Lyrics Matter: Exploiting the Power of Learnt Representations for Music Popularity Prediction

Anonymous ACL submission

Abstract

Accurately predicting the popularity of a music is a critical challenge in the music industry given the potential benefits to artists, producers and streaming platforms. Historically, research on music success was focused on factors such as audio features and extrinsic metadata (e.g., artist demographics, listener trends), or advancing prediction model architecture. This paper addresses the under-explored area of exploiting lyrical content to predict music popularity. We present a novel automated pipeline that uses LLMs to extract mathematical representations from lyrics, capturing their semantic and syntactic structure, while preserving sequential information. These features are then integrated into a novel multimodal architecture, HitMusicLyricNet, combining audio, lyrics, and social metadata for predicting popularity score. Our method outperforms the available baseline in end-to-end deep learning architecture for music popularity prediction on the SpotGenTrack (SPD) dataset. We achieve an overall 9% and 20% improvement in prediction model performance metrics MAE and MSE respectively. We confirm that the improvements result from the introduction of our lyrics feature engineering pipeline (LyricsAENet) in our model architecture, HitMusicLyricNet.

1 Introduction

In 2023, the global recorded music market generated $28.6 billion[1] in revenues. With the advent of social media and streaming services, it has become increasingly difficult to define mathematical metrics for music success. Music popularity prediction could help the music industry and artists in maximizing future success of a newly released song.

Research in music popularity prediction has been driven by the advancements in machine learning with researchers applying classical ML approaches to predict popularity using acoustic features, and further with the growth of social networks, information about music consumers' tastes capturing consumer response and their evolving music preferences. Advancements in deep learning further sharpen the prediction model capability of capturing and learning complex patterns of evolving music taste, and researchers have worked on incorporating multiple modalities such as audio, lyrics and social metadata to predict song success (Zangerle et al., 2019) (Martín-Gutiérrez et al., 2020). In all these works, the popularity score is typically defined as the time the song remains on the Billboard Top charts, and the evaluation metrics used include MAE, MSE, R^2 for regression, and accuracy, precision, recall, and F1 for classification. Recent developments in large language models have led to further research in music-related fields such as recommendation systems, sentiment/emotion analysis, data augmentation, understanding and composing song lyrics, using song lyrics text as the data source (Rossetto et al., 2023), (Sable et al., 2024), (Ma et al., 2024), (Ding et al., 2024). Music Popularity Prediction research has still not fully exploited the power of lyrics in the models, while recent research have shown lyrics contributing significantly to song popularity (Yu et al., 2023). Through our work, we address the gap in the existing literature with the following chief contributions:

- **LyricsAENet**: A novel automated lyric feature extraction pipeline that uses LLMs to encode music lyrics into rich, learned representations.
- **HitMusicLyricNet**: An end to end multimodal deep learning architecture which predicts the popularity score in range (1,100) and outperformed current baseline by 9% and 20% in MAE and MSE metrics respectively .

The next section reviews related work. This

[1] IFPI Report'23

is followed by a discussion of our methods, the dataset and our experiments.

2 Related Work

Music Popularity Prediction. Studied as a classification or regression problem in a supervised learning fashion, where a model learns to predict either binary class labels (hit or no-hit) or generate a continuous popularity score. These predictions are based on a comprehensive analysis of a song's features and associated social factors. Song popularity is primarily measured using charts like Billboard [2], and UK Singles Charts[3], which rank songs based on sales, radio airplay, and streaming activity. Researchers determine success metrics based on these rankings. Other measures include economic metrics like merchandise sales, and engagement metrics such as interactions on social media and streaming services (Seufitelli et al., 2023).

Traditional research focused on using various machine learning techniques, including Logistic Regression, Decision Trees, Support Vector Machines (SVM), Bayesian Networks, Naive Bayes, Random Forest Ensemble, XGBoost, and K-Nearest Neighbors (KNN). These approaches advanced further to neural networks and deep learning techniques, building much stronger predictive models. A significant number of studies (Bischoff et al., 2009), (Dorien Herremans and Sörensen, 2014), (Zangerle et al., 2019), (Silva et al., 2022) focused on using acoustic characteristics of songs along with metadata that includes factors such as social influences. Other works such as (Dhanaraj and Logan, 2005), (Singhi and Brown, 2015b), (Martín-Gutiérrez et al., 2020) also emphasized the importance of song lyrics in determining song success using handcrafted text-based features that captured sentiment, emotions, and the syntactic structure of lyrics. These studies were often limited by their capabilities to capture central expressions of the song's lyrics.

Multiple datasets have been released to drive research further and quench the thirst of data-heavy deep learning models. This includes Million Song Dataset[4], SpotGenTrack[5], and AcousticBrainz [6] sourced from different platforms like Spotify, Billboard, Genius [7], Youtube, and others. These datasets comprise a wide range of features, from low-level features like Mel-Frequency Cepstral Coefficients (MFCCs), lyrics text, and temporal features to high-level audio features such as danceability and loudness. Additionally, they include metadata on artists, albums, genres, demographics, and other relevant information.

Learned Representations of Lyrics. Lyrics form an integral part of music and carry a deep emotional meaning, which can strongly influence how listeners feel—sometimes even more than the song's acoustic features alone (Singhi and Brown, 2015a). Yet, lyrics have often been overlooked as compared to acoustic attributes and social metrics of songs (Seufitelli et al., 2023). Earlier studies used methods like Probabilistic Latent Semantic Analysis (PLSA) (Hofmann, 1999) to capture the semantic content of lyrics, which helped researchers understand their role in defining a "hit" song (Dhanaraj and Logan, 2005). Later work moved beyond basic semantic analysis, focusing on more detailed features. For instance, (Hirjee and Brown, 2010)and (Singhi and Brown, 2014) relied on various rhyme and syllable characteristics to predict hit songs using only their lyrics, while other researchers applied Latent Dirichlet Allocation (LDA) (Blei et al., 2003) to discover thematic topics within lyrics (Ren et al., 2016).

Progress of deep learning techniques advanced the use of multimodal approaches that combine lyrics with audio and metadata, using stylometric analysis to extract lyric text features (Martín-Gutiérrez et al., 2020). Sentiment analysis also emerged as a way to glean emotional insights from lyrics when predicting popularity (Raza and Nanath, 2020). More recent research has turned to learned lyric representations, such as embeddings (Kamal et al., 2021) (McVicar et al., 2022), which offer a more robust way to capture lyrical meaning. (Barman et al., 2019) demonstrated that these distributed representations can effectively predict both genre and popularity, reducing the need for handcrafted features. Datasets such as Music4All-Onion (Moscati et al., 2022) provide lyric embeddings that make it easier to study how lyrical content relates to a song's success. Finally, a recent study found that a song's lyrical uniqueness has a significant contribution towards its popularity (Yu et al., 2023), using TF–IDF for lyric vector

[2] Billboard Hot 100
[3] Official Singles Chart Top 100
[4] Million Song Dataset
[5] SpotGenTrack
[6] AcousticBrainz
[7] Genius.com

representation; however, this approach inherently lacks the capacity to capture deeper sequential and contextual nuances, emphasizing the growing importance of learning robust, richer representations of lyrics to better understand what makes certain songs resonate with audiences.

To the best of our knowledge, there are limitations in existing literature for efficient automated lyrics feature extraction that are expressive and capture the underlying complexity of song lyrics. Thus, we have built a novel pipeline to exploit the power of Large Language Models. It has the potential to provide rich lyric representations that encapsulate both semantic and syntactic understanding, while preserving the sequential structure of the lyrics.

3 Methodology

In this section, we provide the theoretical foundation of our approach. We begin by defining the problem of music popularity prediction in mathematical equations. This is followed by explaining the baseline approach and its implementation, including details of the dataset. Finally, we present a formal description of our proposed architecture.

3.1 Problem Formulation

Given a song S, its features are represented in a multi-dimensional space $X \in \mathbb{R}^d$, which comprises three key modalities: audio waveform $w \in \mathbb{R}^k$, lyrical text $l \in \mathbb{R}^m$, and metadata attributes $m \in \mathbb{R}^p$, where $d = k + m + p$ represents the total dimensionality of our feature space. Our primary objective is to extract meaningful features from the song lyrics to effectively encode each song into a unique vector representation. Next, the prediction task is formulated as learning a mapping function $f : X \rightarrow Y$, where we minimize the expected prediction error: $\mathbb{E}[(f(X) - Y)^2]$ across the training distribution. Here, $Y \in \mathbb{R}$ represents the continuous popularity score.

3.2 Baseline Methodology

We trained *HitMusicNet*, a multimodal end-to-end Deep Learning architecture as proposed by (Martín-Gutiérrez et al., 2020) and validated the results using the SpotGenTrack Popularity Dataset (SPD). The model outputs a popularity score between 1 and 100, using audio features, text features, and metadata containing artist and demographic information as inputs. A complete description of the feature set used is provided in Table 1.

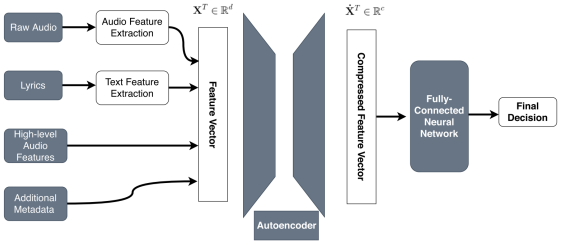

Figure 1: Diagram of the HitMusicNet pipeline outlining the principal functionalities and data components. Image src (Martín-Gutiérrez et al., 2020).

Feature Type	Features
Text Features	Sentence count, Avg words, Word count, Avg syllables/word, Sentence similarity, Vocabulary wealth
High-Level Audio	Danceability, Energy, Key, Loudness, Mode, Speechiness, Acousticness, Instrumentalness, Liveness, Valence, Tempo, Duration, Time Signature
Low-Level Audio	Mel-spectrogram, MFCCs, Tonnetz, Chromagram, Spectral Contrast, Centroid, Bandwidth, Zero-Crossing Rate
Meta-Data Features	Artist followers, Artist popularity, Available markets

Table 1: Summary of features used in the HitMusicNet architecture (Martín-Gutiérrez et al., 2020).

HitMusicNet architecture as shown in Fig 1, employs an autoencoder for feature compression through two encoder layers with dimensions $d/2$ and $d/3$, followed by a bottleneck layer of $d/5$. Each layer uses ReLU activation, and the output layer employs a sigmoid activation for reconstruction. The autoencoder was trained using the Adam optimizer and an MSE loss function. The compressed features are then passed through a fully connected neural network with four layers, where the number of neurons in each layer is scaled by factors $\alpha = 1$, $\beta = 1/2$, and $\gamma = 1/4$. The model is trained using an 80%-20% train-test split with stratified cross-validation (SCV) using $k = 5$. These settings helped us in effectively replicating the baseline results on the SPD dataset.

3.3 Dataset

The SpotGenTrack Popularity Dataset (SPD) proposed by (Martín-Gutiérrez et al., 2020) and used in this research contains 101,939 tracks, 56,129 artists, and 75,511 albums sourced using Spotify and Genius APIs. The data was gathered from 26 countries where Spotify is available, including the top 50 playlists per category for each country.

Popularity scores for tracks range between 1 and 100 and are provided by Spotify based on internal metrics. The scores follow a Gaussian distribution with $\mu = 40.02$ and a standard deviation of $\sigma = 16.79$. The dataset contains low-level audio features extracted using audio waveform, text features extracted using stylometric analysis of lyrics. High-level audio features and metadata are sourced from Spotify. The lyrics in the SPD dataset had to be cleaned and pre-processed to align with the objectives of this research. We inspected long tails of lyrics length distribution and observed that extremely short or long entries typically contained irrelevant content such as random numbers, out-of-context text, or placeholder text. Based on this analysis, we retained songs with lyrics lengths between 100 and 7000 characters. Furthermore, we filtered the dataset to include only English lyrics, which comprised approximately 60% of the total data. These steps resulted in a clean dataset comprising 51,319 tracks, 30,024 unique artists, and 39,371 unique albums. The resulting popularity distribution, as shown in Fig 2, had $\mu = 41.11$ and a standard deviation of $\sigma = 17.51$, retaining original data characteristics.

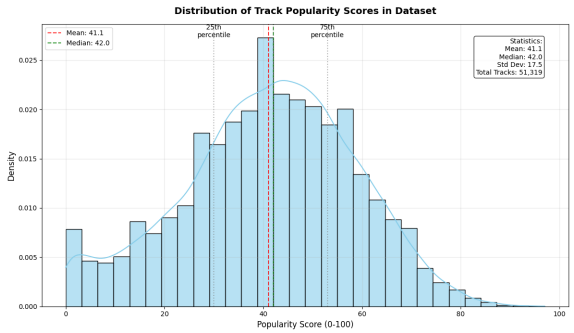

Figure 2: Popularity Distribution in cleaned SpotGenTrack(SPD) with $\mu = 41.11$ and a standard deviation of $\sigma = 17.51$.

3.4 HitMusicLyricNet

This section details our proposed HitMusicLyricNet, an end-to-end multimodal deep learning architecture built upon the foundation of HitMusicNet. HitMusicLyricNet comprises of three key components: AudioAENet, LyricsAENet, and MusicFuseNet. AudioAENet compresses the low-level audio features. LyricsAENet compresses the lyric embeddings into a fixed-size representation using an Autoencoder, thereby encoding information while reducing noise. MusicFuseNet then combines these compressed audio and lyric representations with metadata and high-level audio features as described in Table 1.

In the HitMusicNet architecture, a single autoencoder compressed the combined feature vector of audio, lyrics, and metadata. We hypothesize that this can lead to information loss, particularly for the less abundant lyrics and metadata features. We believe that lyrics and metadata features should be fed directly into the popularity prediction network to retain their predictive power for song popularity. Furthermore, our reasoning behind the new approach of introducing distinct Autoencoders for audio and lyrics is based on the bipolar and directional nature of lyrics embeddings, requiring a different architecture for compression(Bałazy et al., 2021).

3.4.1 AudioAENet

The Autoencoder used for compression has a similar architecture to that of MusicAENet, but takes in only low-level audio features as described in Table 1 for compression. For input dimension d = 209, it gradually compresses the data to dimension d/2, d/3, and d/5. The output layer employs a sigmoid activation for reconstruction, whereas all remaining layers use ReLU activation functions. The model is trained using the Adam optimizer with a MSE loss function, achieving a loss value in the range of 1e-5, indicating negligible loss in compression.

3.4.2 LyricsAENet

LyricsAENet implements a tied-weights autoencoder architecture (Li and Nguyen, 2019) designed to reduce parameter size and risk of overfitting. Compressing lyric embeddings is susceptible to overfitting due to high dimensionality. The encoder follows a progressive compression with the following dimensions (d/2, d/4, d/8), followed by bottleneck layers (d/12 or d/16). The decoder mirrors the structure in reverse order, utilizing the tranpose of the encoder weight. The progressive dimensional reduction is designed to minimize reconstruction losses in compressed embeddings extracted out of language models and LLMs such as BERT (Devlin et al., 2019), LLaMA 3 Herd (Grattafiori et al., 2024), and OpenAI's embedding models[8].

We use Scaled Exponential Linear Unit (SELU) (Klambauer et al., 2017) as the activation function for its self-normalizing characteristics and the ability to handle the bipolar nature of embeddings. Comparative analyses include alternate activation

[8] Open AI text Embedding Model

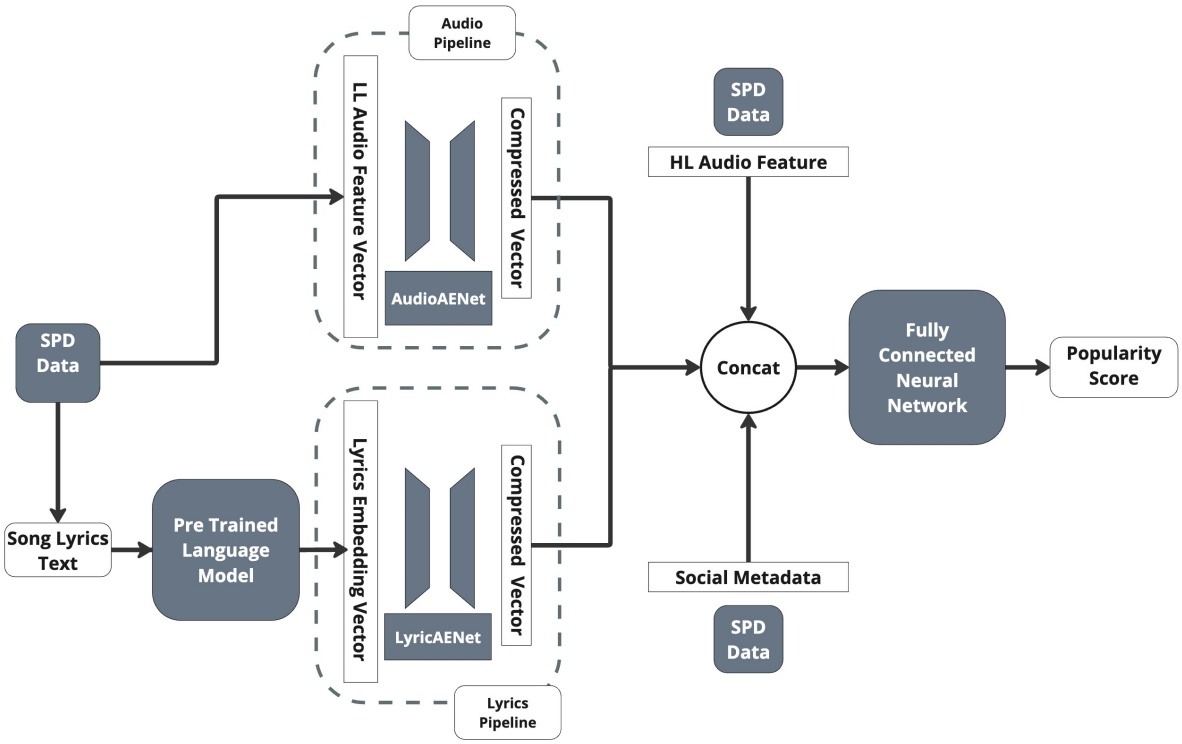

Figure 3: Block-Schema of the *HitMusicLyricNet* architecture comprising of two Autoencoders and a Fully Connected NN predicting popularity score. 'HL' stands for high-level and 'LL' stands for low-level.

functions such as the Sigmoid Linear Unit (SiLU) (Elfwing et al., 2018) and the Gaussian Error Linear Unit (GELU) (Hendrycks and Gimpel, 2016). LyricsAENet was trained using the Adam optimizer with a MSE loss function, achieving loss values of approximately 1e-5. To further refine the training process, we incorporated a directional loss function inspired by (Bałazy et al., 2021) to preserve the directional characteristics of the embeddings during compression. This combined loss function is defined as:

$$L(Y, \bar{Y}) = \alpha_1 \cdot \text{MSE}(Y, \bar{Y}) + \alpha_2 \cdot CD(Y, \bar{Y}), \quad (1)$$

where $\text{MSE}(Y, \bar{Y})$ represents the Mean Squared Error. $CD(Y, \bar{Y})$ denotes the Cosine Distance, which captures the angular similarity between the vectors Y and $\bar{Y}$. The constants α_1 and α_2 control the relative importance of the two loss terms.

3.4.3 MusicFuseNet

We employ a similar architecture configuration as MusicPopNet by (Martín-Gutiérrez et al., 2020) for our MusicFuseNet. It uses a concatenation of compressed audio feature vectors from AudioAENet, compressed lyrics embeddings vectors from LyricsAENet, high-level audio features and metadata as mentioned in Table 1. The output of this neural net is a popularity score in the range [0, 1]. The architecture consists of a fully connected network with scaling parameters of (1, 1/2, 1/3) and ReLU activation functions, followed by a Sigmoid activation in the final layer, as empirically validated by (Martín-Gutiérrez et al., 2020). To train the model, we used the Adam optimizer with an MSE loss function and applied dropout regularization to mitigate overfitting.

4 Experiments and Results

Using the **Code**[9] to implement HitMusicNet and selecting the configuration details described in Section 3.2, we trained HitMusicNet on the SPD dataset with an 80-20 split. To replicate the results obtained by (Martín-Gutiérrez et al., 2020), we employed Stratified Cross-Validation (SCV) with k=5 folds and used MAE and MSE as performance metrics. As Table 4 shows, we achieved similar results on all performance metrics, validating our training and testing strategy. Further, removing the lyrics text features proposed by (Martín-Gutiérrez et al., 2020) did not degrade the metrics, so we dropped those features for further experiments.

To train HitMusicLyricNet, we extracted lyric

[9] Github: HitMusicNet

LyricsAENet Config	MAE (Train)	MAE (Val)	MAE (Test)
SELU, MSE	**0.0769**	**0.0746**	**0.0775**
SiLU, MSE	0.0736	0.0731	0.0790
GELU, MSE	0.0740	0.0731	0.0792
SELU, Dir.	0.0741	0.0740	0.0799

Table 2: Results of training and testing HitMusicLyricNet on cleaned SPD data with various LyricAENet configurations (activation function, loss function), using BERT Large embeddings throughout. 'Dir' indicates directional loss 1.

Embeddings Model	MAE (Train)	MAE (Val)	MAE (Test)
BERT large	**0.0793**	**0.0784**	**0.0786**
Llama 3.1 8B	0.0774	0.0759	0.0795
Llama 3.2 1B	0.0775	0.0754	0.0800
Llama 3.2 3B	0.0781	0.0766	0.0798
OpenAI Small	0.0746	0.0738	0.0788
OpenAI Large	**0.0761**	**0.0743**	**0.0772**

Table 3: Results of training and testing HitMusicLyricNet on cleaned SPD data with different lyric embeddings sent to LyricAENet (Selu activation, MSE loss).

Model Config	Dataset Config	MSE (Train)	MSE (Val)	MAE (Train)	MAE (Val)	MAE (Test)
HitMusicNet	SPD	0.0116	0.0115	0.0836	0.0851	0.0862
HitMusicNet w/o lyrics	SPD	0.0114	0.0116	0.0843	0.0859	0.0870
HitMusicLyricNet	*SPD	**0.0095**	**0.0091**	**0.0761**	**0.0743**	**0.0772**
HitMusicLyricNet w/o lyrics	*SPD	0.0109	0.0113	0.0818	0.0841	0.0852

Table 4: Performance comparisons with the baseline (HitMusicNet) on SPD and SPD* data respectively, where SPD* denotes cleaned SPD data. Here, we report the best results from Table 3.

embeddings from language models. For open-source models (BERT, Llama), we downloaded vanilla weights from Hugging Face[10] and loaded its vanilla configuration. We used Nvidia A100 GPU for compute requirements. After tokenizing lyrics, we forward-passed them through each model, extracted the last hidden-layer states, and applied max/mean pooling to obtain fixed-size vectors for our Autoencoder. Specifically for BERT, we considered mean pooling and concat (max + CLS token). To get embeddings from OpenAI text models, we used the API endpoint, costing $3 for the small model and $6 for the large. We then studied LLM model architecture and its training corpus effects on music popularity prediction with BERT, BERT Large, Llama 3.1 8B, Llama 3.2 1B, Llama 3.2 3B, and OpenAI text embeddings (small and large).

After extracting these embeddings, we examined different activation layers (Selu, Silu, Gelu) for embedding compression using LyricsAENet and introduced a directional loss function with $\alpha_1 = 0.5$ and $\alpha_2 = \frac{0.5}{5}$ as suggested by (Bałazy et al., 2021), alongside our standard MSE loss for LyricsAENet, to see their impact on the HitMusicLyricNet performance metric MAE. As reported in Table 2, using SELU with the MSE loss function in LyricsAENet yielded the least MAE error while training HitMusicLyricNet on popularity prediction. Directional loss produced comparable metrics but not enough improvement to be included further. Other activation functions performed closely, but for simplicity and observing 1–2% randomness error, we proceeded with SELU and MSE.

Next, we compressed embeddings for different LLM models. While we experimented with two variants of BERT (small and large) and considered mean embeddings and concat (max + CLS token) embeddings, here we only report results for BERT large with mean embeddings, as it yielded the best results as seen in Table 3. All the Llama variants had very close performance metrics, whereas the OpenAI large text embedding model surpassed all of them. We attribute these small differences ($\sim 2\%$ variation) in HitMusicLyricNet's performance to variations in each model's training data and architecture, since none was specifically trained for our downstream task, leading to large differences in rich embedding representation.

Hence with HitMusicLyricNet used the OpenAI large text embeddings and the SELU activation with MSE loss function in lyricsAENet. Overall, we achieved close to a 9% improvement compared to the SOTA architecture, despite training on 40% less data. Dropping the lyrics feature pipeline and retraining and testing HitMusicLyricNet led to performance metrics comparable to that of HitMusicNet, validating the effectiveness of our proposed lyric feature extraction pipeline using LLMs and the overall advancements in music popularity prediction pipeline.

[10] Hugging Face

5 Conclusion and Future Work

The work presented in this paper showcases the power of leveraging lyrics to predict the popularity of a song, with the help of LLMs with capabilities of capturing the deeper meaning of sentences using embeddings. We believe that advancements in music-aware language models will lead to more explainable and expressive lyric features based on domain-specific knowledge. This research presented a novel architecture, HitMusicLyricNet, along with an ablation study. HitMusicLyricNet beats the SOTA by 9% by incorporating lyric embeddings and improving upon the SOTA architecture. With advancements in compression techniques and multimodal learning architecture, we believe accuracy and commercial use can be improved. Furthermore, with advancements in audio representation learning using neural audio codecs, richer music audio representations can be scoped into the study. Current research aggregates features over an entire song. However, contemporary phenomena of virality suggest that local features within different musical segments need to be studied deeply and cannot be ignored given the micro-content consumption driven by platforms like Instagram and SnapChat.

6 Limitation

Our research results are potentially limited to the music genres represented in our dataset and may not generalize across genres, demographics, and cultural contexts. Some limitations arise as a result of the choice of dataset used in our study, SpotGenTrack. The findings are highly dependent on the quality and size of the SpotGenTrack dataset. The dataset has been cleaned to filter out lyrics that are not in the English language. Though this reduced the size of the raw dataset by 40%, it limits the model's ability to be generalized across different languages and associated cultural contexts. The use of LLMs such as BERT and Llama 3 in our model will lead to a lack of domain-specific context, as horizontal LLMs are not typically trained or fine-tuned on music-focused data. While adequate measures have been made to address the risk of overfitting, the risk cannot be completely eliminated due to the high dimensionality of the data. The lyric embedding vectors are flowing downstream and are used to predict the popularity of a song. Finally, since we are assessing the quality of lyric embeddings using the performance metrics of downstream tasks (music popularity prediction), this requires a further examination to evaluate the intrinsic qualities of lyric embeddings vector in capturing rich representation. We are limited by the explanability of our lyrics feature vector which can be further improved by using fine-tuned LLMs for music lyric analysis and explainable feature extraction.

References

Klaudia Bałazy, Mohammadreza Banaei, Rémi Lebret, Jacek Tabor, and Karl Aberer. 2021. Direction is what you need: Improving word embedding compression in large language models. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 322–330, Online. Association for Computational Linguistics.

Manash Pratim Barman, Kavish Dahekar, Abhinav Anshuman, and Amit Awekar. 2019. It's only words and words are all i have. In *Advances in Information Retrieval: 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14–18, 2019, Proceedings, Part II*, page 30–36, Berlin, Heidelberg. Springer-Verlag.

Kerstin Bischoff, Claudiu S. Firan, Mihai Georgescu, Wolfgang Nejdl, and Raluca Paiu. 2009. Social knowledge-driven music hit prediction. In *Advanced Data Mining and Applications*, pages 43–54, Berlin, Heidelberg. Springer Berlin Heidelberg.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ruth Dhanaraj and Beth Logan. 2005. Automatic prediction of hit songs. In *International Society for Music Information Retrieval Conference*.

Shuangrui Ding, Zihan Liu, Xiaoyi Dong, Pan Zhang, Rui Qian, Conghui He, Dahua Lin, and Jiaqi Wang. 2024. Songcomposer: A large language model for lyric and melody composition in song generation. *arXiv preprint arXiv:2402.17645*.

David Martens Dorien Herremans and Kenneth Sörensen. 2014. Dance hit song prediction. *Journal of New Music Research*, 43(3):291–302.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. 2018. Sigmoid-weighted linear units for neural network

function approximation in reinforcement learning. *Neural networks*, 107:3–11.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrst-

edt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Hussein Hirjee and Daniel G. Brown. 2010. Rhyme analyzer: An analysis tool for rap lyrics. In *International Society for Music Information Retrieval Conference (ISMIR)*. ISMIR. Late-Breaking Demo.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 50–57, New York, NY, USA. Association for Computing Machinery.

J. Kamal, P. Priya, M. R. Anala, and G. R. Smitha. 2021. A classification based approach to the prediction of song popularity. In *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*, pages 1–5. IEEE.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-normalizing neural networks. *Advances in neural information processing systems*, 30.

Ping Li and Phan-Minh Nguyen. 2019. On random deep weight-tied autoencoders: Exact asymptotic analysis, phase transitions, and implications to training. In *International Conference on Learning Representations*.

Yinghao Ma, Anders Øland, Anton Ragni, Bleiz MacSen Del Sette, Charalampos Saitis, Chris Donahue, Chenghua Lin, Christos Plachouras, Emmanouil Benetos, Elona Shatri, et al. 2024. Foundation models for music: A survey. *arXiv preprint arXiv:2408.14340*.

David Martín-Gutiérrez, Gustavo Hernández Peñaloza, Alberto Belmonte-Hernández, and Federico Álvarez García. 2020. A multimodal end-to-end deep learning architecture for music popularity prediction. *IEEE Access*, 8:39361–39374.

Matt McVicar, Bruno Di Giorgi, Baris Dundar, and Matthias Mauch. 2022. Lyric document embeddings for music tagging.

Marta Moscati, Emilia Parada-Cabaleiro, Yashar Deldjoo, Eva Zangerle, and Markus Schedl. 2022. Music4all-onion – a large-scale multi-faceted content-centric music recommendation dataset. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, page 4339–4343, New York, NY, USA. Association for Computing Machinery.

Agha Haider Raza and Krishnadas Nanath. 2020. Predicting a hit song with machine learning: Is there an apriori secret formula? In *2020 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA)*, pages 111–116.

Jing Ren, Jialie Shen, and Robert J. Kauffman. 2016. What makes a music track popular in online social networks? In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, page 95–96, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Federico Rossetto, Jeffrey Dalton, and Roderick Murray-Smith. 2023. Generating multimodal augmentations with llms from song metadata for music information retrieval. In *Proceedings of the 1st Workshop on Large Generative Models Meet Multimodal Applications*, LGM3A '23, page 51–59, New York, NY, USA. Association for Computing Machinery.

Prof. R.Y. Sable, Aqsa Sayyed, Baliraje Kalyane, Kosheen Sadhu, and Prathamesh Ghatole. 2024. Enhancing music mood recognition with llms and audio signal processing: A multimodal approach. *International Journal for Research in Applied Science and Engineering Technology*.

Danilo B. Seufitelli, Gabriel P. Oliveira, Mariana O. Silva, Clarisse Scofield, and Mirella M. Moro. 2023. Hit song science: a comprehensive survey and research directions. *Journal of New Music Research*, 52:41 – 72.

Mariana O. Silva, Gabriel P. Oliveira, Danilo B. Seufitelli, Anisio Lacerda, and Mirella M. Moro. 2022. Collaboration as a driving factor for hit song classification. In *Proceedings of the Brazilian Symposium on Multimedia and the Web*, WebMedia '22, page 66–74, New York, NY, USA. Association for Computing Machinery.

Abhishek Singhi and Daniel G. Brown. 2014. Hit song detection using lyric features alone. In *International Society for Music Information Retrieval Conference (ISMIR): Late-Breaking Demo*, Waterloo, Canada. University of Waterloo, Cheriton School of Computer Science, ISMIR. Late-Breaking Demo.

Abhishek Singhi and Daniel G. Brown. 2015a. Can song lyrics predict hits? In *Proceedings of the 11th International Symposium on Computer Music Multidisciplinary Research (CMMR)*, pages 457–471.

Anurag Singhi and David G. Brown. 2015b. Can song lyrics predict hits. In *International Symposium on Computer Music Multidisciplinary Research*, pages 457–471. The Laboratory of Mechanics and Acoustics.

Yulin Yu, Pui Yin Cheung, Yong-Yeol Ahn, and Paramveer S. Dhillon. 2023. Unique in what sense? heterogeneous relationships between multiple types of uniqueness and popularity in music. *Proceedings of the International AAAI Conference on Web and Social Media*, 17(1):914–925.

Eva Zangerle, Michael Vötter, Ramona Huber, and Yi-Hsuan Yang. 2019. Hit song prediction: Leveraging low- and high-level audio features. In *International Society for Music Information Retrieval Conference*.