Gender Bias in a Music Recommendation ADS

Jeffrey Gordon (jg5837) and Cedric Lam (tvl9586)
December 8, 2025

1 Background

1.1 Introduction

An automated decision system (ADS), by definition, makes consequential decisions about individuals (Stoyanovich et al. [2020]). As such, some might argue that content recommendation systems fall outside the purview of traditional algorithmic fairness research because they make decisions about content rather than people, which seem relatively inconsequential. However, this perspective ignores a few important subtleties. While content recommendation systems do make decisions about content, these decisions also impact people, the **content creators**. Therefore, skew in the recommendation of content with regard to the characteristics of the content creator can have profound implications for which creators can succeed in a given industry.

We are interested in exploring recommendation skew in music recommenders. Notably, success in the music industry (both personal and financial) is largely dependent on an artist's ability to garner large audiences on content distribution platforms (Hesmondhalgh et al. [2023]). Recommendation systems play an interesting role in this relationship–not only do these systems reflect the success of an artist by training on data that uses artist popularity as an input, but they also help shape an artist's success by recommending or not recommending their work. This feedback loop illuminates the fact that the decisions of music recommendation systems are quite consequential. Therefore, the main focus of our analysis will examine this fairness question: **Are music recommenders fair with regard to artists of different demographic groups?**

1.2 Purpose

In particular, we examine the top solution (Bai [2018]) to a music recommendation Kaggle competition hosted by the Taiwanese streaming service KKBox Howard et al. [2017]. As defined in the Kaggle details, the competition's stated goal is to help KKBox build a better recommendation system. It is important to recognize that this problem formulation is largely user-centric. As a for-profit company, this makes sense: KKBox is interested in building a service that users feel compelled to come back to. Therefore, it is likely that fairness concerns, and particularly artist-centered fairness concerns, are not taken into account in pursuit of this end.

1.3 Tradeoffs

A good recommendation system has to balance accuracy and fairness. To maximize engagement, the recommender should suggest songs that users will "enjoy," even going beyond

their known favorites to introduce novelty. However, a system that prioritizes accuracy alone might inadvertently replicate demographic imbalances in training data; not to mention a user's personal listening biases! To some extent, this behavior is acceptable: a good recommender should not force a user to listen to music that they are not interested in. However, if an ADS interprets a user's lack of interaction with artists from one demographic background as a lack of interest, rather than a lack of exposure, this may introduce fairness concerns. In turn, this will perpetuate a feedback loop that disproportionately affects underrepresented groups, creating emergent bias.

2 Input and Output

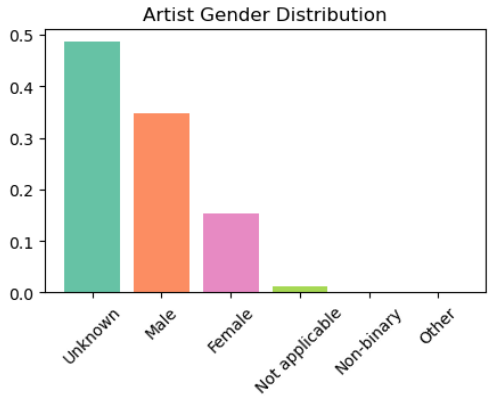
2.1 Data Collection

The main dataset for the Kaggle competition, `Train`, was compiled from real KKBox users' listening history over 12 years. The dataset does not include data on every listening event; instead, it only includes information on a user's first observable listening event for a given song. Described in more detail below, this is because the target variable is whether or not a user chooses to re-listen to a given song in the month after their first listen. The intuition for this structure is that the details about a user's first encounter with a song offer predictive signal about whether they will re-listen to it (i.e., if a user explicitly searched for a song, they may care a lot more about this song than if they had just encountered it randomly through a popular playlist). In addition, KKBox provided a few additional auxiliary datasets that supplement the listening history data. These datasets include `Songs` and `Song_extra_info`, as well as `Members` (extra info about users).

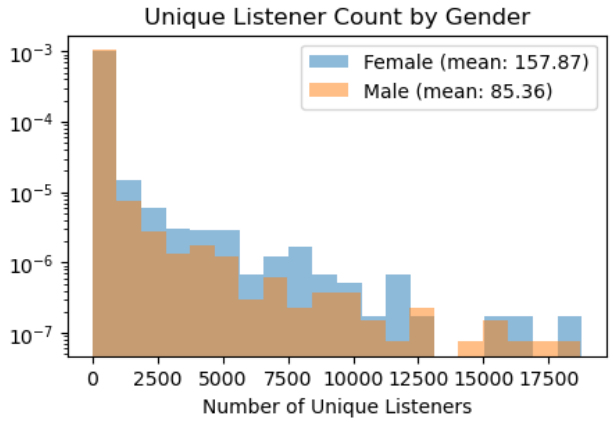
Notably, KKBox does not provide any demographic information for artists, which is necessary for our artist-centric fairness analysis. Instead, we sourced this information ourselves from a popular music database called MusicBrainz (MusicBrainz [2025]), mirroring the strategy employed in another music recommender study (Ferraro et al. [2021]). This strategy is imperfect–from a quick inspection of the results, we found that the information was generally correct, but not always. However, because this method has been used by other researchers in the field, we deem this data suitable enough our purposes. Note that our artist-centric audit is restricted by the demographic information available on MusicBrainz. For instance, MusicBrainz does not collect artist race or ethnicity, but does collect gender information, which we make the focus of our report.

2.2 Data Profiling

Appendix 3 details the table structure of the provided data files (`Train`, `Members`, `Songs`, and `Song_extra_info`) and our self compiled artist demographic dataset (`Artists`). The appendix also includes information on possible values and missing value percentage for each input feature. We spare the reader the details of every single feature, but there are a few observations worth noting. In general, KKBox's data is quite messy and not well-documented. For instance, the member city field takes on only one of 21 possible integer IDs, but it is impossible to know from inspection alone what these IDs correspond to. Our



(a) Artist gender distribution



(b) Unique listener count by gender, a proxy metric for popularity. Note the log y-axis.

Figure 1: Artist gender information

self-compiled artist demographic dataset reveals that most of the artists are Taiwanese, so it's likely that members in the dataset are also predominantly from the region. Accordingly, the 21 cities probably correspond to Taiwan's 23 cities, with the most popular city (ID 1) being Taipei. Another peculiarity of the dataset is that member age takes on some obviously incorrect values, ranging from -43 to 1051.

As our report examines artist-centric gender bias, we focus our profiling on artist gender information. Figure 1a details the distribution of artists by gender. The figure demonstrates that the most common gender among artists is `NULL`. However, this is not necessarily due to missing information. Instead, this value is left blank when the artist is a band or group of musicians, and gender cannot be easily defined. We therefore restrict our analysis to solo artists with well-defined genders. Further, because there are very few non-binary artists in the dataset, we will further limit our analysis to only male or female artists.

Figure 1a also shows a heavy skew towards male artists, with more than double the number of female artists. However, figure 1b shows that female artists tend to be much more popular, on average having nearly double the number of unique listeners as male artists. This paints an interesting picture of the gender skew in our data–**there are more male artists than female artists, but female artists tend to garner more attention.** Ultimately, in the dataset used to train the recommender (where an artist will appear multiple times), the gender imbalance washes out. As demonstrated by table 1, the proportion of all listening events by male vs. female artists is more equal than the proportion of unique male vs. female artists in the dataset. Even though there are fewer unique female artists in the dataset, these artists receive more plays on average than male artists.

2.3 Output

The prediction target for this task is a Boolean value indicating whether a user re-listened to a song within one month of the initial listen (1 for re-listened, 0 otherwise). Intuitively,

Artist Gender	Proportion of Plays	Average Plays
Male	0.3657	191.44
Female	0.3069	363.88

Table 1: Proportion of plays for all artists of a given gender and average plays for an artist of a given gender

re-listening is a strong proxy for user enjoyment, so this target is appropriate for the task of predicting music that a user will like. Note that this prediction is only one component of a complete recommendation system. Its output identifies a candidate pool of recommendable songs for a given user, from which a song will be selected as-needed during deployment. For our analysis, we assume that any song with a positive classification has an equal chance of being served to a user. Therefore, if the ADS' aggregate recommendations are fair, we can conclude that the system itself is fair.

The target variable distribution is well-balanced in the dataset, with 50.35% positive labels, and 49.65% negative. Due to the gender skew detailed in section 2.2, about 37.41% of positive targets are male artists, compared to 30.45% for female artists. Thus, a recommender that strictly mirrors the base rates of prevalence in the listening history will inherently favor male artists (as noted by Ferraro et al. [2021]). We do not consider this baseline distribution disparity to be a fairness concern on its own, as this is due to a combination of user preference and artist publishing behavior. However, those interested in increasing female representation in the music industry should take note of this disparity.

The output of the system is a value from 0 to 1, representing the probability that the user will re-listen to a song. We used a threshold of 0.5 to convert these probabilities into labels.

3 Implementation and Validation

The following sections are largely informed by the Kaggle winner's detailed solution writeup (Bai [2018]), as well as our own understanding of the source code ((lystdo [n.d.]) used for model training and prediction.

3.1 Data Cleaning and Preprocessing

The data cleaning and preprocessing is computationally expensive, involving millions of user interactions and a large amount of song and member metadata. This process consists of eight sequential stages that transform the raw data into a rich feature set.

The first feature engineering step encodes categorical data, including genres, artist names, and member attributes, with special handling to separate primary artists from collaborators. Dates are converted to Unix timestamps. Songs and members that do not appear in the train or test sets are filtered out to reduce dimensionality.

Then, features are aggregated into count-based statistics at multiple levels, from user-song interaction frequencies to artist and genre-level frequencies. The ADS also incorporates

temporal features to capture both short-term and long-term listening patterns, computing interaction counts across multiple time-based windows.

The ADS treats missing values in certain categorical features as missing-not-at-random (MNAR) by creating binary indicator flags that capture missingness as a predictive signal. For missing numerical features, a variety of imputation methods are employed, including mean imputation or assigning placeholder values. Some data outside reasonable bounds, such as the extreme age values mentioned in section 2.2, are considered to be missing.

The data is then converted into low-dimensional embeddings using Singular Value Decomposition. Features from member, song, and interaction tables are merged, with low-importance features removed based on importance scores. The final feature matrix combines encoded categoricals, normalized counts, low-dimensional embeddings, temporal patterns, contextual sequence information, and missingness indicators, for a total of 386 features!

3.2 High-Level System Implementation

The final system implementation uses a two-model ensemble architecture, combining a LightGBM model and a Deep Neural Network (PReLU and LeakyReLU activation). Each model outputs a probability that every user-song pair in the dataset would result in a repeat listen. The ADS then blends these probability with a weighted average (60% LightGBM and 40% Neural Network, determined via cross-validation). For our analysis, we trained both models using 80%-20% train-test splits across 10 random seeds.

3.3 ADS Validation

The ADS is validated on AUC score, achieving an out-of-sample test result of 0.85. This metric makes sense for this task as opposed to metrics that require thresholding, as a streaming service may wish to adjust the positive classification threshold depending on their tolerance for false positives (and the platform's catalog size). A streaming service wants users to stay on the platform, so we reason that the ADS has met its goal if it can identify music that a listener will re-listen to at a rate substantially higher than chance. A high AUC aligns with the streaming service's priorities, but not necessarily those of the artists who create the content it recommends. We will consider other metrics in the next section to determine whether this ADS is fair for artists with respect to their gender.

4 Outcomes

4.1 Metric Justification

To analyze the performance of this recommendation system, we will measure **accuracy**, which gives us a broad picture of the recommender's ability to detect signal in the data. However, in this scenario, accuracy is probably not the most important metric for any stakeholder. Instead, it is important to consider who is impacted by different prediction errors. False positives most strongly impact listeners, and by extension, the company deploying

the recommendation system. If the recommendation system suggests songs users aren't interested in, they might spend less time on the platform, or even stop using it altogether. Therefore, to maximize revenue, streaming services need to optimize for **Precision**, which addresses the following question: if a given song gets recommended, how likely is it that the user will actually enjoy it? We will also consider **FPR**, which measures the proportion of songs that got recommended but users will not enjoy.

By contrast, false negatives most strongly impact artists. If an artist spends time, money, and energy writing a song, they want to know that the streaming service will recommend that song to listeners who will enjoy it. **Recall** helps us quantify this, measuring the probability that a song gets recommended if we know that a user would enjoy it. Similarly, we will also consider **FNR**, which helps us measure the fraction of songs a user would actually enjoy that don't get recommended. From an artist's perspective, a high FNR is not ideal, suggesting that there are users who would enjoy their music, but the recommendation system does a poor job at finding these users. Note that FNR is far less important from the perspective of a streaming service–as long as users enjoy the songs that do get recommended, what doesn't get recommended is of little importance.

4.2 Performance

Table 2a lists the above metrics calculated at the row level.[1] The table suggests decent, but not perfect performance. That is, the classifier performs substantially better than just choosing the most popular label, but still makes quite a few errors, with roughly equal FNR and FPR. Note, however, that these metrics are calculated at the row level, meaning that they represent a blunt evaluation of the classifier's ability to recommend songs that a user would enjoy. Because popular songs and artists appear more frequently in the dataset, these metrics are dominated by what is popular. In a sense, these metrics are appropriate from the perspective of the listener and the streaming service because they correctly characterize the expected performance of any given recommendation. However, for an artist-centric analysis, these metrics are misleading, since they are dominated by the most successful artists.

To account for this skew, we developed a method of aggregating metrics across artists. This involves measuring performance for each artist separately, and then averaging metrics across all artists. By weighting each artist equally, this method more accurately captures classifier's performance for the typical artist. The results are listed in table 2b. While the accuracy and precision are about the same as the row-level metrics, we see a decrease in Recall and FPR, along with higher FNR. This implies that the classifier is more risk-averse for a typical artist than the row-level metrics suggest, only predicting positively when it is sure that a user will enjoy a song. Intuitively, this makes sense; the ADS is biased toward popular artists due to their broad appeal, and is more cautious about recommending less popular music. We also examined member-level metrics, where performance is then averaged across all users. This captures how individual listeners experience the performance

[1]These metrics were calculated on a single run of the classifier. We performed ten different train/test splits, but found that metrics were very stable across runs (likely due to large sample size). So, these and other reported results were calculated on one run only for simplicity and keeping runtimes reasonable.

Accuracy	0.76522
Precision	0.77512
Recall	0.75459
FNR	0.24541
FPR	0.22391

(a) Row-level

Accuracy	0.77033
Precision	0.74116
Recall	0.56596
FNR	0.40297
FPR	0.14089

(b) Artist-level

Table 2: Overall performance, calculated at the row level vs. aggregated across artists.

Gender	**Accuracy**	**Precision**	**Recall**	**FNR**	**FPR**
Male	0.7743	0.7423	0.5659	0.4001	0.1382
Female	0.7645	0.7386	0.5718	0.4033	0.1420

Table 3: Performance metrics aggregated across artists of the same gender.

of the recommendation system. These metrics closely mirror artist-level patterns, confirming that the aggregation method does not significantly impact the performance metrics.

Table 3 shows the same artist-level metrics, but split by artist gender. While there are very slight differences between male and female artists, these metrics do not suggest evidence of gender bias. **The classifier performs about the same on male and female artists**, with metrics for both being roughly equal to the aggregate artist-level metrics.

4.3 Fairness

The results in the previous section indicate fairly similar performance across artist genders. For completeness, we report fairness metrics in table 4, which make explicit the lack of gender bias.[2] The low FNR difference is especially important, suggesting that the recommendation system is equally good at serving music by artists of different genders to users that would enjoy it. Similarly, the low FPR difference suggests that the recommender does not systematically favor content by artists of one gender over another. Finally, the low selection rate difference suggests that male and female artists are recommended at similar rates. Overall, we conclude that **there is little evidence of gender bias in the classifier's performance.**

4.4 Additional Analysis

As alluded to above, artist popularity can drastically affect the recommendation system's performance, which is usually more generous in recommending popular music that most users will enjoy. Our profiling analysis revealed substantial gender skew with respect to popularity, as female artists in the dataset are more popular than male artists on average. Therefore,

[2]Note that these metrics had to be computed by hand from the artist-level aggregate performance metrics (and other metrics not reported here).

FNR Difference	0.00321
FPR Difference	0.00382
Demographic Parity Ratio	0.97406
Equalized Odds Ratio	0.96849
Selection Rate Difference	0.00818

Table 4: Fairness metrics for artist gender.

we also investigate gender bias while controlling for popularity to confirm that this system does not systematically disadvantage low (or high) popularity artists of one gender.

Figure 2 shows performance and fairness metrics by artist gender across different tiers of artist popularity. We use number of unique listeners (shown in figure 1b) as a proxy metric for popularity, bucketing artists into low popularity (less than 100 listeners), medium popularity (between 100 and 1000 listeners), and high popularity (greater than 1000 listeners). As expected, we observe a clear trend between popularity and model confidence, where the recommendation system becomes more generous as popularity increases, resulting in higher recall and FPR, along with lower FNR.

More importantly, when controlling for popularity, the fairness metrics indicate similar treatment of artists across genders by the recommendation system. While there are some slight differences in fairness at different popularity levels, the magnitude remains small enough to conclude that **there is minimal gender bias in the predictions, even when controlling for popularity level**. These results were extremely stable, with performance and fairness metrics varying with a standard deviation ≈ 0.0005 across 10 seeded runs, further validating the reliability of our findings.

5 Summary

While the ADS exhibits a clear bias towards popular artists, our analysis finds no evidence of systematic gender bias in predictive performance. Aggregate fairness metrics show negligible differences between male and female artists, and this parity remains consistent when controlling for artist popularity and member listening patterns.

5.1 Data Appropriateness

The data was largely suitable for the task, allowing for the development of a recommendation system that performs substantially better than guessing. Given the balance in re-listening rates between genders, the system achieves "fairness through blindness" without requiring explicit demographic inputs. However, if KKBox wants to increase the representation of female artists, they should collect artist demographic data themselves, and promote female artists at a higher rate than male artists. In-house data collection is also helpful for routine auditing, allowing for the detection of data drift and prevention of future bias.

Figure 2: Performance and fairness metrics by artist gender and popularity

5.2 Robustness/Stakeholders

The accuracy of the system is not perfect. However, as discussed above, optimizing for accuracy is generally not in the interest of any stakeholders in this scenario. Instead, a streaming service will usually want to optimize for high precision and low false positive rates–ensuring that users are likely to enjoy recommended songs. This goal can be achieved by raising the probability threshold for positive classification. With a large enough catalog, streaming services should be able to identify enough songs that meet a high threshold for every user. Artists publishing music on this streaming service should take confidence in the fact that the recommendation system exhibits comparable recall and false negative rates across genders, suggesting similar levels of support to artists regardless of gender.

5.3 Deployment

Based on the analysis results, **we would be comfortable deploying this ADS within the music streaming industry**. However, the model requires further calibration to minimize popularity bias, as it can be challenging to to find less popular (but not bad) music that listeners will enjoy. Deployment should also be paired with an in-house demographic data pipeline for fairness monitoring, without relying on third-party estimates. Furthermore, the ADS should incorporate granular feedback system (like thumbs up/down on recommendations) to improve recommendations over time, explainability features ("You might like X because you enjoyed Y"), and automated monitoring of any fairness drift.

5.4 Improvements & Future Considerations

For simplicity, we used a positive classification probability threshold of 0.5 for our analysis. However, real-world deployment might require stricter thresholds to minimize false positives. As such, we recommend that a fairness audit be performed at the deployer's chosen threshold, and future analysis could examine whether our results hold across the spectrum of possible thresholds.

Also, our artist-centric fairness analysis was limited to gender due to the limited demographic information collected by MusicBrainz. It would be prudent to examine artist-centric fairness with respect to other demographic characteristics, such as race, though we are unaware of any data sources that collect this information currently.

References

Bai, B. (2018, January 4). *A brief introduction to the 1st place solution (codes released).* Kaggle. https://www.kaggle.com/competitions/kkbox-music-recommendation-challenge/writeups/bing-bai-a-brief-introduction-to-the-1st-place-sol

Hesmondhalgh, D., Campos Valverde, R., Bondy Valdovinos Kaye, D., & Li, Z. (2023, February 9). *The impact of algorithmically driven recommendation systems on music consumption and production: A literature review.* Centre for Data Ethics and Innovation & Department for Digital, Culture, Media & Sport. Government of the United Kingdom. https://www.gov.uk/government/publications/research-into-the-impact-of-streaming-services-algorithms-on-music-consumption/the-impact-of-algorithmically-driven-recommendation-systems-on-music-consumption-and-production-a-literature-review

Ferraro, A., Serra, X., & Bauer, C. (2021, March 14–19). *Break the Loop: Gender imbalance in music recommenders.* In Proceedings of the 2021 ACM SIGIR Conference on Human Information Interaction and Retrieval (pp. 249–254). ACM. https://doi.org/10.1145/3406522.3446033

Howard, A., Chiu, A., McDonald, M., msla, Kan, W., & Yianchen. (2017). *WSDM – KKBox's music recommendation challenge [Competition].* Kaggle. https://www.kaggle.com/competitions/kkbox-music-recommendation-challenge

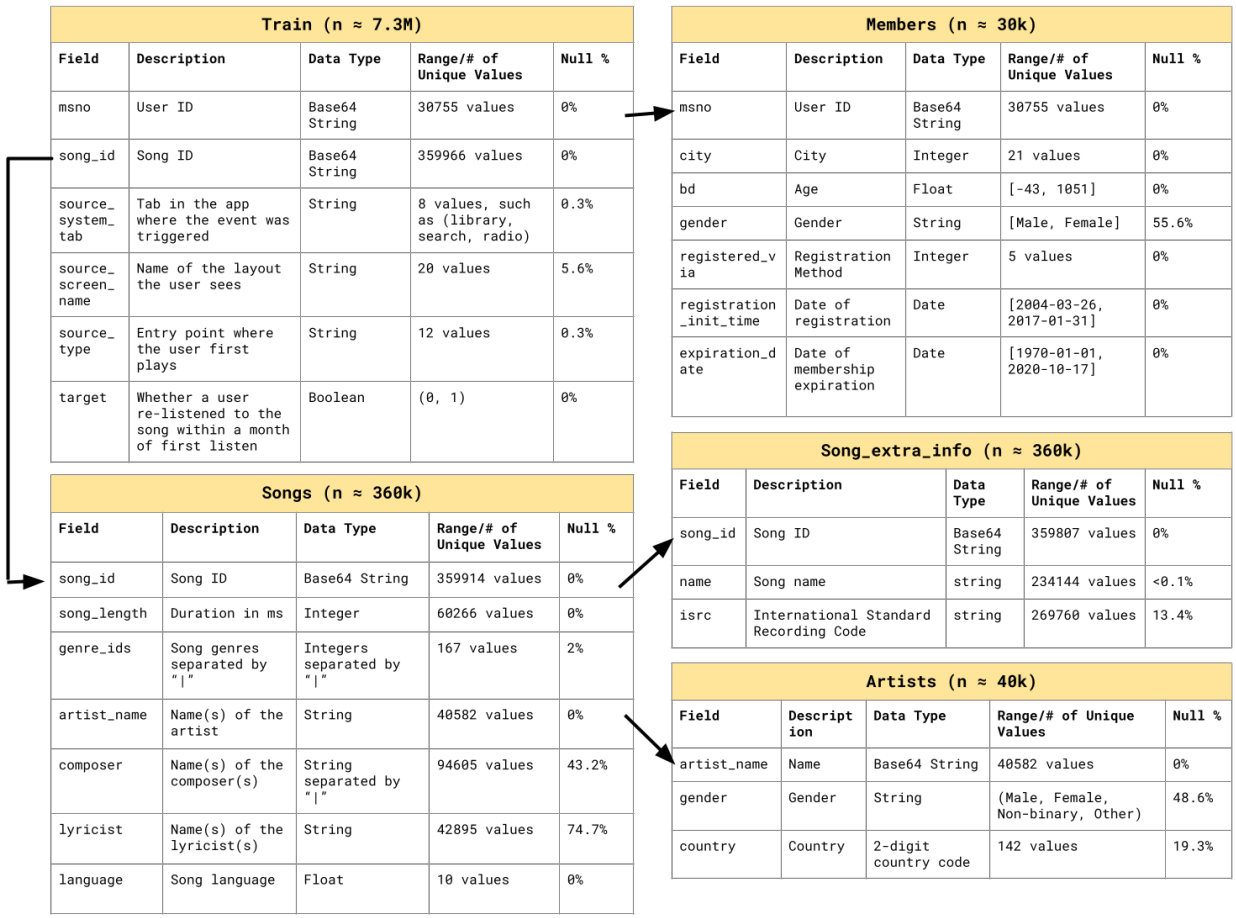
lystdo. (n.d.). *Codes for WSDM-CUP Music Rec 1st place solution [Source code].* GitHub. https://github.com/lystdo/Codes-for-WSDM-CUP-Music-Rec-1st-place-solution

MusicBrainz. (2025). *MusicBrainz: The open music encyclopedia.* Retrieved October 13, 2025, from https://musicbrainz.org/

Stoyanovich, J., Howe, B., & Jagadish, H. V. (2020). *Responsible data management.* Proceedings of the VLDB Endowment, 13, 3474–3488. http://dl.acm.org/citation.cfm?id=3424532

Appendix

Code used in our analysis can be found at `https://github.com/jeffgord/RAI-final`. This includes data collection, profiling, and analysis, as well as modifications made to the original model for the purposes of this project.



Train (n ≈ 7.3M)

Field	Description	Data Type	Range/# of Unique Values	Null %
msno	User ID	Base64 String	30755 values	0%
song_id	Song ID	Base64 String	359966 values	0%
source_system_tab	Tab in the app where the event was triggered	String	8 values, such as (library, search, radio)	0.3%
source_screen_name	Name of the layout the user sees	String	20 values	5.6%
source_type	Entry point where the user first plays	String	12 values	0.3%
target	Whether a user re-listened to the song within a month of first listen	Boolean	(0, 1)	0%

Members (n ≈ 30k)

Field	Description	Data Type	Range/# of Unique Values	Null %
msno	User ID	Base64 String	30755 values	0%
city	City	Integer	21 values	0%
bd	Age	Float	[-43, 1051]	0%
gender	Gender	String	[Male, Female]	55.6%
registered_via	Registration Method	Integer	5 values	0%
registration_init_time	Date of registration	Date	[2004-03-26, 2017-01-31]	0%
expiration_date	Date of membership expiration	Date	[1970-01-01, 2020-10-17]	0%

Songs (n ≈ 360k)

Field	Description	Data Type	Range/# of Unique Values	Null %
song_id	Song ID	Base64 String	359914 values	0%
song_length	Duration in ms	Integer	60266 values	0%
genre_ids	Song genres separated by "|"	Integers separated by "|"	167 values	2%
artist_name	Name(s) of the artist	String	40582 values	0%
composer	Name(s) of the composer(s)	String separated by "|"	94605 values	43.2%
lyricist	Name(s) of the lyricist(s)	String	42895 values	74.7%
language	Song language	Float	10 values	0%

Song_extra_info (n ≈ 360k)

Field	Description	Data Type	Range/# of Unique Values	Null %
song_id	Song ID	Base64 String	359807 values	0%
name	Song name	string	234144 values	<0.1%
isrc	International Standard Recording Code	string	269760 values	13.4%

Artists (n ≈ 40k)

Field	Description	Data Type	Range/# of Unique Values	Null %
artist_name	Name	Base64 String	40582 values	0%
gender	Gender	String	(Male, Female, Non-binary, Other)	48.6%
country	Country	2-digit country code	142 values	19.3%

Figure 3: table structure and field information (label, description, data type, range/number of unique values, and missing value percentage) for the provided and self-compiled data. Arrows show "foreign key" relationships. Note that the auxiliary datasets have been filtered to include only data relevant to the primary dataset (`Train`).