades after the Cold War display a lower correlation.

This concise result gives instructions for the division of years in classification. In the classification step, we perform two classes classification (pre and pro Cold War); four classes classification (pre Cold War, 1990s, 2000s and 2010s); as well as six classes classification (the six decades).

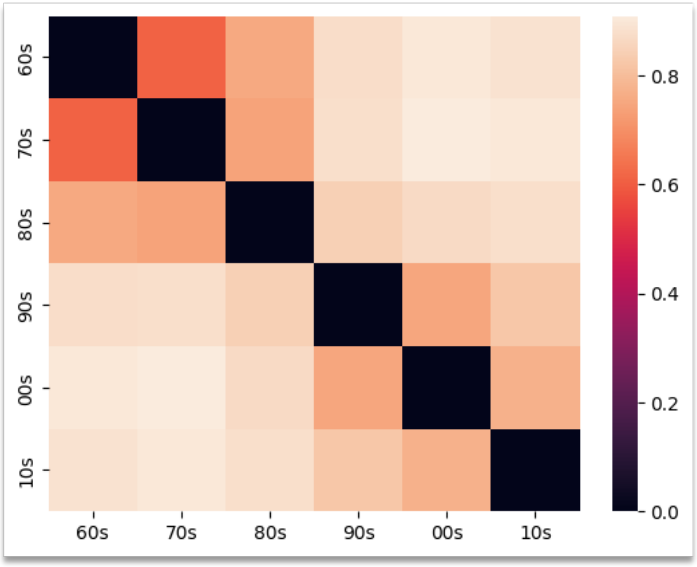


Figure 11: the accuracy of logistic regression between each decade

5 Interpretable classification

In this section, logistic regression with penalization is performed to achieve interpretable classification.

5.1 Setting

The coefficient of logistic regression can directly reflect its importance in prediction, as long as features are of the same scale. Therefore, normalization is performed over all the word frequency variable. Note that word frequency matrix holds the property of non-negativity and sparsity. To preserve these properties, scaling is performed by dividing each feature by the largest value of it.

There are two major choices of multi-class classification for logistic regression, respectively voting and one against all. For interpretability, one against all is superior, because popular words of a specific period can be shown directly. Additionally, it also has better prediction performance than voting. Therefore, one against all logistic regression is preferred.

As for the choice of penalty, due to the large amount of data, elastic net fails to converge and suffers

from computation speed in many occasions. Although L_2 penalty sometimes have better prediction accuracy than L_1 penalty, L_1 penalty is chosen, for L_1 penalty is able to select features which meets the need of this research.

The parameter of logistic regression is chosen via 5-fold Cross Validation with error rate as metric. Considering the prevalence of each one-against-all logistic regression, we performed Bootstrap in the Cross Validation pipeline. However, it hugely increases computational complexity with limited performance improvement. Thus, the final Cross Validation does not include the Bootstrap step.

After training the logistic regression, the coefficients are ranked in descending order. The top 100 features are chosen to generate word cloud for direct interpretation.

5.2 Prediction Accuracy

The prediction accuracy of 2 classes, 4 classes and 6 classes are presented. From the table below, logistic regression for 2 classes and 4 classes shows satisfactory prediction accuracy, while accuracy for 6 classes is not ideal. It coincides with our intuition in the preceding section.

	2 classes	4 classes	6 classes
Accuracy	87.34%	75.32%	52.58%

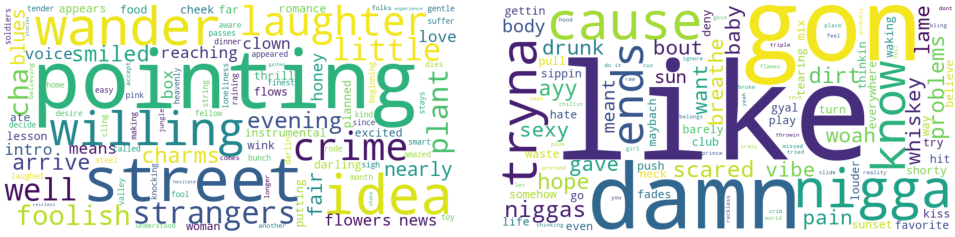
Table 5: Prediction Accuracy of Logistic Regression

5.3 Interpretation

5.3.1 2 classes

Popular words before Cold War are formal and mostly written words, such as willing, crime, wander, instrumental, foolish, laughter. Emotional adjectives are more frequent.

Words after Cold War are more frivolous, where languages appear frequently, like damn, n-word, sexy. Besides, oral abbreviations are frequent, such as cause (because), gon (going to), tryna (try to), bout (about). Emotions are expressed more often in verbs rather than adjectives.



(a) Popular Words before 1990

(b) Popular Words after 1990

Figure 12: Word cloud of pre and post 1990

5.3.2 4 classes

Recall that logistic regression is one against all. Thus, popular words before 1990 will be the same. Popular words in the 1990s reflects the upsurge of hip-hop genre, such as rhythm, flow, Brooklyn. Also, the 1990s witness two major global economic crisis. Depression is shown clearly in the lyrics, like despair, confused, needed.

In the 2000s, with the advancement of the Internet and blooming economy, hedonism spreads over lyrics. Provocative words like club, thighs, crazy, shorts appear often. Additionally, anti-war emotion is prevalent, with scared, ends, escape, stress, breathe etc. This is largely due to the second Iraq war.

The 2010s seem to be marked by decadent culture. Words like whiskey, drunk, swear, shoot have shown lyrics is distinguished for its association with alcohol and not noble behavior.

(a) Popular Words of 1990s

(b) Popular Words of 2000s

Figure 13: Word cloud of 4 classes

Figure 14: Popular Words of 2010s

5.3.3 6 classes

As seen before, 1960s to 1980s enjoy similar themes in lyrics. The generated word clouds are also alike, except for the 1970s. Words like woman, lady indicates feminist acts in the 1970s. There are also anti-war emotions, reflected by pardon, lord, peaceful, shame, sorrow, by the Vietnam War.

Figure 15: Popular Words of the 1970s

6 Conclusion and Discussion

6.1 Conclusion

First, through EDA and unsupervised learning, it is observed that every time period of lyrics has its own set of popular words. This indicates that every period has its distinctive trends and characteristics in terms of popular music.

The EDA, unsupervised learning, heat map and interpretable classification analysis all reveal that the end of Cold War serves as the most significant milestone that leads to a notable change in popular words used in lyrics. After Cold War, lyrics become more oral and frivolous.

For each specific decade, it is found that major milestones and events that took place during that period play a crucial role in shaping the popular words in lyrics. These events include significant wars such as the Vietnam War or the Iraq War, as well as movements like feminism. In addition, the upsurge of a certain genre also contributes to shaping the popular words used in lyrics, such as the rise of Hip-hop in 1990s.

One technical insight gained from this project is the association between prediction accuracy and interpretability. A period division that results in a high prediction accuracy leads to more interpretable results. Therefore, even in the pursuit of interpretability, prediction accuracy is an important indicator of the plausibility and reliability of the interpretation.

6.2 Discussion

Although we have reached very convincing and interpretable result, there still exist several unresolved issues.

6.2.1 Neural network & clustering

We have experimented with the Variational Auto-encoder (VAE) to reduce the dimensions of input data. To be more specific, we want to train a VAE and extract the outputs of the bottleneck layer

as low-dimension data. Based on our earlier work, we believe that for lyrics, word count has more competence than word embedding in lyrics data. Therefore, we integrated both TF-IDF vectorization and word embedding into the inputs.

Combining the above two types of data, we created 100,000-dimension input vectors and fed them into our neural network, which consists of an encoder and a decoder each with 2 fully connected layers. The loss function is the sum of the reconstruction loss and the KL divergence loss, which is typical in VAE models

[Doe21]. The reconstruction loss is the mean squared error between the input and the output of the decoder. The KL divergence loss is the KL divergence between the output of the encoder and the standard normal distribution. However, in our experiment, the loss of our VAE model hardly decreases. The reason might be that VAE is based on the assumption that there is a hidden Gaussian Mixture Distribution behind the data. In our case, however, as the topics change over time (if there are indeed), the center of the Gaussian Distribution could shift, which could make it extremely hard for the model to discover. There are more advanced VAEs like [LJY20], but it is beyond our capability due to our limited time and experience.

6.2.2 Other Concerns

First, no viable and effective dimension reduction method is appealing enough. Although the number of samples is larger than the number of features, dimension reduction may still enhance prediction accuracy and interpretability. Feature selection by its correlation with the label is impractical, since all variables have a weak association with the label, as shown in the box plot below.

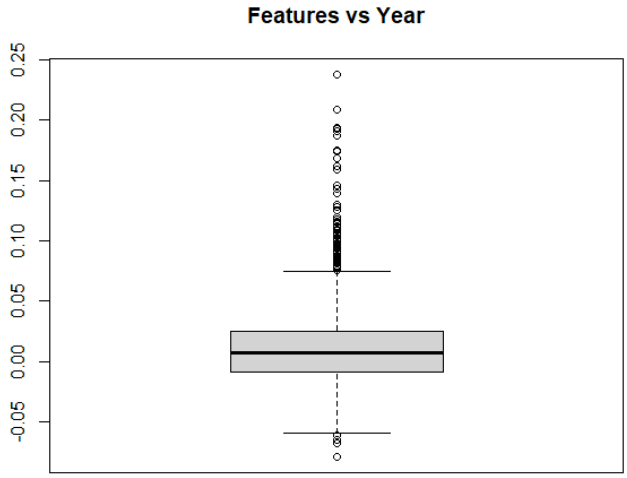


Figure 16: Correlation of variables w.r.t. label

Further, the label is an ordinal variable, while treated as an unordered variable. To make use of the order information, which is infeasible for logistic regression with penalty, an alternative view might be beneficial. The classification problem can be treated as multi-label regression with inter-label correlation. [BXT12] presents a way of dealing with correlations between labels in logistic regression.

It modifies the assumption on the conditional distribution to,

$$p(y|x) = p_{lr}(y|x)q(y) \tag{1}$$

where

$$p_{lr}(y|x) = \frac{\exp(x^T\beta)}{1+\exp(x^T\beta)}$$
$$q(y) = \exp(\sum_{i<j} \alpha_{ij} y_i y_j)$$

Note that the "penalization" term $q(y)$ is similar to Gaussian kernel, which measures similarity. In this project, the kernel should be revised, since the inner product of labels is 0. Because there is no existent package of CorrLog and more time is needed for the choice of similarity measure $q(y)$, it has not yet been adopted by this project.

Though BERTopic shows some interpretability, a simpler model may be preferred, considering computation complexity. Therefore, ETM combined with TF-IDF may yield similar or even more stable results than BERTopic, which will be explored in the future.

7 Contribution

Zhiyu Xu: Responsible for data cleaning, logistic regression and interpretation. Writes the corresponding parts, including those in Introduction and Discussion.

Zhixiao Xiong: Responsible for collecting data (scraping data from online websites and first-round cleaning), word embedding models and deep learning models (BERTopic, Bert for classification, neural classifier and VAE, though none of them being successful). Writes the corresponding part in Time Period Division and Discussion.

Cenhao Zhu: Responsible for data cleaning, EDA, feature engineering, decade division and interpretation. Write the corresponding parts, including those in Introduction and Discussion.

The three authors contributed equally to the paper, the author names are listed in alphabetical order.

References

[Bah] Hareesh Bahuleyan. Music Genre Classification using Machine Learning Techniques.

[BL] David M. Blei and John D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 113–120. Association for Computing Machinery.

[BNJ] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. 3:993–1022.

[BXT12] Wei Bian, Bo Xie, and Dacheng Tao. Corrlog: Correlated logistic models for joint prediction of multiple labels. In Neil D. Lawrence and Mark Girolami, editors, *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, volume 22 of *Proceedings of Machine Learning Research*, pages 109–117, La Palma, Canary Islands, 21–23 Apr 2012. PMLR.

[Doe21] Carl Doersch. Tutorial on Variational Autoencoders, January 2021.

[DRBa] Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. The dynamic embedded topic model.

[DRBb] Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. Topic Modeling in Embedding Spaces. 8:439–453.

[DS] Trung-Thanh Dang and Kiyoaki Shirai. Machine Learning Approaches for Mood Classification of Songs toward Music Search Engine. In *2009 International Conference on Knowledge and Systems Engineering*, pages 144–149.

[F.R01] Karl Pearson F.R.S. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[GGKC20] Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. Locally linear embedding and its variants: Tutorial and survey, 2020.

[Gro22] Maarten Grootendorst. BERTopic: Neural topic modeling with a class-based TF-IDF procedure, March 2022.

[Hof] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 50–57. Association for Computing Machinery.

[JMN12] Noraini Jamal, Masnizah Mohd, and Shahrul Azman Noah. Poetry classification using support vector machines. *Journal of Computer Science*, 8(9):1441, 2012.

[Joa] Thorsten Joachims. *Learning to Classify Text Using Support Vector Machines*. Springer US.

[KM] Vipin Kumar and Sonajharia Minz. Poem Classification Using Machine Learning Approach. In B. V. Babu, Atulya Nagar, Kusum Deep, Millie Pant, Jagdish Chand Bansal, Kanad Ray, and Umesh Gupta, editors, *Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012), December 28-30, 2012*, Advances in Intelligent Systems and Computing, pages 675–682. Springer India.

[KW22] Zheng Tracy Ke and Minzhe Wang. Using svd for topic modeling, 2022.

[LJY20] Kart-Leong Lim, Xudong Jiang, and Chenyu Yi. Deep Clustering With Variational Autoencoder. *IEEE Signal Processing Letters*, 27:231–235, 2020.

[MCCD] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space.

[MHM20] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.

[TSD08] Hamid R Tizhoosh, Farhang Sahba, and Rozita Dara. Poetic features for poem recognition: A comparative study. *Journal of Pattern Recognition Research*, 3(1):24–39, 2008.

[vdMH08] Laurens van der Maaten and Geoffrey Hinton. Viualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 11 2008.

[XWW08] Yunqing Xia, Linlin Wang, and Kam-Fai Wong. Sentiment vector space model for lyric-based song sentiment classification. *International Journal of Computer Processing Of Languages*, 21(04):309–330, 2008.