

# **Lyrics and Popularity**

**Exploring the Value of Lyrics**

Justin Ng

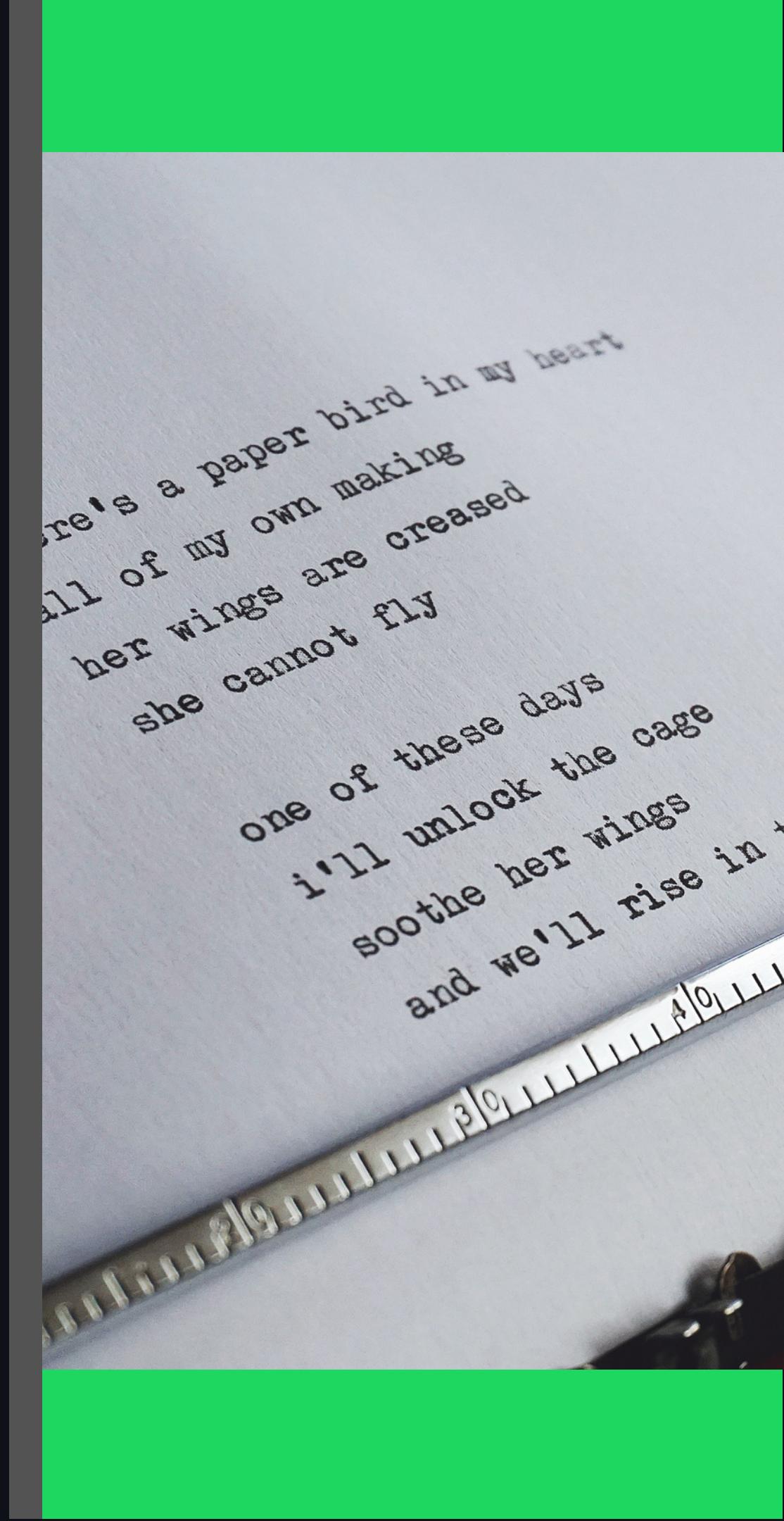
April 12, 2023

Tahoma

# The Problem

Lyrics have been a **neglected component of songs when attempting to predict** song popularity.

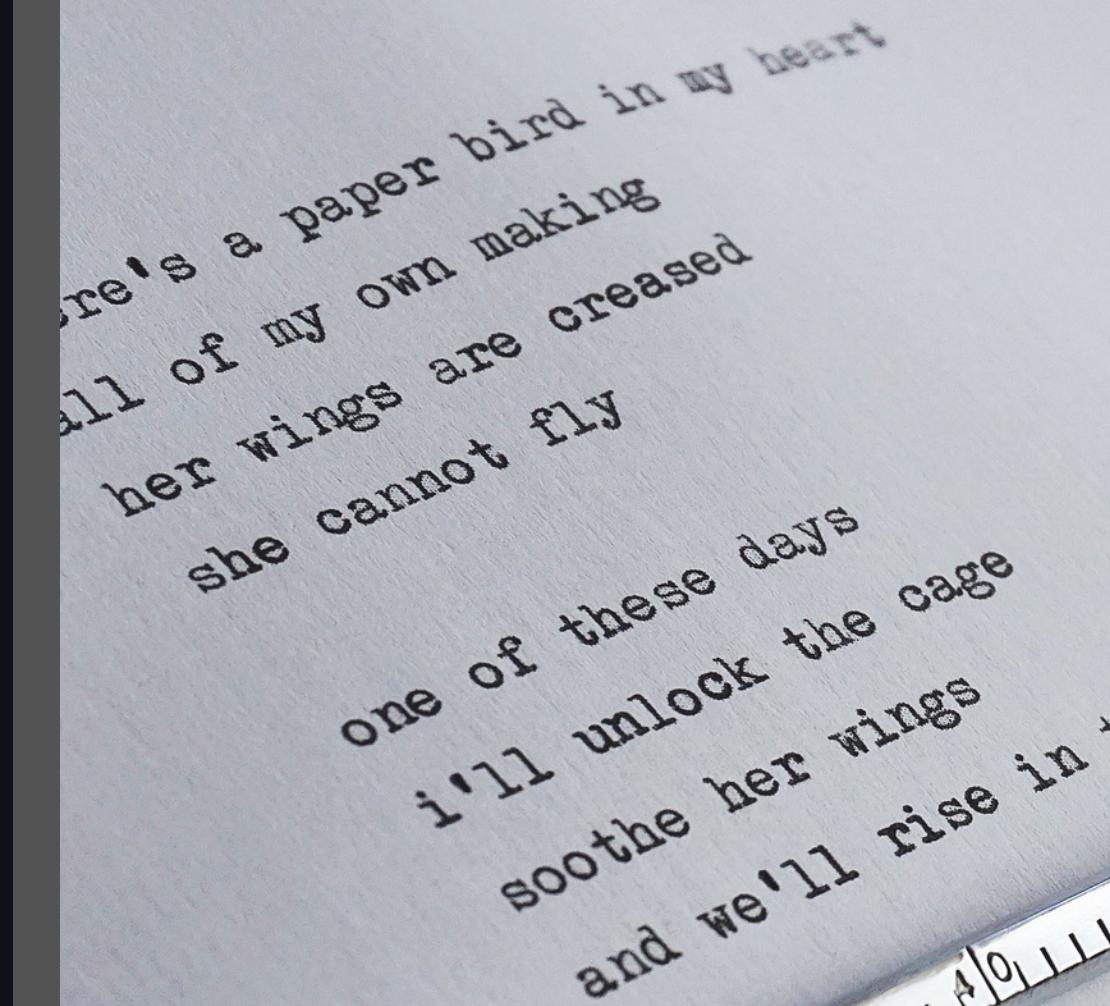
Difficulties occur from working with text data and **differing vocabularies between genres**.



# The value

A deeper understanding of lyrics could help songwriters **create popular songs more efficiently.**

Understanding popular lyrics could also provide a **snapshot into the cultural themes** of a given time.



there's a paper bird in my heart  
all of my own making  
her wings are creased  
she cannot fly  
one of these days  
i'll unlock the cage  
soothe her wings  
and we'll rise in \*

# The Objective

---

Predict **song popularity** using  
only **song lyrics**.

---

Tahoma



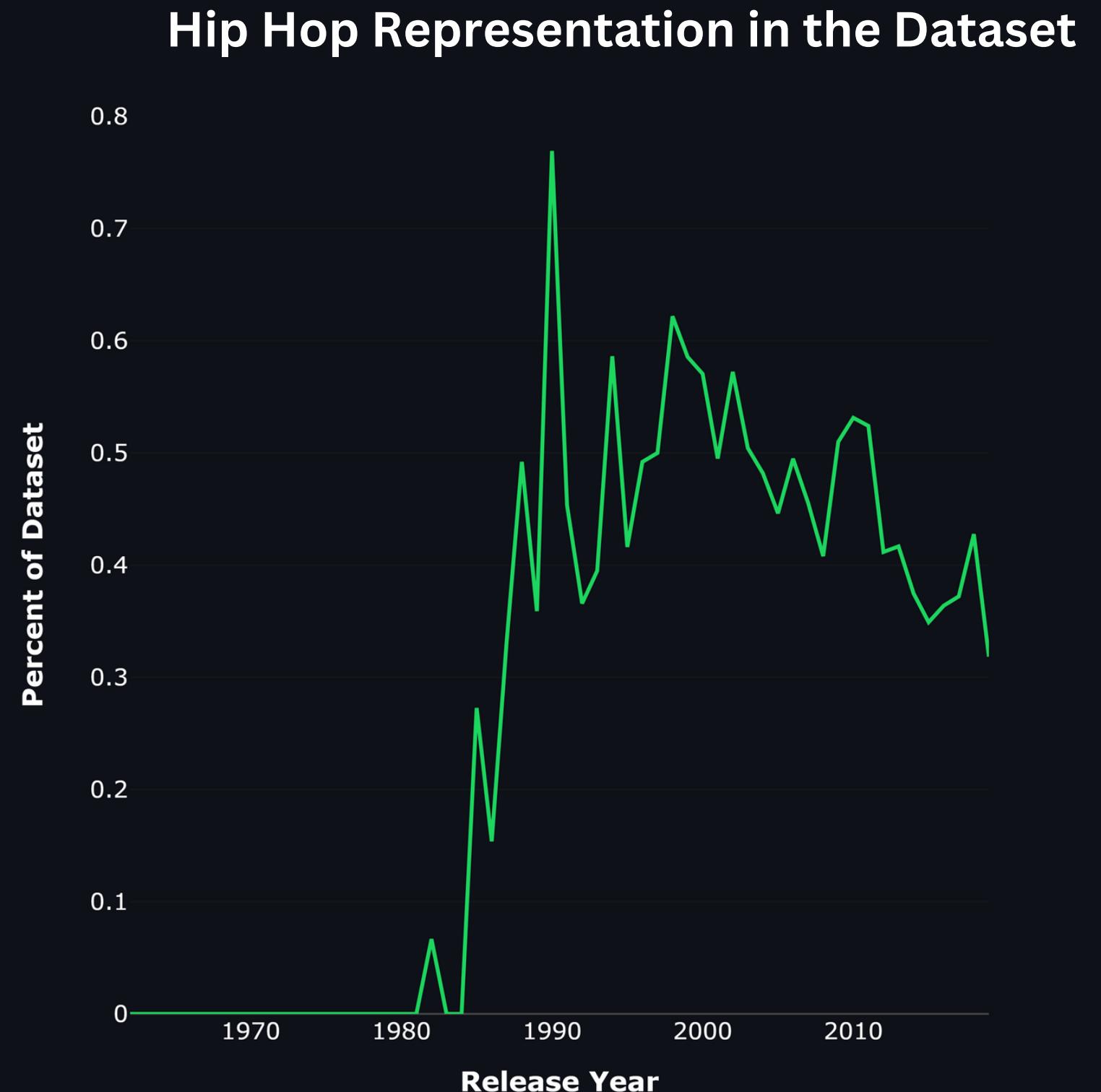
# The Data

**28 560 Lyrics**

scraped from Genius.com.

**41.2%**

of the dataset was **Hip Hop**



# The Target

Translation: What we are trying to predict.

Scraped **Spotify popularity rating**, using the Spotify API. This rating is defined mainly by Spotify **plays and the recency of those plays**.

Divided into three classes, **low, medium and high popularity**, based on the popularity rating.



# The Models

Table 1. Modeling Results After Tuning

Model	Text Transformation	Test Accuracy	AUC of Micro-Average ROC Curve
Logistic Regression	TF-IDF	0.42	0.60
	TF-IDF + NMF	0.40	0.58
	Ada Embeddings	0.45	0.63
	Ada Embeddings + PCA	0.42	0.60
Multinomial Naive Bayes	TF-IDF + Hip Hop Only	0.43	0.62
	TF-IDF	0.42	0.60
Random Forest	TF-IDF + Hip Hop Only	0.45	0.63
	TF-IDF	0.42	0.60

Using Only Lyrics is Difficult.

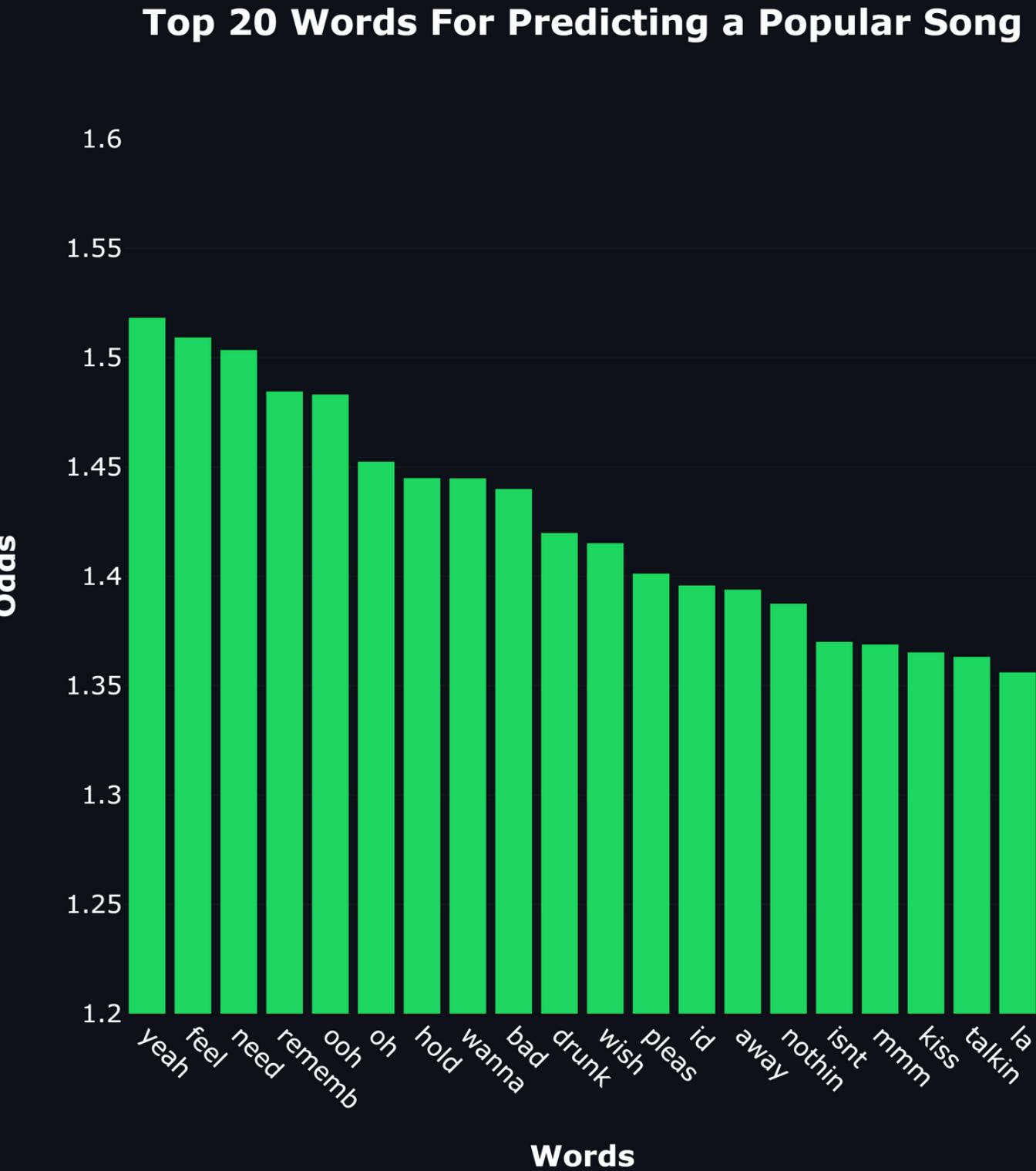


# The Findings

Popular Songs

Words that dealt with **positive feelings and physical connection** demonstrated increased odds of being predicted as having **High popularity**.

Yeah  
Feel  
Need  
Kiss  
Wish  
Hold



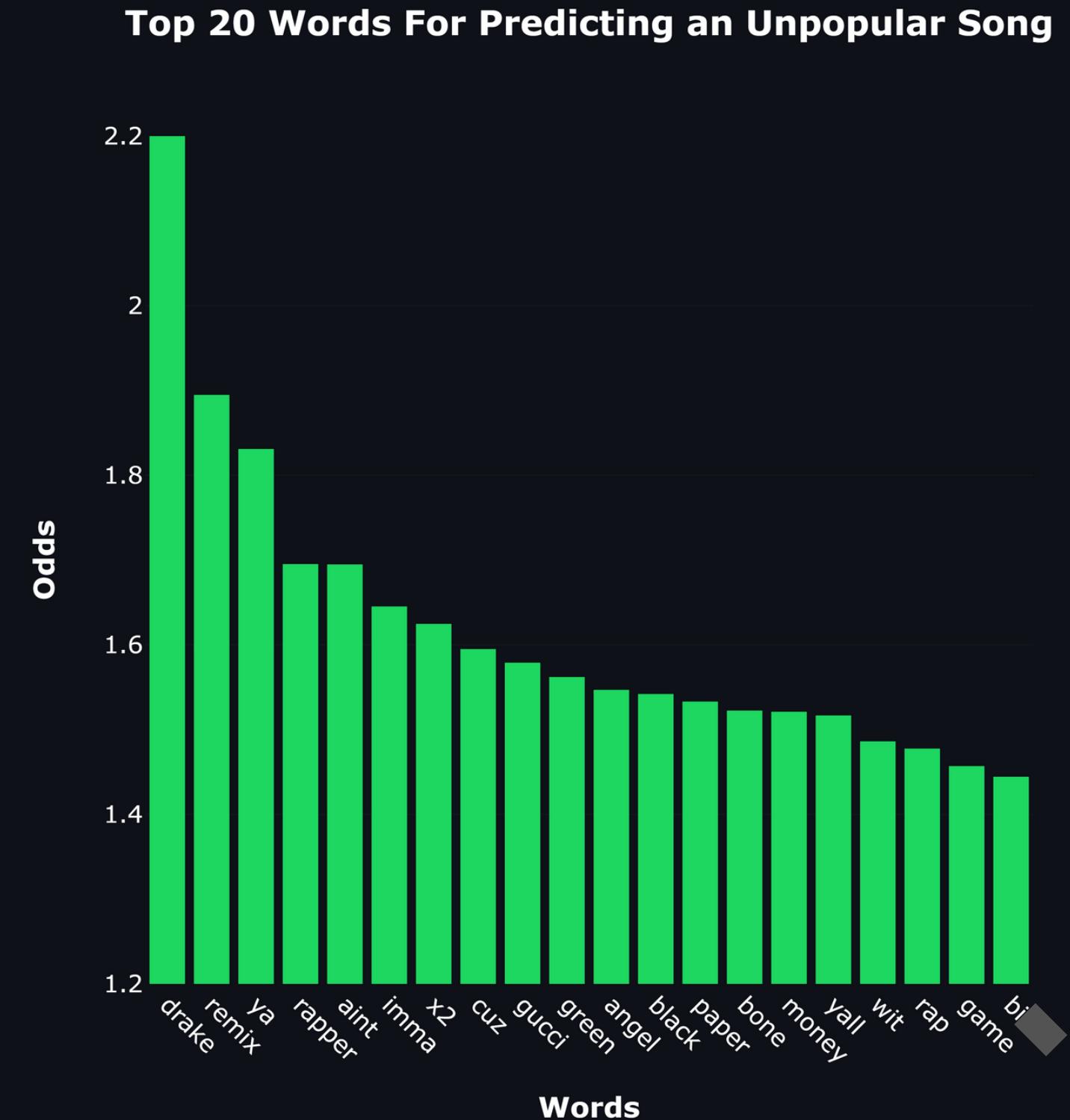
# The Findings

Unpopular Songs

Words that dealt with **generic "SoundCloud rapper" themes** demonstrated increased odds of being predicted as having **Low popularity**.

Mentioning **Drake** increased the odds of predicting a low popularity song by a staggering 2.2 times.

## Money Gucci Green Paper



# The App

Allows for user input of lyrics and **produces a prediction** on the songs popularity.

Users can also get a more **in-depth look into the dataset**.

The screenshot shows a dark-themed Streamlit application. At the top, there's a navigation bar with three items: 'Introduction', 'Predicting Popularity' (which is highlighted in a light gray box), and 'Exploring Lyrics'. Below the navigation is a main content area with the title 'Predicting Spotify Popularity'. The text reads: 'Do you have what it takes to write a popular song? Well, here's your chance to test your skills. Input some lyrics below and lets see whether or not it will be a hit.' At the bottom of the content area, it says 'Made with Streamlit'.

The  
End

# Supplementary Slides

# Text Cleaning

Removed **tags identifying parts of a song** (chorus, verse, hook, etc.).

Removed **all punctuation and uncapitalized** all words.

Removed stop words. These words are considered **insignificant in NLP** (the, a, etc.).

Stemmed the lyrics. Convert words to their root. Ex. **Running** is converted to **Run**.

# Text Transformations

## Vectorizers

**CountVectorizers** - Count how many **times a word occurs in a lyric** and assigns that number for that word. Do this with all words that appear in a certain percentage of songs.

**TF-IDF** - **Assigns the frequency of a word instead of a count**. Also includes a factor based on how uncommon a word is. The **more uncommon a word** is the larger the value will be.

# Text Transformations

## Embeddings

**LexVec Embeddings** - Vectors that represent complex relationships between words. Model was trained on **English Wikipedia 2015**.

**Ada Embeddings** - OpenAI second generation text embedding. This model converts text into a ~1500 dimensional vector that **captures semantics and other relationships** between words.