Beyond Lyrics: Exploring Song Themes with Text Mining

Text Mining Project

Marta Brasola 905305

Andrea Malinverno 847340

Mattia Proserpio 846858

Genius

Genius is an America digital media comany

is an online music encyclopedia

provides annotations and interpretation

Project Objectives

Topic Modeling

Analysis of songs lyrics through **decades** and **genres**

Text Classification

Song lyrics Classification

Songs Metadata

- Artist: Artist Name
- Lyrics: Lyrics of the song
- Tag:
  - rap,
  - pop,
  - rb,
  - rock,
  - country
- Title: Title of the song
- Views: clicks on the song

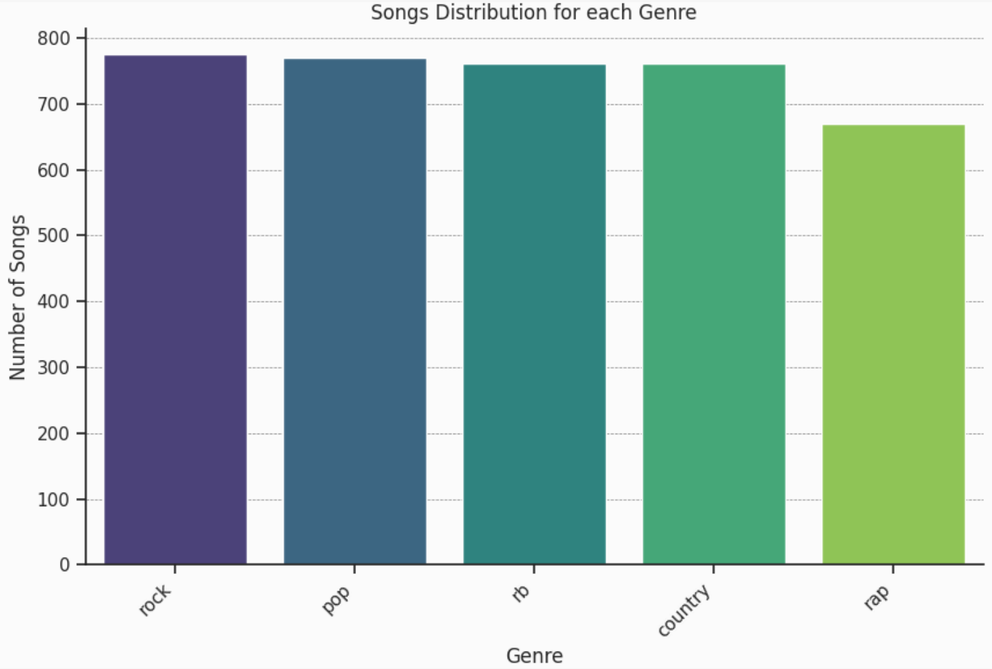
Dataset

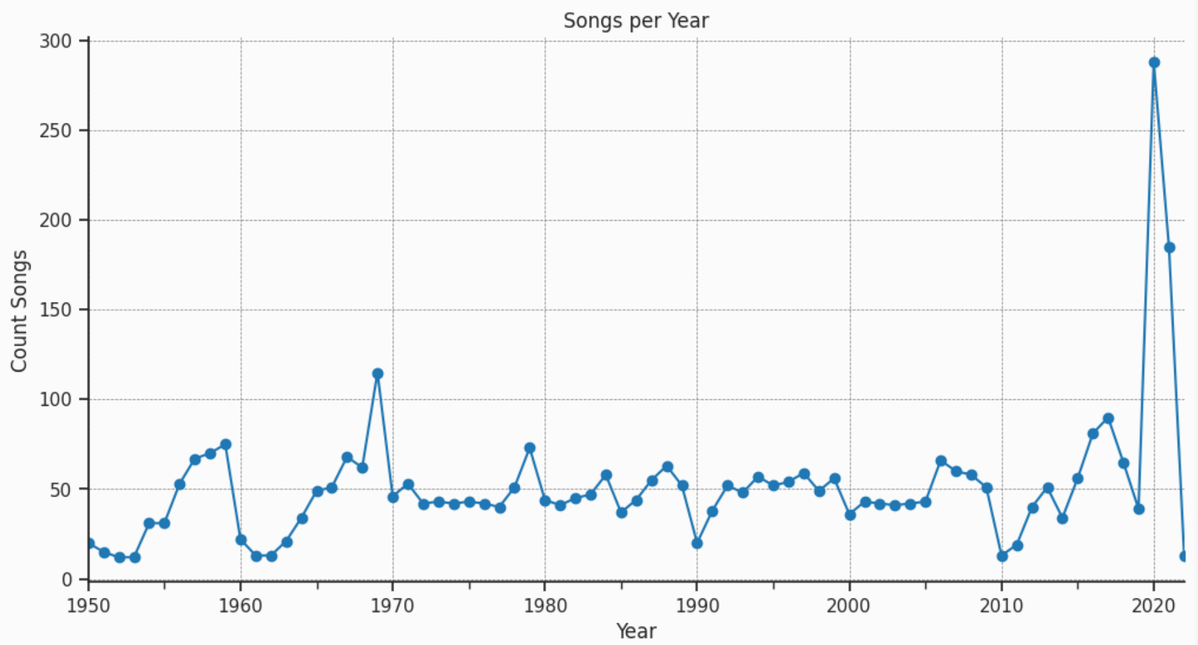
Original Dataset

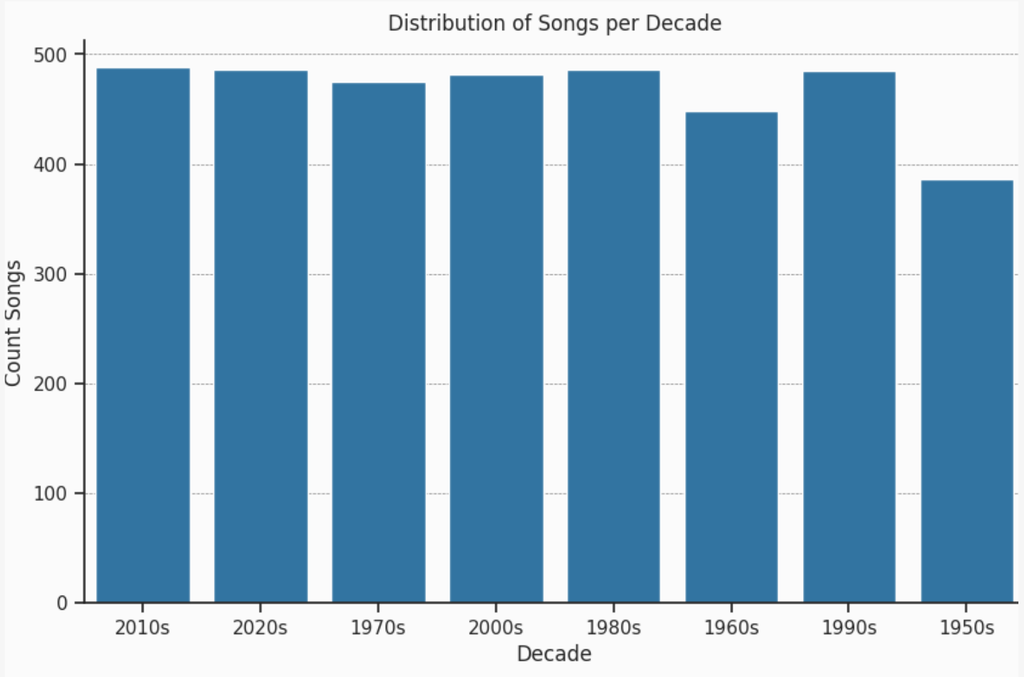
Dataset from Kaggle with more than 5 millions of songs scraped from Genius

Subset

For our analysis we used a subset of 3736 songs







EDA

1.Songs distribution for each genre
2. Songs distribution per decade
3.Songs per year

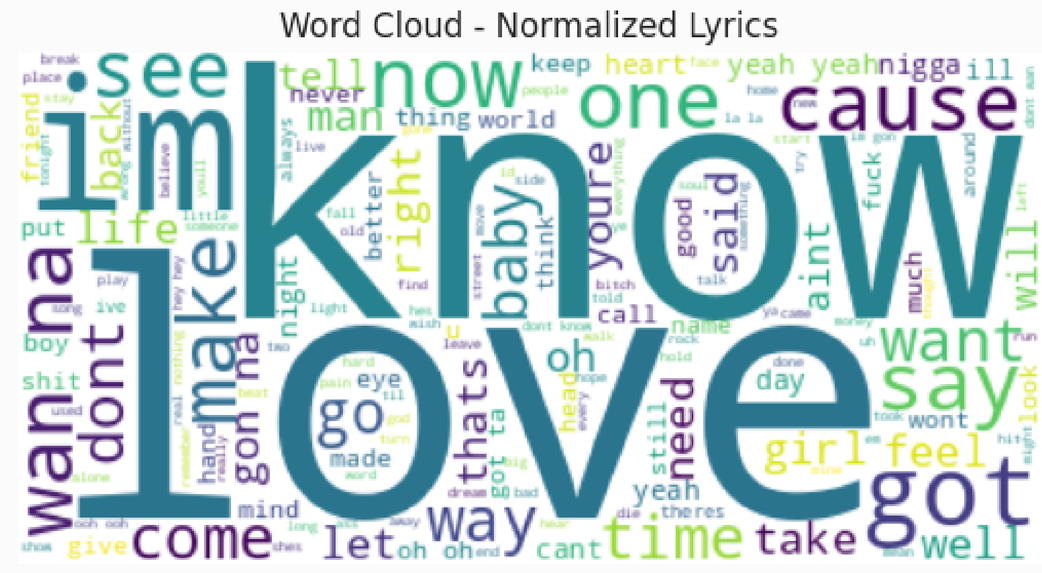
Text processing

Text Preprocessing steps:

1. Normalization
2. Tokenization
3. Stop words removal
4. Lemmatization

	Total Words Count	Avg Words per Song	Unique Words Total	Avg Unique Words per Song
norm_lyrics	1387318	371.338	31348	8.39079
no_stop	723647	193.696	31201	8.35145
lemm_no_stop	720600	192.88	26233	7.02168

World Cloud



To the left, a word cloud with the inclusion of stop words; to the right, one where they are excluded and the words are lemmatized.

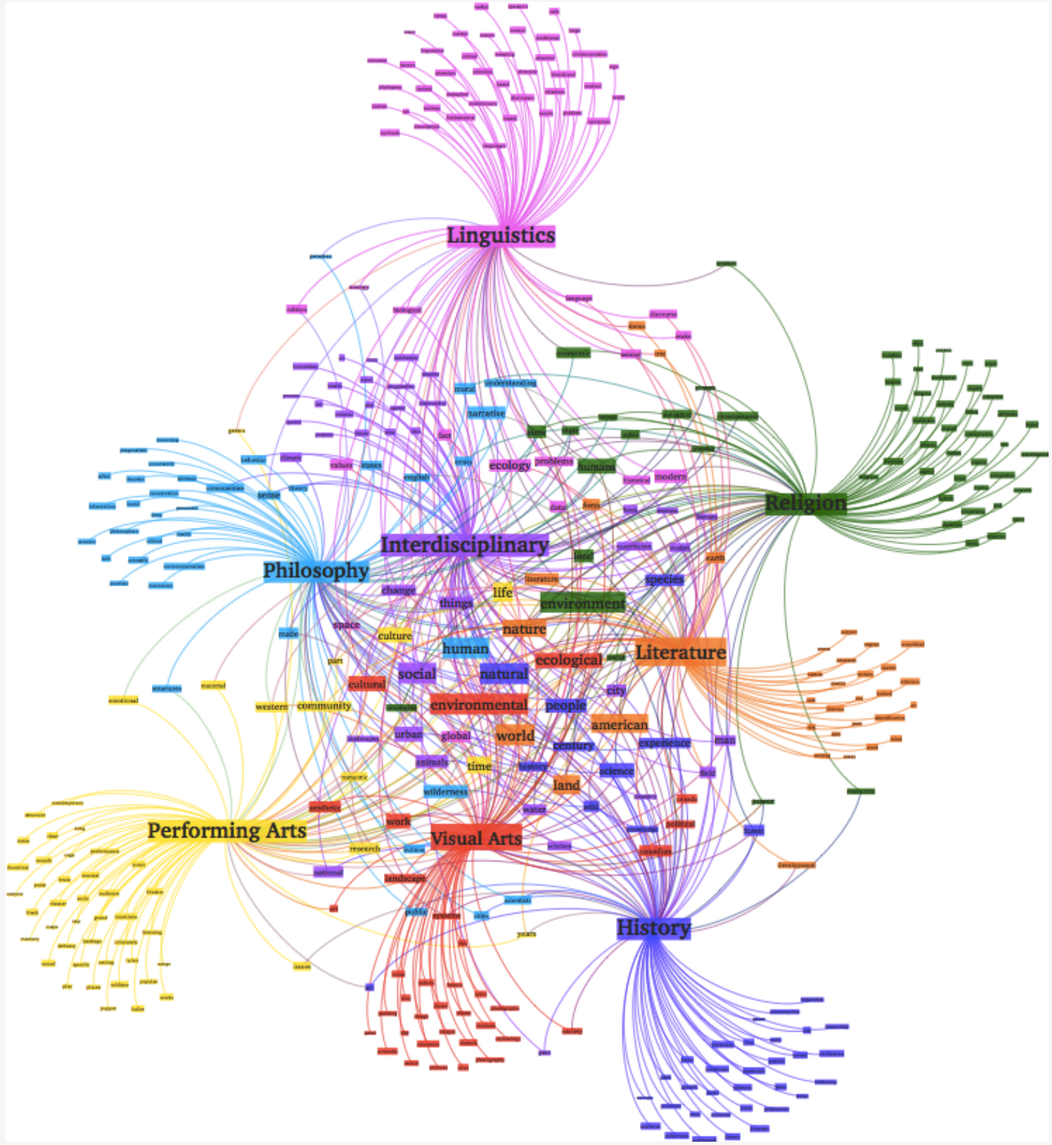
Topic Modeling

01

Latent Semantic Analysis - LSA

02

Latent Dirichlet Allocation - LDA



Latent Semantic Analysis - LSA

Representation → **SVD**

Representation through TF-IDF

On the TF-IDF matrix

The LSA model fails to make Topics with distinguishable words

Topic	0	1	2	3
0	love	take	man	baby
1	youre	come	cause	let
2	baby	well	fuck	girl
3	time	away	shit	wan
4	make	day	aint	la
5	never	make	nigga	aint
6	cant	time	bitch	come
7	one	one	rock	see
8	want	see	make	want
9	feel	run	back	make

Latent Dirichlet Allocation - LDA

Preprocessing → **Representation** → **Hyperparamter Tuning** → **Evaluation**

Filtering with cuf-offs levels, from *26233* to *5882* words

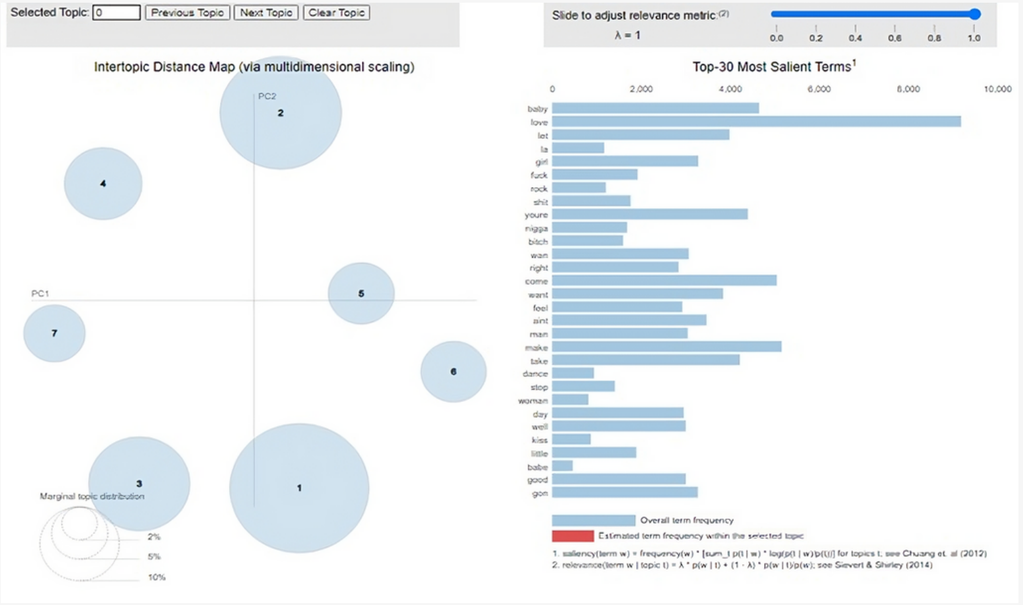
Representation through Bag-of-Words

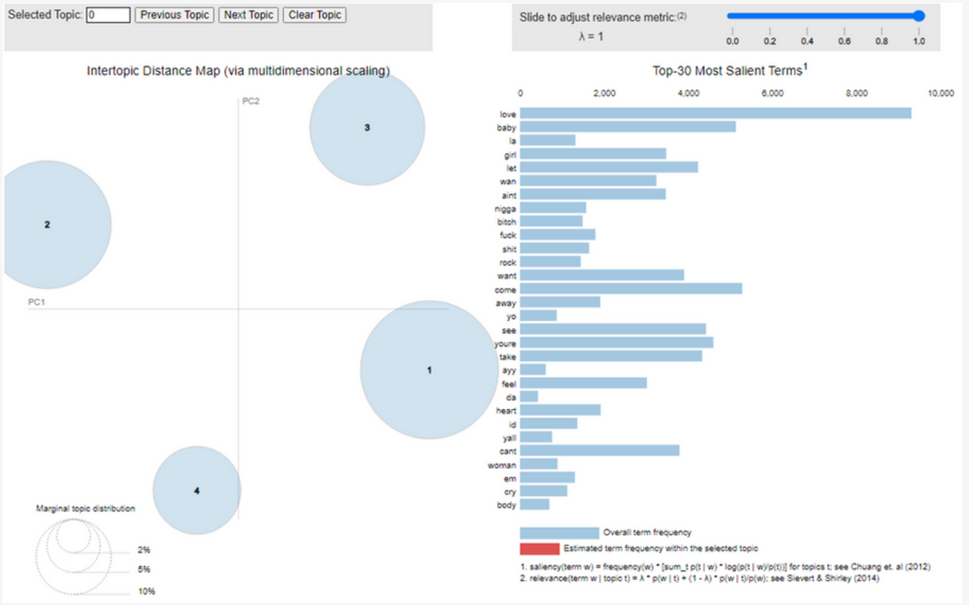
- alpha
- beta
- number of topics

- u_mass
- c_v

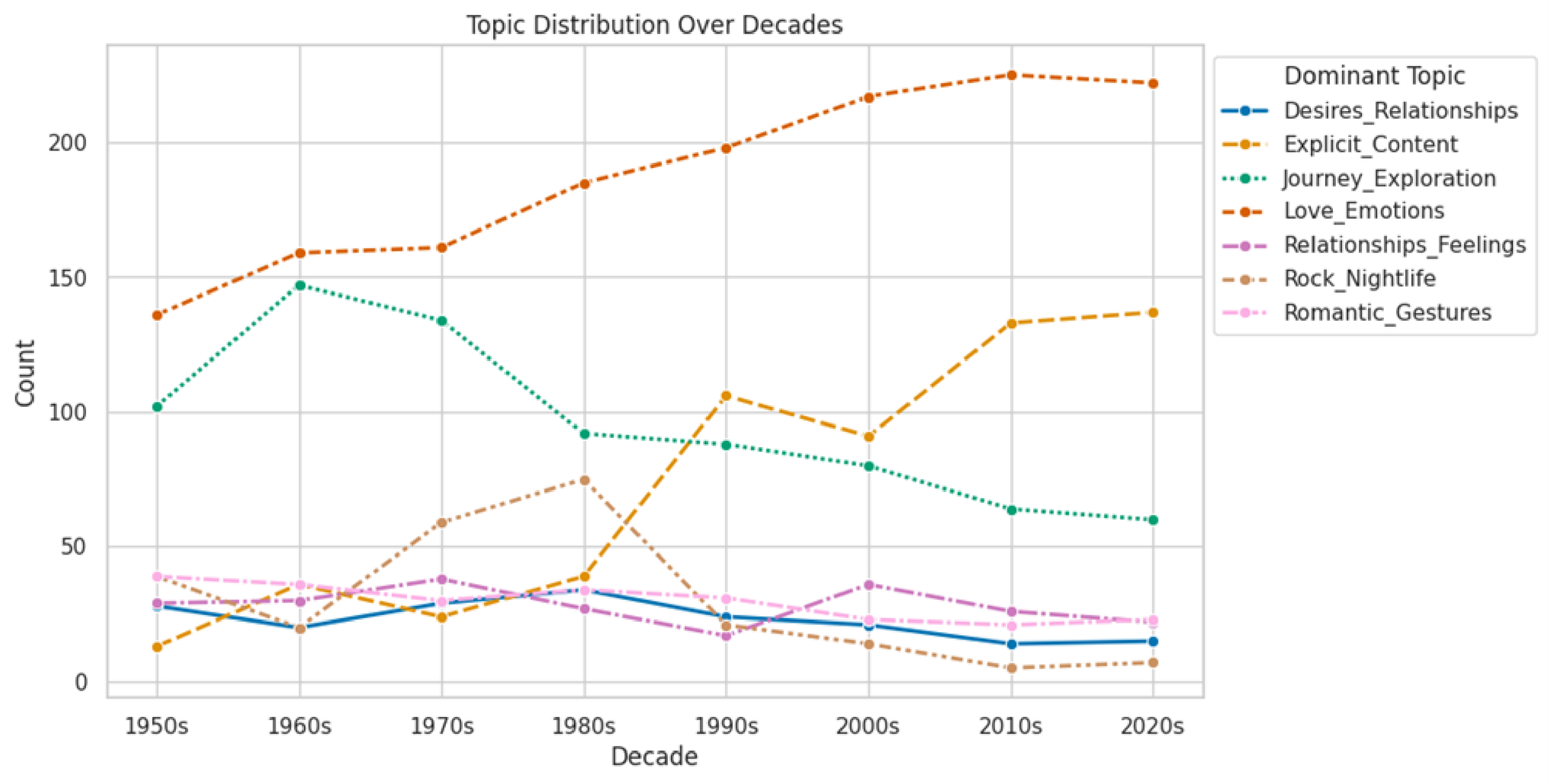
Topic Modeling - LDA

Model	n_topics	alpha	beta	u_mass coherence	c_v coherence
Model 1	7	0.1	0.5	-1.285	0.336
Model 2	4	0.1	0.1	-0.995	0.362

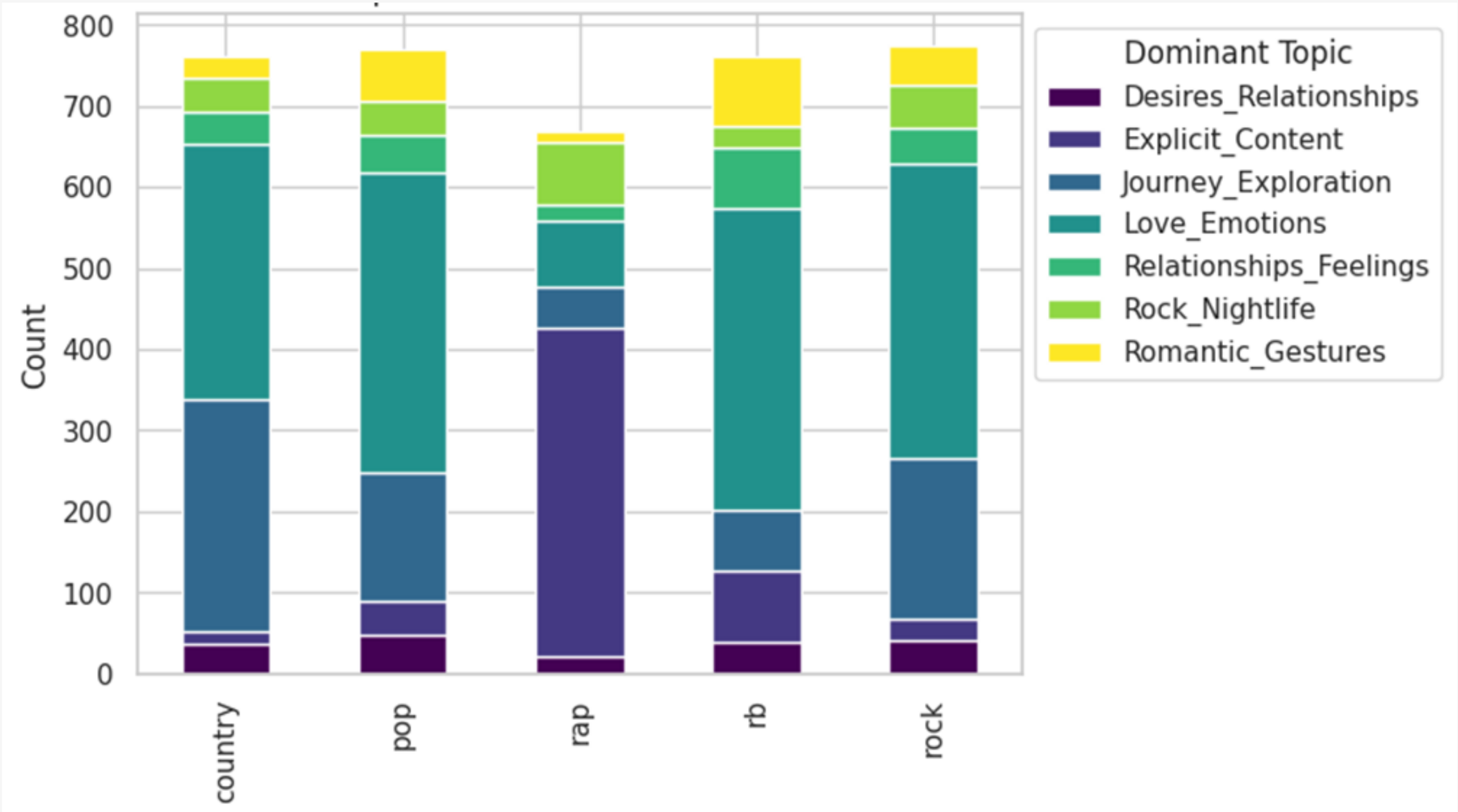




Topic for Decade



Topic for Genre



Text Classification

BINARY CLASSIFICATION

- **1 = Rap**
- **0 = other genre**

TF-IDF

Models :

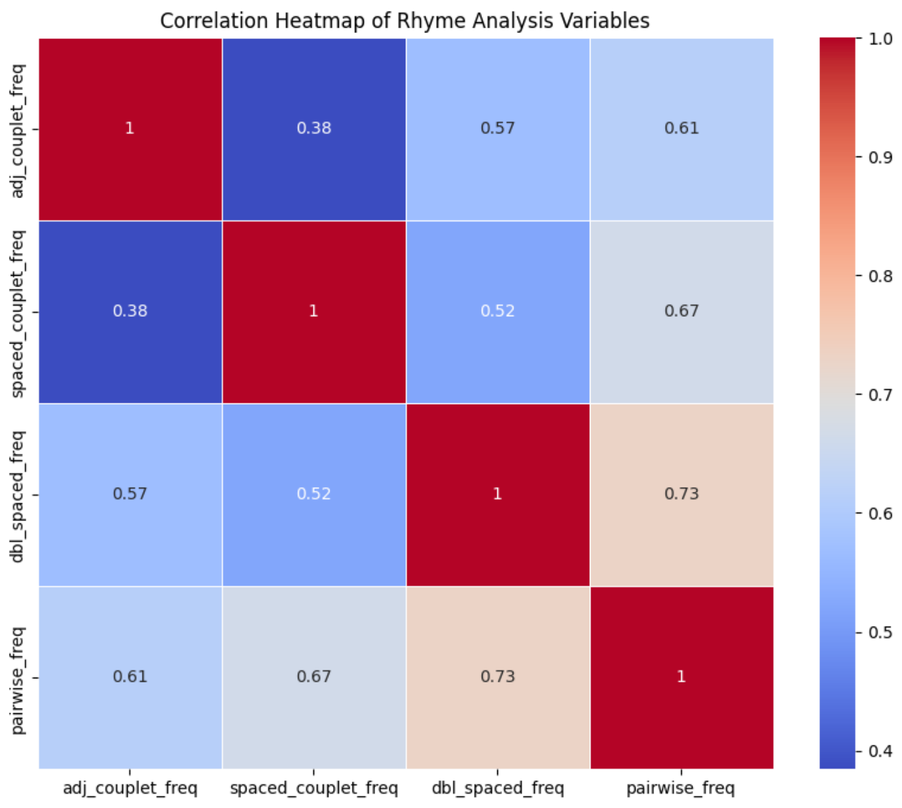
- **NaiveBayes**
- **Logistic Regression**
- **Random Forest**
- **Gradient Boosting**

RESULTS

Model	max_features	min_df	max_df	Accuracy	ROC
Naive Bayes	1000	0.01	0.8	0.906	0.882
Logistic Regression	1000	0.01	0.8	0.909	0.889
Random Forest	1000	0.01	0.8	0.916	0.887
Gradient Boosting	500	0.01	0.9	0.913	0.881

Table 4.1: Classification Results

Rhyme Features



1. **Splitting lines**
    - Splitting lines at “\n” character
2. **Pronunciation**
    - get the phonetic pronounciation for every word
3. **Check Rhyme Patterns**
    - ratio of AA
    - ratio of ABA
    - ratio of any double pattern
    - ratio of any pattern

Rhyme Features

tag	adj_couplet_freq	spaced_couplet_freq	dbl_spaced_freq
country	0.132	0.105	0.040
pop	0.148	0.135	0.073
rap	0.162	0.094	0.052
rb	0.172	0.153	0.091
rock	0.175	0.156	0.068

Models :

- **NaiveBayes**
- **Logistic Regression**
- **Random Forest**
- **Gradient Boosting**

Model	Accuracy	ROC
Naive Bayes	0.902	0.863
Logistic Regression	0.909	0.895
Random Forest	0.907	0.872
Gradient Boosting	0.902	0.883

Text Classification

Pre-Trained Model

BERT

BIDIRECTIONAL ENCODER
REPRESENTATIONS FROM TRANSFORMERS

From pre-trained 'bert- base-uncased'

- **Optimizer : Adam**
- **Learning Rate = 0.0001**
- **Epochs = 10**

RESULTS:

Accuracy → **0.91**

ROC-AUC → **0.8**

Future Improvements

- larger dataset
- creating a better algorithm to check rhymes
- testing more text representation techniques