Lyrics for success: embedding features for song popularity prediction

Giulio Prevedello[1,2,*], Inès Blin[1,3,*], Bernardo Monechi[1], Enrico Ubaldi[1]

[1]Sony CSL Paris Research, 6 Rue Amyot, 75005, Paris, France
[2]Enrico Fermi's Research Center (CREF), via Panisperna 89/A, 00184, Rome, Italy
[3] Vrije Universiteit Amsterdam, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands
[*] These authors contributed equally.

Correspondence:{giulio.prevedello, ines.blin}@sony.com,

Abstract

Accurate song success prediction is vital for the music industry, guiding promotion and label decisions. Early, accurate predictions are thus crucial for informed business actions. We investigated the predictive power of lyrics embedding features, alone and in combination with other stylometric features and various Spotify metadata (audio, platform, playlists, reactions). We compiled a dataset of 12,428 Spotify tracks and targeted popularity 15 days post-release. For the embeddings, we used a Large Language Model and compared different configurations. We found that integrating embeddings with other lyrics and audio features improved early-phase predictions, underscoring the importance of a comprehensive approach to success prediction.

1 Introduction

Predicting music release success is crucial for the music industry and influences artists' signings and careers. Strategies are planned before release and adjusted based on success expectations (Steininger and Gatzemeier, 2019). Post-release efforts could target demographics that may not have initially responded. As depicted in Figure 1, a song reaches peak audience within two weeks of release, going from novelty to stabilized exposure.

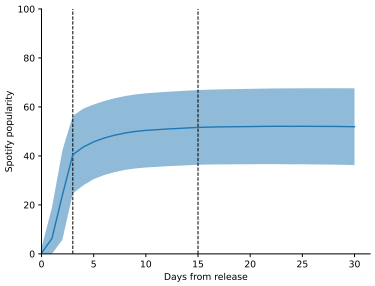

Figure 1: Daily average Spotify popularity scores since song release, with the daily mean within ± 1 standard deviation. Vertical lines: day 3 ("reactions" features, left) and day 15 (prediction target, right).

A song's lifecycle include production, release planning and implementation, and post-release reactions. The further in the lifecycle, the more information can gather to improve a song's success prediction. Audio and lyric data are the primary information available from the early stages. While much research in Music Information Retrieval (MIR) has focused on utilizing audio data to extract predictive features (Zangerle et al., 2019), less attention has been given to leveraging lyrics (Arora and Rani, 2024; Singhi and Brown, 2014).

We investigate the predictive power of lyrics features in the context of song success prediction. We first compile a dataset of 12,428 Spotify tracks and set the popularity at day 15 as the prediction target, which is based on recency and relative quantity of plays (Spotify, 2023) thus a good proxy of actual streams. We split our data into training, validation, and test sets in chronological order. To the best of our knowledge, no previous work has framed music success in this way. Second, we combine Large Language Models (LLMs) lyrics embeddings with stylometric features, and integrate them with other features to train regression models. Third, we perform an evaluation to compare the efficacy of using lyrics features alone versus integrating them with other features across the phases of the song's lifecycle. LLM embeddings improve the contribution of lyrics features already after the song production phase. We make our code openly available[1].

2 Related Work

Hit Song Science aims to predict whether a song can attain a "hit" status based on song features extracted by MIR techniques (Dhanaraj and Logan, 2005; Pachet, 2012). This has sparked debates about its efficacy (Pachet and Roy, 2008; Ni et al., 2011), yet subsequent work have tack-

[1]https://github.com/SonyCSLParis/foremusic-nlp

Group	Availability	#	Features
Audio	Post-production	14	acousticness, danceability, duration ms, energy, instrumentalness, liveness, loudness, speechiness, tempo, valence, album type (album, single or compilation), explicit (true or false), genre class, music style
Platform	Release planning	8	number of countries where song is available at release, days of delay between release and appearance on playlists, release month, release year, release week, release weekday, artist popularity at release, artist followers at release
Playlists	Shortly before release	5	followers from largest listing playlist, song position in largest listing playlist, sum over days of followers from largest listing playlist, sum of followers over all listing playlists, number of listing playlists at release
Reactions	Post-release	4	track popularity at day 0, 1, 2, 3 from release

Table 1: Features from Spotify metadata, grouped by domain and ordered by availability time.

led the challenge of music popularity prediction by framing it as a **classification task** of "hit" or "miss" based on: rankings in music charts like Billboard (Singhi and Brown, 2014); listing in playlists (Araujo et al., 2020); or categorisation of levels of popularity (Sharma et al., 2022; Yee and Raheem, 2022).

Music success prediction has also been tackled as a **regression problem**, based on the Spotify popularity scores (Spotify, 2023) from 0 to 100. XGBoost (Chen et al., 2015) has been praised for its efficacy (Xing, 2023). Other methods included lyrics features (Martín-Gutiérrez et al., 2020), genre-based and cluster-based approaches (Agarwal et al., 2023), and Vector Autoregressive model (Machmudin et al., 2023). We refer to Arora and Rani (2024) for a comparative analysis.

Lastly, we highlight the work of Berjamin et al. (2024) in which **features are grouped** in song intrinsic (e.g., from audio), song extrinsic (e.g., about the release context) and crowdsourced opinions. In turn, we group features by their availability over time. Our research emphasizes the utility of LLM embeddings from lyrics, by evaluating them alongside stylometric features (Zangerle et al., 2018). To the best of our knowledge, we are the first ones to use lyrics embeddings to predict music popularity within a chronological regression framework.

3 Data Collection and Preprocessing

We monitored two sets of Spotify playlists: the first, relevant for the French market, of 299 playlists from 22-07-2019 to 17-09-2022; the second, relevant to the UK market, of 235 playlists from 24-11-2020 to 03-02-2023. We kept track of the songs that were released and listed during this time, totalling 24,266 tracks. We stored the metadata made available through Spotify's API. We kept evolving data (like song popularity, artist's followers, positions in playlists, etc.) recording their dynamics daily. For the lyrics, we queried the title and artist's name through the Genius API (Miller, 2024).

We focused on success prediction for new releases, which is more relevant to the music industry. By tracking song's popularity daily, we set the regression target for success as the popularity on the 15th day after release, uniformly for every song. Figure 1 shows that, by that day, novelty has faded and popularity stops growing. Tracks with no value for the popularity target (due to faulty or late collection) were removed. After manually inspecting randomly sampled data, lyrics were automatically preprocessed by: cleaning recurrent artifacts in the text; removing tracks with no English lyrics. For language extraction, a transformer model for 51 language classification was used (Conneau, 2019). 12,428 tracks remained after preprocessing.

4 Approach

Feature extraction. Table 1 shows the **features we extracted from Spotify**, grouped by domain specificity and ordered by their availability in the song's lifecycle: *Audio* features cover sound and intrinsic song properties; *Platform* features include platform's metadata about the context of publication; *Playlists* features are a summary statistics of the monitored playlists in which the song was listed; *Reactions* features are the level of song popularity in Spotify as in the four days after release. We record features at release date, and use those as a proxy for the information that is available before release. This concerns only time-varying features, such as playlist and artist followers, which are rather stable and therefore negligibly impacted.

Table 2 summarises the **stylometric lyrics features** we re-used from Zangerle et al. (2018) and Martín-Gutiérrez et al. (2020). For the **embedding features**, we used sentence embeddings (Reimers and Gurevych, 2019a) designed to improve the sentences semantic representations. We selected the tokenizer and the model *all-mpnet-base-v2* (Reimers and Gurevych, 2019b; Hugging-

Type	#	Features
Lexical	14	token count, unique token ratios, avg. token length, repeated token ratio, hapax dis-/tris-/legomenon, unique tokens/line, avg. tokens/line, line counts, punctuation and digit ratios, stop words ratio, stop words/line
Linguistic	1	lemma ratio
Semantic	4	VADER scores (4)
Syntactic	3	pronoun frequency, past tense ratio

Table 2: Stylometric lyrics features used. We used whitespace for token separation.

face, 2024) that was fine-tuned on the MPNet architecture (Song et al., 2020) and that stood out as one of the top-performing sentence embedding models (Sbert, 2024). The resulting embeddings have 768 dimensions, which we sought to reduce to a size comparable to the number of other features used in the prediction of popularity. To do so, we either included a dimensionality reduction step or applied the UMAP method (McInnes et al., 2018).

We explored (i) fine-tuning the embeddings for popularity prediction by adding a regression layer on top of the original language model to directly optimize for predicting popularity scores; (ii) fine-tuning the LLM through unsupervised Masked Language Modeling (MLM), thus continuing to train the model to refine its contextual understanding.

We identified six strategies for lyrics embeddings: 1) `b`, embeddings from the pre-trained `all-mpnet-base-v2` model; 2) `b-reg`, `b` + fine-tuning for regression; 3) `b-red-reg` `b` + dimensionality reduction + fine-tuning for regression; 4) `ft`, embeddings from the fine-tuned `all-mpnet-base-v2` model; 5) `ft-reg`, `ft` + fine-tuning for regression); 6) `ft-red-reg`, `ft` + dimensionality reduction + fine-tuning for regression.

Regression Model. We used LightGBM (LGBM), a tree-based model built on gradient boosting (Ke et al., 2017). The model was trained with 5-fold cross-validation (Pedregosa et al., 2011), with parameters: $learning\_rate = 0.001$, $n\_estimators = 10,000$, and hyperparameter grids: $max\_depth \in 6, -1$, $num\_leaves \in 40, 60$, $colsample\_bytree \in 0.5, 0.7, 1$. Feature importance is measured by the frequency of a feature's use in LGBM's decision splits.

5 Experimental Set-Up

We first select the embedding strategy that best predicts popularity, then measure the contributions of each group of features to the popularity prediction.

We divided the data into train, validation and test sets, in a $80/10/10$ split based on time of release. Songs in the training set were released before those in validation, which were released before those in the test, to reflect a realistic scenario. After preprocessing, the train, validation and test sets contained $9,812$, $1,391$ and $1,225$ songs respectively.

We used both the training and validation sets to fine-tune the LLM with MLM, as this method is unsupervised. For the regression task, we fine-tuned the embeddings using the training set and evaluated them on the validation set. We reduced the embeddings to dimensions 5, 10 or 20, using either Euclidean or cosine distance for UMAP. We assessed ten different embeddings for each of the three sizes: `b`, `b-reg`, `ft`, `ft-reg` with UMAP l2 or UMAP cosine ($4 \cdot 2$ models); `b-red-reg` and `ft-red-reg` (2 models). 30 embeddings were compared on the training and validation sets joint together.

For comparability with sizes from other feature groups (see Table 1), we focused on the 10-dimensional embeddings. The best-performing was used to assess how well lyrics features predict popularity when used alone and jointly in the four different stages of the song life. We compared the performance of the LGBM model on the test set, using Spotify features with and without lyrics features (stylometric alone, embedding alone, stylometric+embedding). We also compared on the lyrics features only. 19 LGBM models were trained. We used Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R2) to assess the LGBM models.

6 Experimental Results

6.1 Embedding selection

Embedding dimension		5			10			20		
Embedding	UMAP	MAE	RMSE	R2	MAE	RMSE	R2	MAE	RMSE	R2
`b`	Euc.	9.61	12.2	0.27	9.39	11.92	0.3	9.23	11.75	0.32
`b`	Cos.	9.7	12.31	0.25	9.34	11.86	0.31	9.15	11.63	0.33
`ft`	Euc.	9.4	11.99	0.29	9.15	11.62	0.34	9.02	11.5	0.35
`ft`	Cos.	9.42	12.01	0.29	9.19	11.71	0.33	8.95	11.42	0.36
`b-reg`	Euc.	8.97	11.5	0.35	8.83	11.28	0.37	8.63	11.04	0.4
`b-reg`	Cos.	8.97	11.49	0.35	8.82	11.3	0.37	8.66	11.07	0.4
`ft-reg`	Euc.	8.75	11.22	0.38	**8.56**	**10.98**	**0.41**	8.35	10.72	0.44
`ft-reg`	Cos.	**8.69**	**11.16**	**0.39**	8.57	10.99	**0.41**	8.34	10.7	0.44
`b-red-reg`	None	9.04	11.52	0.35	8.82	11.22	0.38	**7.85**	**10.17**	**0.49**
`ft-red-reg`	None	9.46	12.01	0.29	9.51	12.07	0.28	10.14	12.8	0.19

Table 3: Scores from LGBM models for popularity regression, cross-validated on joint train and validation sets. Euc.: Euclidean, Cos.: Cosine.

Table 3 presents the scores of the LGBM models, cross-validated on the training and validation sets. There is a minor difference between the Euclidean and the cosine distances for the UMAP dimension-

Spotify features	None			Audio			+Platform			+Playlists			+Reactions		
Lyrics features	MAE	RMSE	R2	MAE	RMSE	R2	MAE	RMSE	R2	MAE	RMSE	R2	MAE	RMSE	R2
None				10.77	13.53	0.3	6.23	8.13	0.75	5.27	7.02	0.81	**3.57**	**5.82**	**0.87**
Stylometric	11.03	13.75	0.24	9.48	12.04	0.42	5.87	7.69	**0.76**	5.03	6.75	**0.82**	4.17	7.74	0.76
Embedding	10.56	13.23	0.25	9.27	11.76	0.41	5.8	7.59	0.75	5.03	6.7	0.81	4.11	7.77	0.74
Stylometric+Embedding	**10.3**	**13.02**	**0.28**	**9.12**	**11.57**	**0.43**	**5.77**	**7.54**	**0.76**	**4.99**	**6.66**	0.81	4.13	7.78	0.74

Table 4: Results on the test set from combination of different features. Spotify features are added cumulatively from left to right, reflecting the incremental disclosure of information through the song's lifecycle. Combining stylometric with embedding features yield moderate but consistent boost in performances in the earlier stages.

ality reduction. Fine-tuning the LLM with MLM improves performances, except when the layer for dimensionality reduction is added before the regression. The best performing strategies are `ft-reg` for dimensions 5 and 10, and `base-red-reg` for dimension 20. To make the lyrics features comparable to other features (cf. Section 5), we selected the best embedding strategy with dimension 10: **`ft-reg` + UMAP with Euclidean distance**.

Figure 2 shows that the importance ranks of embedding and stylometric features are evenly distributed, indicating that both feature sets provide complementary information.

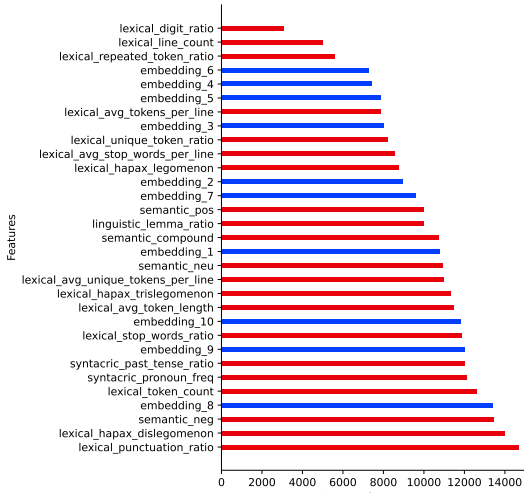

Figure 2: Feature importance of stylometric (red) and embedding (blue) features, measured by the number of times the feature is used from the LGBM model.

6.2 Lyrics with Incremental Information

Table 4 shows the scores of the LGBM models trained with incremental information, with and without lyrics features. Predictions improve as more Spotify features are included. Performance boosts added by lyrics are important for audio only, moderate for audio+platform and audio+platform+playlists. Lyrics become detrimental when reactions are included. The individual contribution of stylometric features and lyrics embedding is comparable, with the former scoring marginally, but consistently, better that the former.

Table 4 suggests that adding lyrics features improves the performance of music popularity prediction in the earlier stages of the song life, when only lyrics and audio features are available. When more features become available, the added value of the lyrics features becomes less visible or even detrimental. We contend this effect is caused by the regression model, for which only a random subsample of features are used to train each decision tree. Thus, reaction features, which are strongly predictive but few in number, become less likely to be sampled for the training of each tree as more features are included.

7 Conclusion

We incorporated lyrics features into regression models to predict the popularity of a song at day 15. We experimented with various models with stylometric and embedding-based features, selected the best ones on the training and validation sets, and evaluated how the prediction improved if we included lyrics features at different stages of the song life. We find that lyrics embeddings are useful for song popularity prediction at early stages, complementing with other features.

Future work may benefit by the rapid advances of LLMs. Multilingual models could be used to process lyrics from languages other than English. We also plan to extend the features to include text aesthetics (Kao and Jurafsky, 2012) and social media communications. There is a lack of data about marketing campaigns, despite their centrality in the business, and it would be valuable to quantify the predictive power derived from those interventions.

Spotify popularity was set as a proxy for music success, yet this metric does not offer the same resolution as actual streams, which have a richer dynamic. Other aspects could also be targeted beyond popularity, such as relative success or potential audience, providing new insights on the Science of Success (Wang et al., 2023).

References

Saket Agarwal, Jayant Goyal, Sneha Thapa, Akshada Deshpande, Aryan, and Deepa Kumari. 2023. Live Music popularity prediction using genre and clustering based classification system: A machine learning approach. In *2023 9th International Conference on Smart Computing and Communications (ICSCC)*, pages 67–71.

Carlos Vicente Soares Araujo, Marco Antônio Pinheiro de Cristo, and Rafael Giusti. 2020. A Model for Predicting Music Popularity on Streaming Platforms. *Revista de Informática Teórica e Aplicada*, 27(4):108–117. Number: 4.

Shruti Arora and Rinkle Rani. 2024. Soundtrack Success: Unveiling Song Popularity Patterns Using Machine Learning Implementation. *SN Computer Science*, 5(3):278.

Sandra Angela Berjamin, Angeli Dianne Mata, Paolo Montecillo, and Rafael Cabredo. 2024. Exploring the influence of intrinsic, extrinsic, and crowdsourced features on song popularity. In *Workshop on Computation: Theory and Practice (WCTP 2023)*, pages 395–412. Atlantis Press.

Tianqi Chen, Tong He, Michael Benesty, Vadim Khotilovich, Yuan Tang, Hyunsu Cho, Kailong Chen, Rory Mitchell, Ignacio Cano, Tianyi Zhou, et al. 2015. Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4.

A Conneau. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Ruth Dhanaraj and Beth Logan. 2005. Automatic prediction of hit songs. In *Ismir*, pages 488–491.

Huggingface. 2024. Huggingface sentence transformers. Accessed: 2024-04-01.

Justine Kao and Dan Jurafsky. 2012. A computational analysis of style, affect, and imagery in contemporary poetry. In *Proceedings of the NAACL-HLT 2012 workshop on computational linguistics for literature*, pages 8–17.

G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Daffa Adra Ghifari Machmudin, Mila Novita, and Giannina Ardaneswari. 2023. Analysis of Spotify's Audio Features Trends using Time Series Decomposition and Vector Autoregressive (VAR) Model. *Proceedings of The International Conference on Data Science and Official Statistics*, 2023(1):613–627. Number: 1.

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.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

John W. Miller. 2024. LyricsGenius: a Python client for the Genius.com API. Accessed: 2024-01-01.

Yizhao Ni, Raul Santos-Rodriguez, Matt Mcvicar, Tijl De Bie, et al. 2011. Hit song science once again a science. In *4th International Workshop on Machine Learning and Music*.

François Pachet. 2012. Hit song science. *Music data mining*, pages 305–326.

François Pachet and Pierre Roy. 2008. Hit song science is not yet a science. In *ISMIR*, pages 355–360.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Nils Reimers and Iryna Gurevych. 2019a. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *arXiv preprint*.

Nils Reimers and Iryna Gurevych. 2019b. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Sbert. 2024. Sbert pretrained models. Accessed: 2024-04-01.

Dr. Neha Sharma, Dr. Prashant Pareek, Mr. Pushpak Pathak, and Ms. Nidhi Sakariya. 2022. Predicting Music Popularity Using Machine Learning Algorithm and Music Metrics Available in Spotify. *JOURNAL OF DEVELOPMENT ECONOMICS AND MANAGEMENT RESEARCH STUDIES*, 09(11):10–19.

Abhishek Singhi and Daniel G Brown. 2014. Hit song detection using lyric features alone. *Proceedings of International Society for Music Information Retrieval*, 30.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and Permuted Pre-training for Language Understanding. In *Advances in Neural Information Processing Systems*, volume 33, pages 16857–16867. Curran Associates, Inc.

Spotify. 2023. Spotify for Developers. Accessed: 2023-01-01.

Dennis M Steininger and Simon Gatzemeier. 2019. Digitally forecasting new music product success via active crowdsourcing. *Technological Forecasting and Social Change*, 146:167–180.

Xindi Wang, Alexander Gates, and Albert Laszlo Barabasi. 2023. An overivew of the science of success. In *Handbook of Computational Social Science*. Edward Elgar Publishing Ltd.

Zehao Xing. 2023. Popularity Prediction of Music by Machine Learning Models. *Highlights in Science, Engineering and Technology*, 47:37–45.

Yap Kah Yee and Mafas Raheem. 2022. Predicting Music Popularity Using Spotify and YouTube Features. *Indian Journal Of Science And Technology*, 15(36):1786–1799.

Eva Zangerle, Michael Tschuggnall, Stefan Wurzinger, and Günther Specht. 2018. ALF-200k: Towards Extensive Multimodal Analyses of Music Tracks and Playlists. In *Advances in Information Retrieval*, pages 584–590. Springer International Publishing.

Eva Zangerle, Michael Vötter, Ramona Huber, and Yi-Hsuan Yang. 2019. Hit song prediction: Leveraging low-and high-level audio features. In *ISMIR*, pages 319–326.