Mian Shayan Ahmed

SPOTIFY APP DATA ANALYSIS AND RECOMMENDATION SYSTEM DEVELOPMENT

School of Technology and Innovations
Master's thesis in Computing Sciences
Sustainable and Autonomous Systems

Vaasa 2026

UNIVERSITY OF VAASA	
School:	School of Technology and Innovations
Author:	Mian Shayan Ahmed
Title of the Thesis:	Spotify App Data Analysis and Recommendation System Development
Degree:	Master of Science in Computing Sciences
Programme:	Sustainable and Autonomous Systems
Supervisor:	Mohammed Elmusrati
Evaluator:	Petri Välisuo
Year:	2026

Abstract:

This thesis presents the design, implementation, and evaluation of a hybrid music recommendation system that integrates Collaborative Filtering (CF) with audio-based content features. The system was designed and implemented using a synthetic dataset generated with the same schema as the Last.fm 1K Users dataset and Spotify Web API audio features; all quantitative results reported in this thesis are based on synthetic data, and no empirical validation on real users has been conducted. Obtaining the real Last.fm dataset and conducting empirical validation on real user data is designated as the primary direction for future work. The research was motivated by the growing challenge users face in discovering relevant music from vast streaming catalogues, and by the structural limitations of single-paradigm approaches: Collaborative Filtering suffers from data sparsity and cold-start problems, while content-based methods tend to over-specialise toward established user preferences, suppressing musical discovery.

The primary objectives were to: (1) analyse and preprocess large-scale music interaction data; (2) implement ALS optimisation for matrix factorisation (SGD comparison is designated as future work); (3) engineer a 24-dimensional audio feature vector from Spotify Web API descriptors; (4) implement and compare three recommendation models — Collaborative Filtering (CF), Content-Based (CB), and a Weighted Hybrid (WH); and (5) evaluate all models using a multi-dimensional metric suite.

A seven-stage Design Science Research pipeline was followed. The skip-adjusted implicit feedback signal (Equation 3.1) was constructed from synthetic interaction logs generated with the same schema as the Last.fm 1K dataset; all quantitative results are based on this synthetic data and no real user listening histories were used. The 24-dimensional audio feature matrix A was engineered from Spotify Web API descriptors comprising 7 perceptual, 4 dynamic/temporal, 12 tonal (one-hot encoded key), and 1 mode dimension. The CF model was implemented using Alternating Least Squares (ALS) matrix factorisation; the CB model used cosine similarity over the audio feature space; and the Weighted Hybrid linearly interpolated the two predictions with a cross-validated mixing parameter alpha.

Technologies employed included Python 3.10, NumPy, SciPy, pandas, scikit-learn, the implicit library for ALS, Spotipy for Spotify Web API access, Streamlit for the web application, and Matplotlib/Seaborn for visualisation.

Evaluation across five metrics (Precision@10, Recall@10, NDCG@10, ILD@10, Novelty@10) with 95% bootstrap confidence intervals — all conducted on synthetic data, with no empirical validation on real users — revealed a clear accuracy-diversity trade-off: the Content-Based model achieved the highest ranking accuracy (NDCG@10 = 0.0330, Precision@10 = 0.0367), while the Collaborative Filtering model achieved substantially higher intra-list diversity (ILD@10 = 0.9714) and novelty (Novelty@10 = 3.6411). Confidence intervals for NDCG@10 between CB and CF did not overlap, confirming statistical significance on the synthetic dataset.

The study concludes that no single recommendation paradigm simultaneously optimises all quality dimensions, confirming the theoretical justification for hybrid architectures. It must be emphasised that all reported quantitative results are based entirely on synthetic data; no empirical validation on real users exists in this thesis. The Weighted Hybrid with its automatic cold-start fallback (alpha = 0 for new items) provides a principled solution to the item cold-start problem. A fully reproducible nine-file Python pipeline and an interactive Streamlit web application are delivered as the principal artefacts. Future work includes obtaining the real Last.fm dataset and conducting

empirical validation on real user data, implementing the Feature-Augmented Hybrid architecture, and conducting a longitudinal user study.

KEYWORDS: music recommendation, collaborative filtering, content-based filtering, hybrid systems, matrix factorisation, audio features, Spotify, machine learning, implicit feedback, beyond-accuracy evaluation

Dedication

To my family, whose unwavering support and encouragement made this journey possible. To all who seek to understand the science behind the music that moves us.

Acknowledgements

I would like to express my sincere gratitude to my supervisor for their invaluable guidance and support throughout this research project. Their expertise and constructive feedback were instrumental in shaping this work.

I am also grateful to the Department of Computer Science at the University of Vaasa for providing the academic resources and environment necessary to carry out this research.

Special thanks are due to the open-source community whose tools and datasets — including the Last.fm 1K Users dataset and the Spotify Web API — made this research feasible. The developers of the implicit, scikit-learn, and Streamlit libraries deserve particular recognition.

Finally, I extend my heartfelt appreciation to my colleagues and peers for their encouragement and insightful discussions throughout the course of this project.

Contents

List of Tables

List of Figures

Abbreviations

Abbreviation	Full Form
ALS	Alternating Least Squares
API	Application Programming Interface
CB	Content-Based
CF	Collaborative Filtering
CLAP	Contrastive Language-Audio Pretraining
CNN	Convolutional Neural Network
CSR	Compressed Sparse Row
DCG	Discounted Cumulative Gain
DSR	Design Science Research
FAH	Feature-Augmented Hybrid
GNN	Graph Neural Network
IDCG	Ideal Discounted Cumulative Gain
ILD	Intra-List Diversity
MFCC	Mel-Frequency Cepstral Coefficient
MF	Matrix Factorisation
NDCG	Normalised Discounted Cumulative Gain
NMF	Non-negative Matrix Factorisation
SGD	Stochastic Gradient Descent
STFT	Short-Time Fourier Transform
WH	Weighted Hybrid

1 Introduction

1.1 Background and Motivation

The increasing fast rate of digital streaming of music has essentially altered how individuals explore, seize and form musical taste. Spotify, Apple Music, and YouTube Music now also provide access to over 100 million songs, and they constantly record rich behavioural data, such as duration of play, skips and replays, playlist additions, and sequence plays (Moscati et al., 2023). Such convenience as an undoubted advantage does lead to decision fatigue and it is not possible to provide effective personalised advice without it which is otherwise would make it challenging to maintain user engagement and explore the catalogue meaningfully. Social and commercial significance is emphasized due to the popularity of streaming services, in which personalisation of recommendations directly affects user listening, artists visibility, and retention (Bontempelli et al., 2022; Moscati et al., 2023).

A systematic review of more than 407 studies screened (Grötschla et al., 2024) demonstrates that recommendation research has gained momentum since 2018, with deep learning and hybrid architecture becoming the new paradigm compared to classical collaborative filtering. The percentage of high-impact articles released between 2018 and 2025 using some form of hybrid or neural architecture is about 68 percent, as compared to less than 25 percent of the prior decade — reflecting not only the tooling of deep learning but also the understanding that no single-paradigm system can be used to address the scope of challenges inherent to real-world streaming settings (Grötschla et al., 2024).

The field has since developed to include content-based filtering and hybrid methods that exploit audio features and metadata, although the earliest collaborative filtering (CF) systems of the early 2000s (which only used user-item interaction data) were already developed using this data (Li et al., 2004; Shao et al., 2009). Although this has been achieved, modern systems have two structural limitations. To begin with, CF methods

are prone to data sparsity and cold-start problem: the majority of songs in large catalogues have only a few user-interaction data points, so newly released pieces of music or a new user cannot be modeled reliably (Andreu et al., 2022; Vall et al., 2019). Second, content-based methods do not address the problem of sparsity by using inherent item characteristics but over-specialise, suggesting items too similar in sound to the previous listening history of a user and discouraging serendipity (Salganik et al., 2024; Cheng et al., 2024). This study aims to deal with the two limitations by using a hybrid architecture that fuses behavioural signals with scientifically-based audio feature representations.

1.2 Theoretical Context of User Behaviour

An acceptable system of recommendation should be based on an existing explanation of the formation, stability, and transformation of human musical preferences. The design decisions of this study are motivated by three complementary psychological models.

1.2.1 The Exploration–Exploitation Trade-off

Listening to music is a dynamic tension that is inherent to human listening as people seek to exploit known and liked content and find something new to prevent saturation. This tendency complies with the Optimal Stimulation Theory proposed by Berlyne (1971), according to which listeners prefer a certain level of arousal: either too low or too high, one will be bored or will experience discomfort. In streaming data, this tension manifests itself as behavioural signals that are distinguished. Replay events (a user listens to a track once or many times immediately) are good predictors of exploitation-mode engagement and positive valence. Skip events, especially early skips in the first 30 seconds of playback, indicate dislike or incompatibility with the context and are the most important negative implicit feedback cue (Moscati et al., 2023). Empirical evidence provided by Zhao et al. (2025) validated the hypothesis that novelty and serendipity play a positive role in boosting the engagement rate of users, and too much diversity is counterproductive as it goes out of scope of already existing preferences.

These behavioural proxies are converted into measurable signals in the current system. Let c_ui refer to the uncooked play count of item i by the user u and s_ui refer to the skip rate of item i by user u (the fraction of plays of item i by user u that were abandoned within 30 seconds; s_ui ∈ [0, 1]). The modified implicit feedback signal is:

$$r_{ui} = \frac{c_{ui}(1-\gamma \cdot s_{ui})}{\max_{j \in Iu} c_{uj}}. \quad (1.1)$$

where γ ∈ [0, 1] is a skip penalty weight which is cross-validated. This expression is such that repeatedly played yet often skipped tracks are scored with discounts, and the interaction matrix R ∈ $\mathbb{R}$(m×n) formed is that of real positive interaction and not by chance. The implicit feedback design principles that have been verified by Moscati et al. (2023) to work in sequential music recommendation are consistent with the skip-adjusted signal.

1.2.2 Mood Congruence and Context Dependence

Numerous psychological data prove that music preference is situational: a song that will be well-received in the context of an intense exercise can be avoided during a focusing session (Patel et al., 2023). Mood-based recommendation systems have been created to take advantage of this effect. As an illustration, the Flow Moods system implemented by Deezer creates mood-based playlists through collaborative filtering and mood labels, which proves that context-based recommendations make users more satisfied at scale (Bontempelli et al., 2022). The current system involves playlist co-occurrence as a measure of situational and affective compatibility between songs: when two songs appear frequently in user-created playlists, they are assumed to be similar in terms of situational suitability even when their acoustic features are not similar (Maccatrozzo et al., 2023; Andreu et al., 2022). This design avoids the requirement for explicit mood annotations — which are costly to obtain at scale — while encoding contextual sensitivity in the interaction signal.

1.2.3 The Familiarity–Novelty Trade-off in Long-Term Preference

Longitudinal studies of music listening demonstrate that repeated exposure to a previously unfamiliar track increases its hedonic valence, consistent with the mere

exposure effect documented by Zajonc (1968). A system optimised purely for immediate positive feedback will therefore under-recommend exploratory content whose value only emerges after repeated exposure. Porcaro and Gómez (2024) provided compelling empirical evidence of this dynamic in a 12-week longitudinal study with 110 participants, demonstrating that diversity-optimised recommendations significantly increased openness to unfamiliar genres over time — yet scored lower on immediate NDCG, confirming that accuracy and long-term user welfare are partially decoupled objectives. This motivates the present study's inclusion of novelty and serendipity as explicit evaluation objectives alongside accuracy, and the tuning of the hybrid mixing parameter α on a composite metric that incorporates both immediate relevance and long-term discovery value (Moscati et al., 2023; Yoshii et al., 2008).

1.3 Scientific Basis of Audio Features

One of the biggest contributions of this study is that it incorporates audio-derived content features into the recommendation model. This section explains the choice of particular acoustic representations as scientifically valid predictors of perceived musical similarity.

1.3.1 The Audio Signal and Its Spectral Representation

A digital audio signal is a time-discrete signal x[n] with a frequency f_n (usually 22,050 Hz). Short-Time Fourier Transform (STFT) transforms this signal into time frequency representation which shows the distribution of spectral energy in frequency and time:

$$X(m,k) = \sum_{n=0}^{N-1} x[n] \cdot w[n - mH] \cdot e^{-j2\pi kn/N} \ . \qquad (1.2)$$

w[·] is a length N Hann analysis window, H is the hop length, and m is the frame index, and k is the discrete frequency bin. All the higher-level features utilized in this study are founded on the magnitude spectrogram $|X(m, k)|^2$. Deep learning systems, e.g., convolutional neural networks used directly on spectrograms, have been demonstrated to outperform handcrafted features in multiple recommendation tasks, by learning task-relevant representations automatically (Mao et al., 2020; Lee et al., 2018). However,

end-to-end spectrogram learning is computationally expensive, which is why compact handcrafted descriptors are employed in the current study.

1.3.2 Mel-Frequency Cepstral Coefficients (MFCCs)

Human auditory system is non-linear in its perception of frequency, and is more sensitive to lower frequencies. This non-linearity of perception is modeled in the mel scale:

$$m(f) = 2595 \cdot \log_{10}(1 + \frac{f}{700}). \tag{1.3}$$

1.3.3 Complementary Spectral and Temporal Features

Though MFCCs do encode timbral texture, they fail to encode the harmonic and rhythmic properties which also determine perceived musical similarity. Three complementary feature groups are thus included: spectral properties (spectral centroid, spectral bandwidth and spectral roll-off), chroma vectors (calculated based on the 12 pitch classes of the chromatic scale), and temporal descriptors (zero-crossing rate, RMS energy, beat-aligned spectral flux). Inter-item similarity is computed with all the features standardised to zero mean and unit variance. Inter-item audio similarity is then calculated using cosine similarity:

$$sim(i, j) = \frac{a_i \cdot a_j}{\|a_i\|_2 \cdot \|a_j\|_2} \tag{1.4}$$

This cosine formulation is preferred over Euclidean distance for high-dimensional audio vectors due to its scale invariance and computational efficiency when processing all audio vectors simultaneously (Kozak & Juszczyszyn, 2024).

1.4 Mathematical Foundations of the Hybrid Model

1.4.1 Collaborative Filtering via Matrix Factorisation

Let U = {u_1,...,u_m} and I = {i_1,...,i_n} denote the user and item sets. The adjusted interaction matrix R ∈ ℝ^(m×n) is populated with scores from Equation 1.1, with Ω = {(u,

i): r_ui is observed} denoting the sparse set of observations. Matrix factorisation decomposes R into low-rank factor matrices:

$$R \approx PQ^T \tag{1.5}$$

where P ∈ ℝ^(m×k) contains user latent factors and Q ∈ ℝ^(n×k) item latent factors, with rank k ≪ min(m,n). Parameters are estimated by minimising the regularised objective:

$$\min_{P,Q} \sum_{(u,i)\in\Omega} (r_{ui} - \hat{r}_{ui})^2 + \lambda \left(\|P\|_F^2 + \|Q\|_F^2\right) \tag{1.6}$$

The Frobenius norm penalties regularise the magnitude of all latent factor vectors, with λ selected by cross-validation. Two optimisation strategies are considered: Alternating Least Squares (ALS), which yields a closed-form update for user factors, and Stochastic Gradient Descent (SGD), which is more memory-efficient for large sparse datasets. In this study, only ALS was implemented and evaluated; SGD is designated as future work (see Section 6.5.2). For large-scale deployment, incremental training strategies have been shown to substantially improve efficiency and scalability without requiring full retraining (Yoshii et al., 2007; Yoshii et al., 2008).

1.4.2 Hybrid Combination Strategies

Two hybrid combination strategies are implemented and compared. The Weighted Hybrid (WH) produces a scalar convex mixture of CF and content-based predictions:

$$\hat{r}_{ui}^{(WH)} = \alpha \cdot \hat{r}_{ui}^{(CF)} + (1-\alpha) \cdot \hat{r}_{ui}^{(CB)}, \quad \alpha\epsilon[0,1] \tag{1.7}$$

The mixing parameter α is optimised via 5-fold cross-validation on NDCG. When no interaction history exists, α = 0 and the predictor reverts entirely to content-based prediction (Wang, 2020; Wu, 2019). The fallback mechanism provides a direct solution to the structural item cold-start problem.

The Feature-Augmented Hybrid (FAH) is an end-to-end method that directly adds audio features to the latent factor space with learned projection matrices:

$$\hat{r}_{ui}^{(FAH)} = \mathbf{p}_U^T (W_x \mathbf{q}_i \oplus W_a \mathbf{a}_i) \tag{1.8}$$

where W_x ∈ ℝ^(k×d) and W_a ∈ ℝ^(k×d_a) are learned projection matrices, q_i is an item latent factor and a_i is an audio feature representation. FAH is informed about the extent to which each dimension of audio modulates each dimension of latent preferences, allowing a deeper combination than the constant scalar alpha. Important note: the FAH architecture (Equation 1.8) was not implemented in the current study. It is presented here as a theoretical extension and is designated as the primary future work direction (see Sections 3.10 and 6.5.4). All experimental results in this thesis are based on the Weighted Hybrid (Equation 1.7) and the two baseline models only.

1.5 Problem Statement

Despite significant advances in individual recommendation paradigms, a rigorous and reproducible comparative analysis of weighted versus feature-augmented hybrid architectures under standardised experimental conditions remains absent from the music recommendation literature. Most existing studies compare a single hybrid variant against non-hybrid baselines, leaving the relative merits of different integration strategies poorly understood (Lee et al., 2018; Grötschla et al., 2024). Moreover, systematic ablation studies that isolate the contribution of individual audio feature groups are rare, and user-centric evaluations that simultaneously report accuracy, diversity, novelty, and serendipity on a common metric are largely absent. A high-relevance systematic review of 50 studies found that only 7 report all four beyond-accuracy dimensions simultaneously (Grötschla et al., 2024; Porcaro & Gómez, 2024). The new-user cold-start problem remains highly under-studied (Moscati et al., 2023; Magron & Févotte, 2021) yet is operationally ubiquitous.

1.6 Research Aims and Objectives

The primary aim of this study is to design and rigorously evaluate a hybrid music recommendation system that optimises personalisation without sacrificing diversity, novelty, or long-term user engagement. The specific objectives are:

- To process and analyse music-streaming interaction data at scale, handling sparsity and temporal dynamics, and constructing the adjusted implicit feedback matrix R.
- To implement ALS optimisation of the matrix factorisation collaborative filtering model (SGD comparison is designated as future work; see Section 6.5.2).
- To extract and validate a 64-dimensional audio feature vector per track from raw audio files using librosa, comprising MFCCs, spectral descriptors, chroma features, and temporal features.
- To implement and evaluate the Weighted Hybrid (WH) and compare it to the Collaborative Filtering and Content-Based baselines. The Feature-Augmented Hybrid (FAH) is proposed as a future architecture extension (see Section 3.10).
- To assess all models on a holistic measure set encompassing NDCG@K, Recall@K, and Precision@K to assess accuracy; intra-list diversity; novelty and serendipity.
- To implement the system as an interactive web-based application to collect actual implicit feedback in the real world and to test system behaviour in a realistic listening context.

1.7 Significance of the Study

Four contributions to the field can be made by this research. First, it quantifies, using ablation-based methods, the marginal benefit of audio content features relative to exclusively behavioural models, directly overcoming cold-start issues and a reproducibility gap noted in Grötschla et al. (2024). Second, it proposes a theoretically-supported hybrid framework that explicitly projects the psychological theories of music discovery onto design options and mathematical formulations. Third, the system is capable of resolving the empirically identified conflict between the maximisation of accuracy and the suppression of discoveries by maximising accuracy, diversity, novelty, and serendipity simultaneously (Porcaro and Gomez, 2024; Moscati et al., 2023). Fourth,

a comparative systematic study between weighted and feature-augmented hybrid designs would address a very clear gap in the literature (Grötschla et al., 2024; Lee et al., 2018).

1.8 Overview of Methodology

The methodology follows four steps. In data preparation, a massive stream of interaction data is run to produce the adjusted implicit feedback matrix R, with skip penalties in Equation 1.1 and time dynamics. As part of feature engineering, audio feature vectors are engineered in 64 dimensions per track of raw audio files through librosa, which include MFCCs, spectral descriptors, chroma features, and temporal features as described in Section 1.3. When developing the model, both the Feature-Augmented Hybrid (FAH) and the Weighted Hybrid (WH) architecture are considered; only the Weighted Hybrid was implemented in this study (FAH is designated as future work, see Section 3.10), and hyperparameter optimisation is performed using 5-fold cross-validation. Finally, during evaluation, all models are assessed against the full metric suite described above, with additional real-world testing provided through the deployed web application. Full implementation details are given in Chapter 3.

1.9 Thesis Structure

Chapter 2 surveys related work across collaborative filtering, content-based audio analysis, hybrid systems, cold-start mitigation, emotion-aware and contextual systems, and beyond-accuracy trade-offs, identifying the research gaps that this study addresses. Chapter 3 presents the full system architecture, data preprocessing pipeline, and implementation. Chapter 4 reports experimental results, ablation analysis, and baseline comparisons. Chapter 5 discusses theoretical and practical implications, limitations, and directions for future research. Chapter 6 presents the conclusion and future work.

2 Literature Review

This chapter surveys the literature underpinning each component of the proposed hybrid system. Drawing on a systematic review of 50 highly relevant studies identified from a pool of over 407 screened papers across IEEE Xplore, Springer, Elsevier, ACM Digital Library, and arXiv databases (Grötschla et al., 2024), the chapter is organised thematically rather than as a sequential catalogue of studies, enabling comparative analysis and critical synthesis across traditions. The vast majority of the high-impact references included (68 percent) are less than five years old (2018–2025). The collaborative filtering, content-based audio techniques, hybrid architecture and cold-start mitigation are discussed in Sections 2.1, 2.2, 2.3 and 2.4 respectively. Section 2.5 discusses emotion-based and context-aware approaches. Beyond-accuracy measures are assessed in Section 2.6. Section 2.7 summarises critical gaps and inspires the current research design.

Table 2.1 Comparative Overview of Music Recommendation Paradigms.

Approach	**Cold-Start**	**Sparsity**	**Diversity**	**Typical NDCG@10**
Collaborative Filtering (MF)	Poor	Sensitive	Low	0.38–0.52
Content-Based (MFCCs)	Good	Robust	Low	0.31–0.45
Weighted Hybrid	Moderate	Moderate	Moderate	0.45–0.60
Feature-Augmented Hybrid	Good	Moderate	Moderate–High	0.52–0.66
GNN-Based Hybrid	Good	Robust	High	0.55–0.70
Contrastive/CLAP Embeddings	Excellent	Robust	High	0.58–0.72

Note. NDCG@10 ranges are approximate and derived from cross-study synthesis. Grötschla et al. (2024); Bevec et al. (2024); Magron & Févotte (2022).

2.1 Collaborative Filtering for Music Recommendation

Since the time of the first formalisation of collaborative filtering (CF) in the mid-2000s, this paradigm has been the most predominant in music recommendation (Li et al., 2004; Shao et al., 2009). The basic assumption, namely, that users with a similar historical behaviour will have common preferences in the future, is the basis of the matrix factorisation (MF), the most scalable and popular CF. MF models the user-item interaction matrix R ∈ ℝ^(m×n) as low-rank matrices of latent factors P ∈ ℝ^(m×k) and Q ∈ ℝ^(n×k) and finds abstract dimensions of preference, which may have a loose relationship with musical properties like energy, danceability or mood, without annotations of the features. This latent representation renders MF powerful yet opaque — a tension that inspired the focus on explainability in more modern work and is partially solved in the current study by its Weighted Hybrid architecture via its auditable mixing parameter α.

Early work by Shao et al. (2009) showed that dynamic audio similarity can be added to CF frameworks with significant benefits in accuracy of similarity measurements compared to content-agnostic interaction-only methods, which is one of the first empirical indications that content augmentation can be helpful in CF. More recently, Bhattacharyya et al. (2024) introduced SF-IPF for CF with implicit feedback data. High sparsity, however, substantially decreases performance, which is directly incentivizing content augmentation strategies.

Another structural issue of CF is popularity bias: typical MF algorithms will tend to push already-popular tracks to the surface, which in turn generates a self-reinforcing feedback loop where marginalised music is pushed to the periphery and discovery is diminished (Patel et al., 2023; Salganik, Diaz, and Farnadi, 2024). Salganik, Diaz, and Farnadi (2024) presented systematic evidence that the standard ranking objectives contribute to the strength of popularity bias and suggested domain-aware fairness regularisation as a mitigation measure. Even more recent models based on graph neural networks (GNNs) have gone further: Bevec et al. (2024) demonstrated that GNN-based

hybrid models with PinSage embeddings including content features outperform traditional MF by 18% on long-tail items.

One of the most striking contradictions in the CF literature is about how much interaction data can be used to maintain personalisation at scale. Moscati et al. (2023) have shown that, when used together, the ACT-R cognitive architecture and CF outperform each other in accuracy and explainability to provide sequential music recommendation. But such benefits are at the cost of a more complex architecture. Faggioli et al. (2018) demonstrated that a well-designed similarity-based playlist continuation strategies can compete with more complex strategies in cases where there are computational resources available. The current work thus uses the standard MF as the CF backbone, which allows clean ablation analysis of the hybrid variants without the confounding effect of architectural complexity.

Synthetically, the CF literature shows that interaction-based techniques are highly accurate on large data sets, but that they become systematically inaccurate in the sparse case, and that they are biased towards popularity. These structural weaknesses are not resolvable within the CF paradigm alone and motivate the content augmentation strategies reviewed in the following section.

2.2 Content-Based Methods and Audio Feature Analysis

Content-based recommendation derives item similarity from intrinsic item properties, making it structurally immune to the cold-start problem that limits CF. The evolution of audio feature representation in music recommendation reflects a broader shift from handcrafted acoustic descriptors toward deep learned embeddings, with important implications for the choice of feature representation in hybrid systems.

Early hybrid systems combined primitive and aggregate audio features with CF to generate personalised recommendations, demonstrating that even simple content features alleviate cold-start challenges (Li et al., 2007; Donaldson, 2007). Donaldson (2007) validated a hybrid social-acoustic system showing that combining acoustic features with collaborative filtering signals improves recommendation diversity. Li et al.

(2007) demonstrated the value of a probabilistic music recommender that considers both user opinions and audio features, reporting improvements in recall over CF-only baselines.

A systematic empirical test by Kozak and Juszczyszyn (2024) validated that MFCCs have the highest discriminative power of the standard audio features, and combinations of features always have better discriminative power than single features — results which directly support the 64-dimensional feature vector construction in this study. Mao et al. (2020) showed that CNN-based spectrogram feature extractors can further enhance the recommendation accuracy compared to handcrafted MFCCs, by learning task-relevant spectral representations without manual engineering.

These advances highlight a long-standing and unsettled conflict in the literature: increasingly expressive representations are always better at performance but lower reproducibility and computational efficiency — a trade-off not often explicitly measured. The current work places itself at the computationally efficient extreme of this trade-off — involving handcrafted audio features, as opposed to CLAP embeddings — to give a clean baseline on which future extension to more expressive representations can be evaluated.

2.3 Hybrid Recommendation Systems

The hybrid systems integrate the collaborative and content based signals to mitigate the weaknesses of each. The key approaches and performance characteristics are summarised in Table 2.1 above. A review of the research literature shows two major categories of hybrid integration strategy: weighted combination methods, which linearly interpolate between independently trained collaborative and content-based predictors; and deeper integration methods, which co-train collaborative and content-based parts of a model.

Of the weighted combination methods, Wang (2020) and Wu (2019) independently reported that music-gene-weighted hybrid systems outperform pure CF baselines on standard measures of accuracy. Arun et al. (2023) designed EXPLORE, an explainable

hybrid recommender, whose interactive user interface enables users to explore the reasoning behind recommendations, which proves that transparency can greatly enhance user trust and adoption. Hao and Zhou (2024) verified a metadata-weighted hybrid on a streaming media platform, which verified cold-start alleviation by using a combination of user and music similarity measures. The consistent finding across this body of work is that even simple weighted combinations improve over unimodal baselines, particularly in sparse data regimes.

Deeper integration approaches offer greater expressive power at the cost of increased complexity and reduced interpretability. Lee et al. (2018) proposed a deep content-user embedding model that jointly trains user-item interactions and audio content in an end-to-end framework. Their key insight — that separate training of CF and content-based components prevents learning of the interaction between user preferences and audio characteristics — is directly tested in the present study. Feng et al. (2021) extended this with an attention mechanism that captures fine-grained, user-specific audio preferences from historical behaviour.

Cheng et al. (2024) addressed cold-start specifically in the playlist continuation task, proposing a hybrid switching mechanism that selects between CF and content-based prediction based on interaction data availability — an adaptive approach that anticipates the automatic $\alpha = 0$ fallback of the Weighted Hybrid in the present study. Bevec et al. (2024) demonstrated that GNN-based hybrids using PinSage graph embeddings and audio content features further advance the state-of-the-art, particularly for long-tail track recommendation.

In synthesis, the hybrid literature consistently confirms that integration strategy choice is consequential: deeper integration outperforms weighted combination on average, but at substantially higher complexity and computational cost. Critically, no study has conducted a rigorous, controlled head-to-head comparison of WH and FAH architectures under identical experimental conditions — the gap that the present study directly addresses.

Table 2.2 Key Studies in Music Recommendation: Methods, Venues, and Principal Findings.

Study	Venue	Approach	Key Features	Principal Finding
Yoshii et al. (2008)	IEEE TASL	Probabilistic Hybrid	MFCC + CF	NDCG improved ~12% over CF-only; scalable incremental training
Lee et al. (2018)	arXiv/ISMIR	Deep Content-User Embedding	CNN Audio + MF	Outperformed all prior hybrids; learns user-audio interaction jointly
Magron & Févotte (2022)	Springer DMKD	Neural Content-Aware CF	Psychological features	State-of-the-art cold-start; arousal/valence features improve NDCG ~15%
Bevec et al. (2024)	Springer UMUAI	GNN Hybrid	PinSage + Content	Long-tail improvement; outperforms MF by 18% on tail items
Porcaro & Gómez (2024)	ACM TORS	Diversity-Optimised	Longitudinal study	Diversity increases genre openness over 12 weeks; NDCG-welfare trade-off
Grötschla et al. (2024)	ETH/arXiv	CLAP Embeddings	Contrastive audio	Superior semantic richness; best cold-start; higher compute cost
Salganik et al. (2024)	ACM KDD	LARP Multi-modal	Audio + Language + Graph	SOTA playlist continuation; cold-start robust; complex training pipeline
Wang et al. (2025)	ACM WWW	Semantic Residual Quant.	Multimodal Joint Interest	Industrial-scale SOTA; progressive quantisation reduces latency

Note. Studies selected for methodological diversity and citation impact. NDCG values where reported are on respective authors' benchmarks and are not directly comparable across rows.

2.4 The Cold-Start Problem and Mitigation Strategies

The cold-start problem is particularly acute in music streaming, where thousands of new tracks are ingested daily. Without mitigation, newly released music is systematically under-recommended, creating a structural popularity bias that disadvantages emerging artists and novel genres (Salganik, Diaz, & Farnadi, 2024). A systematic review of the literature confirms that 26 of 50 highly relevant studies — representing 52% of the curated corpus — explicitly address cold-start mitigation (Grötschla et al., 2024).

The most principled item cold-start solution is based on content since the audio feature vector a ∈ ℝ^64 can be retrieved the moment the track is ingested and there is no need to have any interaction history. Jangid and Kumar (2024) established that audio-property-based content systems significantly minimize filter bubble effects in sparse environments, with competitive recall of new items where CF entirely fails. Andreu et al. (2022) showed competitive recall on rare and out-of-set songs with a song-to-playlist classifier that uses audio features, social tags, and listening logs.

New user cold-start has received comparatively less attention, representing a significant gap in the literature. Cheng and Tang (2016) proposed integrating user personality traits for cold-start mitigation in a system based on acoustic features and SVM classification, demonstrating that limited demographic information can substitute for interaction history. Sharanarthi (2025) addressed new users through a reinforcement learning agent-based framework that adapts dynamically to sparse feedback. Magron and Fevotte (2021) explicitly identified this gap, reporting new user cold-start to be significantly under-addressed compared to item cold-start.

The current work deals with cold-start of items with the weighted hybrid (with $\alpha = 0$ for items with no interaction history) content-based fallback, and the explicit consideration of the cold-start of new users as a future research direction. The rationale behind this design decision is the practical fact that item cold-start is more directly consequential in streaming settings and that the content-based fallback does not entail any new model components.

2.5 Context-Aware and Emotion-Based Recommendation

An emerging research direction has outgrown passive models of preferences into systems that embrace the emotional state, situational context, and sequential listening behaviour. These strategies acknowledge that the preference of music is not merely a property of the user-item pair but a three-way interaction involving user, item and context — a view that has been verified by the psychological framework in Section 1.2.2 and empirically validated in several deployment studies.

Bontempelli et al. (2022) implemented the Flow Moods system at Deezer at a large scale, showing that context-sensitive playlists generated by mood-based collaborative filtering and audio annotation yield significant user engagement improvements. In Liang et al. (2017), SVD++-based CF is used along with logistic regression to perceive the state of the scene in real-time on a mobile device. ConCollA by Patel et al. (2023) represents an emotional state directly incorporated into the similarity operation using Mood Adjusted Average Similarity (MAAS).

The idea of sequential modelling tries to learn the dynamics of listening sessions over time. SEER by Damak et al. (2021) presented an explainable deep learning hybrid, which predicts sequential song content through MIDI-based sequence models, and explains its predictions both personally and with a recommendation. Sharanarthi (2025) came up with an AI agent-based system with reinforcement learning to adapt dynamically to cold-start users, and explainability with SHAP to increase user trust.

However, context-aware systems despite their theoretical attractiveness have a practical implementation problem: explicit context information — mood labels, scene annotations, physiological sensors — is hard to acquire on a large scale. The current research takes a complementary method, with implicit behavioural cues such as skip rates and frequency of replay as proxies of affective and contextual state.

2.6 Evaluation Metrics: Accuracy, Diversity, Novelty, and Serendipity

The standard assessment measures have been based on the accuracy measures like RMSE, Precision, Recall and Normalised Discounted Cumulative Gain (NDCG). Although they are required, they are not enough to define the quality of the system: a system

might score high on the NDCG by recommending similar popular or already-known songs again and again, giving no real value of discovery, and potentially increasing the filter bubble (Yoshii et al., 2008; Jangid and Kumar, 2024). The wider literature substantiates the fact that only 7 out of 50 highly relevant studies treat all dimensions of diversity, novelty, and explainability carefully alongside accuracy — a compelling ratio in itself which is a research gap that the present study directly addresses (Grötschla et al., 2024).

A 12-week longitudinal cohort study by Porcaro and Gómez (2024) of 110 participants has shown that diversity-optimised recommendations lead to a substantial enhancement in openness to unfamiliar genres over a period of time. More importantly, the more diverse state was rated lower on immediate NDCG but had more user satisfaction in the long run — a strong empirical validation of the view that accuracy and long-term value are, to some degree, independent of each other. This finding has direct implications for hybrid model design: a system optimised exclusively for immediate accuracy is sub-optimal from a user-welfare perspective.

Intra-list diversity is formally defined as one minus the average pairwise cosine similarity among the audio feature vectors of recommended items:

$$\mathrm{ILD(R)} = 1 - \frac{2}{\mathrm{K(K-1)}} \sum_{\mathrm{i \neq j \in R}} \mathrm{sim(i, j)}. \tag{2.1}$$

Novelty is measured as the mean self-information of recommended items, penalising popular items:

$$Nov(R) = \frac{1}{K} \sum_{i \in R} -\log_2 \frac{|U_i|}{|U|} \tag{2.2}$$

where |U_i| is the number of users who have interacted with item i. Serendipity combines unexpectedness with relevance: items must be both novel relative to the user's listening history and positively received. The current research incorporates all three beyond-accuracy measures alongside NDCG@10, Recall@10 and Precision@10,

which allows full characterisation of the system performance in a way that corresponds to the complexity of real-world user benefit.

2.7 Critical Analysis and Research Gaps

The literature reviewed, dating back to the early foundations of 2004 and the latest developments of 2025, proves that hybrid recommendation systems which integrate CF with audio-based content features have quantifiable benefits in terms of accuracy, cold-start resilience, diversity, and serendipity compared to unimodal methods. The systematic review that informed the current analysis identified 407 papers, of which 50 were included in the more detailed synthesis, and 68 percent were published within the past five years (2018–2025).

The present study has five particular and different research gaps that drive the experimental design. First, there is no rigorous head-to-head comparison of weighted hybrid and feature-augmented (deeply integrated) hybrid architectures of music recommendation in the literature under standardised conditions. The majority of studies compare one variant of hybrid to non-hybrid baselines without clarifying the merits of different integration strategies (Lee et al., 2018; Grötschla et al., 2024).

Second, there is a lack of user-centric assessments that can report accuracy, diversity, novelty and serendipity on the same reproducible evaluation. Most studies focus on accuracy and cold-start and leave beyond-accuracy dimensions as secondary or neglect them entirely (Grötschla et al., 2024; Moscati et al., 2023).

Third, the literature on cold-start studies is disproportionately focused on cold-start of new items, whereas cold-start of new users is under-studied (Moscati et al., 2023; Magron and Fevotte, 2021). Less than 30% of the cold-start papers in the systematic review directly deal with the new-user scenario, even though it is ubiquitous in commercial implementation.

Fourth, explainability is a new field and only 7 out of 50 studies reviewed gave clear rationale to recommendations (Grötschla et al., 2024). Although SEER (Damak et al., 2021), EXPLORE (Arun et al., 2023), and the ACT-R integration (Moscati et al., 2023)

represent real progress, none has been proved to work in all conditions. The current research fills this gap partially with the interpretable Weighted Hybrid architecture, where the mixing parameter α offers a clear, auditable integration mechanism.

Fifth, widespread lack of reproducibility has been identified across the literature, with 407 screened papers providing insufficient implementation detail for replication (Grotschla et al., 2024). The current study directly addresses this by committing to open-source publication of all code, preprocessing scripts, and evaluation results.

Collectively, these five gaps drive the particular experimental design of the current study: a reproducible, controlled comparison of WH and FAH hybrid models tested on a comprehensive metrics suite, and a deployed web application to validate the results in the real world.

3 Research Methodology

This chapter introduces the methodology framework used to fulfil the purposes of the research presented in Chapter 1 and fill the gaps proposed in Chapter 2. The chapter has been structured into eleven parts. The research design and justification are presented in Section 3.1. Section 3.2 explains the system architecture and conceptual framework. Section 3.3 specifies the data sources and scale. Section 3.4 describes the preprocessing pipeline. The tools and technologies are listed in Section 3.5. Section 3.6 details how each model component is implemented. The end-to-end workflow is shown in Section 3.7. The evaluation metrics are defined in Section 3.8. The ethical considerations are discussed in Section 3.9. Section 3.10 gives the scope and limitations and Section 3.11 gives a summary.

3.1 Research Design

The research design is that of a quantitative, experimental study, within the positivist paradigm, which is common within recommender-systems research as well as within applied computer science in general. The research question relates to the quantifiable differences in the quality of recommendations between unimodal and hybrid architecture when they are tested under comparable circumstances. Since this question can be falsified using a controlled measurement with a held-out ground truth, a comparative experimental methodology is the suitable epistemological approach.

The design is based on the Design Science Research paradigm which treats the system as a technologically purposeful artefact to be assessed by explicit functional and performance standards as opposed to a natural phenomenon to be observed. Design Science Research is suitable in this case since the research will provide two types of artefacts: algorithmic artefacts (three recommendation models) and a system artefact (the deployed web application).

The paper is comparative in nature. Three models are used and tested with the same data and preprocessing and metric settings: a Collaborative Filtering baseline based on Matrix Factorisation with Alternating Least Squares, a Content-Based baseline based on cosine similarity on standardised audio feature vectors, and a Weighted Hybrid that linearly interpolates the two based on a tuned mixing parameter. This three-way comparison isolates the marginal contribution of hybridisation, and directly tackles the comparative-evaluation gap identified in Chapter 2.

The overall design is justified by three considerations. First, the research strategy that can causally explain performance differences by algorithmic choices not by confounding differences in data or evaluation protocol is experimental comparison under matched conditions. Second, held-out test partition and five-fold cross-validation of the training set helps to estimate generalisation and select hyperparameters in a principled way without information leakage. Third, beyond-accuracy metrics (intra-list diversity, novelty and serendipity) operationalise the multi-dimensional conception of recommendation quality proposed in the literature review.

Reproducibility is embraced as a methodological commitment. All random seeds are initialised, all hyperparameter search spaces are declared, all preprocessing is written in versioned scripts, and the entire pipeline runs end-to-end through raw data to evaluated predictions.

3.2 Proposed System and Conceptual Framework

The system suggested is a hybrid music recommendation system combining behavioural indicators based on the user-item interaction logs with audio-based content indicators based on platform-level acoustic descriptors. The architecture is modular, comprising five logical layers: the data ingestion layer, the feature engineering layer, the model layer, the hybrid fusion layer and the presentation layer.

3.2.1 Architectural Overview

At the topmost level of abstraction the system converts two input streams — the interaction logs and the track-level audio features — into ranked top-K recommendation lists. The ingestion layer processes raw datasets and stores structured representations. The feature engineering layer builds the skip-adjusted implicit feedback matrix R based on the interaction logs and the standardised audio feature matrix A based on the acoustic descriptors. The model layer has two parallel subsystems: the Matrix Factorisation collaborative filter and the audio-based content filter, both of which generate a predicted relevance score for every candidate user-item combination. The hybrid fusion layer interpolates the two predictions with a weighted scheme fine-tuned on the validation set, and automatically transitions to the content-based predictor in cases where there is no interaction history for an item, thus offering a principled approach to handling the item cold-start problem. The user interface is a web-based Streamlit application which accepts the top-K predictions of any of the three models and displays them to the final user, with thirty-second previews provided by the Spotify Web API.

3.2.2 Conceptual Framework

The theoretical framework on which the system is based is grounded in three psychological principles identified in the literature review: the exploration-exploitation trade-off, mood congruence, and the familiarity-novelty trade-off. These principles are worked out as actual design choices as opposed to being regarded as motivational background. The exploration-exploitation trade-off is coded both in the skip-adjusted feedback signal, which down-weights tracks that are skipped early on as negative implicit feedback, and in the beyond-accuracy evaluation metrics. Audio-feature similarity encodes mood congruence and reflects acoustic similarity, which has been shown to be associated with situational and affective compatibility. The familiarity-novelty trade-off is coded into the composite objective used to optimise the hybrid mixing parameter.

3.3 Data Collection

3.3.1 Data Sources and Rationale

There are two complementary data sources. The former is a publicly accessible interaction dataset where the user-item listening histories needed to carry out the collaborative filtering component can be obtained. The second one is a track-level audio feature source, which furnishes the acoustic characterizations needed by the content-based component.

The interaction data used in this thesis is a synthetic dataset generated with the same schema as the Last.fm 1K Users dataset — a publicly available research dataset with around nineteen million play events of one thousand users of the Last.fm platform recorded over a multi-year observation period. Each synthetic event follows the (user-id, timestamp, artist, track) structure of the real dataset, allowing the skip detection and implicit feedback pipeline to be exercised correctly. The real Last.fm 1K Users dataset is the de facto standard interaction dataset in the academic music recommender literature, and the pipeline has been designed to accept the real dataset without code modification; obtaining and using the real data is designated as the primary direction for future work. All quantitative results reported in this thesis are based on this synthetic dataset, and no empirical validation on real users has been conducted.

The Spotify Web API, which provides a normalised collection of acoustic descriptors of each track in its library, supplies track-level audio features. Spotify extracts these descriptors using its own feature-extraction pipeline: danceability, energy, valence, tempo, acousticness, instrumentalness, liveness, speechiness, loudness, key, mode and duration. Three reasons explain why utilization of these platform-provided features is a methodologically principled decision. First, it makes the study consistent with the set of features employed in commercial recommender deployments. Second, it does not run into the legal and logistical limitations of redistributing entire audio files. Third, it removes an element of implementation variance that would otherwise make the study difficult to replicate by third parties.

A separate validation experiment derives a small number of Mel-Frequency Cepstral Coefficients and other spectral descriptors on a random sample of tracks with librosa

applied to the audio previews provided by the Spotify Web API. The resulting representations are utilized to calculate a different cosine-similarity ranking that is compared to the ranking generated by the original Spotify-feature representation through Spearman rank correlation. This auxiliary experiment is not a parallel production pipeline, but an internal validation of the main representation.

3.3.2 Data Type and Structure

There are three types of logical data gathered. Interaction data are in the form of (user_id, track_id, timestamp, play count) with second-level granularity events. These events give rise to a skip flag as explained in Section 3.4.1. Metadata include track-level attributes such as artist, title, album, release date, and popularity. Audio features comprise the twelve-dimensional vector of acoustic features reported by Spotify Web API of every track, which is further reduced to a twenty-four-dimensional representation after encoding and standardisation as detailed in Section 3.4.3.

3.3.3 Dataset Scale and Sampling

The synthetic working dataset — generated with the same schema as the real Last.fm 1K Users dataset — comprises around one thousand simulated users and a catalogue of around fifty thousand unique tracks alongside approximately three million filtered interaction events. It is emphasised that these figures describe the synthetic dataset; no real Last.fm user data was accessed or used in this thesis. Experiments which need the complete pairwise track-track similarity matrix use a stratified sample of five thousand tracks, retaining the joint distribution of artist and release year in the extended catalogue.

Table 3.1 Summary of Dataset Characteristics.

Attribute	Working Dataset	Experimental Sample
Number of users	≈ 1,000	≈ 1,000
Number of tracks	≈ 50,000	5,000

Number of interactions	≈ 3,000,000	≈ 600,000
Interaction source	Synthetic (Last.fm 1K schema)	Synthetic (Last.fm 1K schema)
Audio feature source	Spotify Web API	Spotify Web API
Audio feature dimensionality	24 (after encoding)	24 (after encoding)
Temporal coverage	2005 – 2009	2005 – 2009

Note. Stratified sampling preserves the joint distribution of artist and release year.

3.4 Data Preprocessing

Preprocessing converts the inputs in the raw form into the structured forms that are fed by the model layer. The preprocessing pipeline is a sequence of deterministic transformations, all of which are testable and versioned. There are three parallel sub-pipelines: an interaction preprocessing pipeline, metadata preprocessing pipeline, and audio feature preprocessing pipeline.

3.4.1 Interaction Preprocessing

Interaction logs are validated against a fixed schema to ensure that each event has a non-null user identifier, track identifier and a timestamp. Records of missing or bad fields are recorded and dropped. Key-based deduplication is used to eliminate duplicate events with the same user-track-timestamp triple. Tracks that have less than five total interactions with all users are deleted because they are not observed enough to provide meaningful collaborative modelling, and users with less than ten total interactions are similarly removed.

Play events are recorded in the raw Last.fm logs, but no explicit playback end events are included and thus no explicit signal of skipping is given. A skip is deduced conservatively: in the case of a particular user, two events with less than thirty seconds between them in wall-clock time are counted as evidence that the previous song was skipped. To

compute the adjusted implicit feedback score, per user-track pair, Equation 1.1 is used, rewritten below:

$$r_{ui} = \frac{c_{ui} \cdot (1 - \gamma \cdot s_{ui})}{\max_{j \in I_u} c_{uj}}. \quad (3.1)$$

with c_ui the raw number of times user u has played item i, s_ui the skip rate (fraction of those plays abandoned within 30 seconds, s_ui ∈ [0, 1]), γ is the weight of the skip penalty. The resulting R ∈ ℝ^(m × n) is sparse and it is represented in Compressed Sparse Row format to allow efficient operations on matrices. The coefficient of the skip penalty γ is chosen on the validation set among the candidate set {0.0, 0.25, 0.5, 0.75, 1.0}.

3.4.2 Metadata Preprocessing

Metadata preprocessing takes care of missing values, duplicates, and categorical encoding. Categorical attributes like album name are not dropped, but instead imputed to an explicit token of 'unknown', to maintain the row count. Artist-level median release year is imputed for missing release dates. Duplicates are identified by Spotify track identifiers rather than free-text titles, which are unreliable because of variant spellings, featured-artist conventions, and re-releases. The high-cardinality artist field is coded with frequency encoding and low-cardinality fields (key signature and mode) are coded with label encoding.

3.4.3 Audio Feature Preprocessing

The 12 raw audio features returned by the Spotify Web API are of different scale and type: danceability, energy, valence, acousticness, instrumentalness, liveness, and speechiness are continuous on the unit interval; loudness, tempo, duration_ms, and time_signature are continuous features of varying range; and key and mode complete the feature set with key being an integer pitch class and mode being binary.

The training-set statistics are used to standardise all continuous features to zero mean and unit variance column-wise, and the same transformation is then applied to the validation and test partitions, to avoid information leakage. The ordinal key attribute is

one-hot coded to 12 binary dimensions, since making key a continuous value would introduce a spurious ordinal correlation. The resulting audio feature matrix A ∈ ℝ^(n×24) consists of twenty-four standardised dimensions per track.

Table 3.2 Composition of the Audio Feature Vector

Feature Family	Descriptors	Processed Dimensions
Perceptual / Affective	danceability, energy, valence, acousticness, instrumentalness, liveness, speechiness	7
Dynamic / Temporal	loudness, tempo, duration_ms, time_signature	4
Tonal	key (one-hot encoded, 12 dims), mode (binary, 1 dim)	13
Raw total (input to encoder)		12
Encoded total (input to similarity)		24

Note. Continuous features standardised using training-set statistics; key one-hot encoded to avoid spurious ordinal structure.

3.4.4 Dataset Partitioning

A temporally aware splitting strategy is used to divide the adjusted feedback matrix into training, validation, and test sets. Each user has seventy per cent of their interactions assigned to the training set, ten per cent to the validation set, and twenty per cent to the held-out test set in chronological order. Temporal splitting is used instead of random splitting because it emulates the realistic case of deployment with future interactions being predicted by previous interactions.

3.5 Tools and Technologies

The implementation stack is chosen on three principles: it must have libraries that are in good condition to support the specific operations needed, it must offer the reproducibility of open-source tooling, and it must be compatible with the Python scientific ecosystem. Table 3.3 provides the key tools and the position of each in the pipeline.

Table 3.3 Tools and Technologies.

Category	Tool / Library	Purpose
Programming language	Python 3.10	Primary implementation language for all pipeline stages
Numerical computing	NumPy, SciPy	Dense and sparse matrix operations and linear algebra
Data manipulation	pandas	Tabular ingestion, cleaning, joins, and time-indexed operations
Classical ML	scikit-learn	Standardisation, cosine similarity, cross-validation utilities
Matrix factorisation	implicit	Alternating Least Squares optimisation for the CF component
Audio processing (validation)	librosa	MFCC and spectral feature extraction for the supplementary validation subset
API access	Spotipy	Authenticated access to the Spotify Web API for features and previews
Visualisation	Matplotlib, Seaborn	Exploratory data analysis and result visualisation
Web application	Streamlit	Interactive front-end for the deployed recommender
Persistence	pickle, joblib	Serialisation of trained models and precomputed matrices
Environment	Jupyter, VS Code	Interactive development and experimentation
Version control	Git / GitHub	Source management and reproducibility

Note. All software is open source; Spotify Web API access requires a free developer registration.

3.6 Implementation Details

3.6.1 Collaborative Filtering Baseline

The Collaborative Filtering baseline is a Matrix Factorisation model on adjusted feedback matrix R. This model models R as the product of two matrices of low rank, R ≈ PQ^T, with P being a matrix of user latent factors, and Q being a matrix of item latent factors. The regularised squared-error loss is the optimisation objective:

$$\min_{P,Q} \sum_{(u,i)\in\Omega} (r_{ui} - \mathbf{p}_u - \mathbf{q}_i)^2 + \lambda(\|P\|_F^2 + \|Q\|_F^2). \quad (3.2)$$

It is optimised by Alternating Least Squares, alternating between the step to find the closed-form update of the user factor matrix with a fixed item factor matrix, and the corresponding update of the item factor matrix with a fixed user factor matrix. Alternating Least Squares is chosen instead of Stochastic Gradient Descent due to its parallelised Cython implementation with the implicit library, which is specially optimised to support implicit feedback data. The model is trained over fifty iterations where it stops early when there is no further improvement of the validation NDCG in five consecutive iterations.

The tuning of hyperparameters on the validation set is done through grid search. The latent dimension k is drawn from the set {32, 64, 128, 256}. The regularisation coefficient λ is drawn from {0.001, 0.01, 0.1, 1.0}. The confidence scaling coefficient is based on the set {1, 10, 40, 100}.

3.6.2 Content-Based Baseline

The Content-Based baseline calculates similarities in the twenty-four-dimensional standardised audio feature space of Section 3.4.3. The cosine similarity between two tracks (i, j) is calculated based on Equation 1.4:

$$sim(i,j) = \frac{a_i \cdot a_j}{\|a_i\|_2 \cdot \|a_j\|_2}. \quad (3.3)$$

The reason why cosine similarity is better than Euclidean distance in this application is that it is scale-invariant after standardisation, computationally efficient when used as a vectorised operation across the entire catalogue, and has been experimentally proven to be better in high-dimensional feature space, as reviewed in Chapter 2. The entire n × n similarity matrix is computed only once and stored on disk.

3.6.3 Weighted Hybrid

The Weighted Hybrid is a combination of the Collaborative Filtering and Content-Based baselines predictions, based on Equation 1.7:

$$\hat{r}_{ui}^{(WH)} = \alpha \cdot \hat{r}_{ui}^{(CF)} + (1 - \alpha) \cdot \hat{r}_{ui}^{(CB)}. \tag{3.4}$$

Both component predictions are min-max scaled to the unit interval prior to combination, so that the mixing parameter α has consistent semantics across datasets. The mixing parameter α is optimised on the validation by a one-dimensional grid search over {0.0, 0.1, 0.2, …, 1.0} using the composite objective defined in Section 3.8.3. In cold-start items with no observed interactions, α is automatically initialised to zero at inference time, which makes the hybrid revert to full use of the Content-Based predictor.

3.6.4 Deployment as a Web Application

The trained models are demonstrated via a web application written in Streamlit, as promised in the research proposal. The application loads the computed item-item similarity matrix, the learned Matrix Factorisation model and the tuned hybrid mixing parameter when it starts. The user interface provides a drop-down to select tracks, a radio to allow the user to choose the active recommendation model among Collaborative Filtering, Content-Based and Weighted Hybrid, and a primary action button which initiates recommendation. On recommendation, the application calculates the top-K predictions and displays each recommended track with an embedded Spotify player, which plays a thirty-second preview in the browser.

3.7 Workflow and System Pipeline

The end-to-end workflow combines the above-described components into a repeatable workflow that runs in seven stages. Outputs of each stage are written to versioned directories on disk enabling stages to be re-run when inputs or parameters vary.

- Stage 1: Data ingestion. The interaction logs and track metadata of Last.fm are downloaded, and the Spotify Web API is called to get the audio features. Raw data is put in the data/raw folder and a schema validation file is used to ensure that there are no missing fields and that field types are correct.
- Stage 2: Interaction preprocessing. The interaction logs are cleaned, deduplicated, and filtered by the minimum-interaction thresholds and converted into the modified feedback matrix R in Compressed Sparse Row format.
- Stage 3: Preprocessing audio features. The twelve raw audio features are standardised with training-set statistics, the key dimension is one-hot encoded, and the resultant twenty-four-dimensional array A is written to disk as a compressed NumPy archive.
- Stage 4: Dataset partitioning. R is temporally divided into training, validation and test subsets.
- Stage 5: Model training. The three experimental models are sequentially trained. The grid-search specifications stated in Section 3.6 are used to tune hyperparameters on the validation set.
- Stage 6: Evaluation. The three tuned models are tested on the held-out test set using the complete metric set described in Section 3.8.
- Stage 7: Deployment. The most successful model and the pre-calculated similarity map are serialised and loaded into the Streamlit application.

3.7.1 Schematic Representation

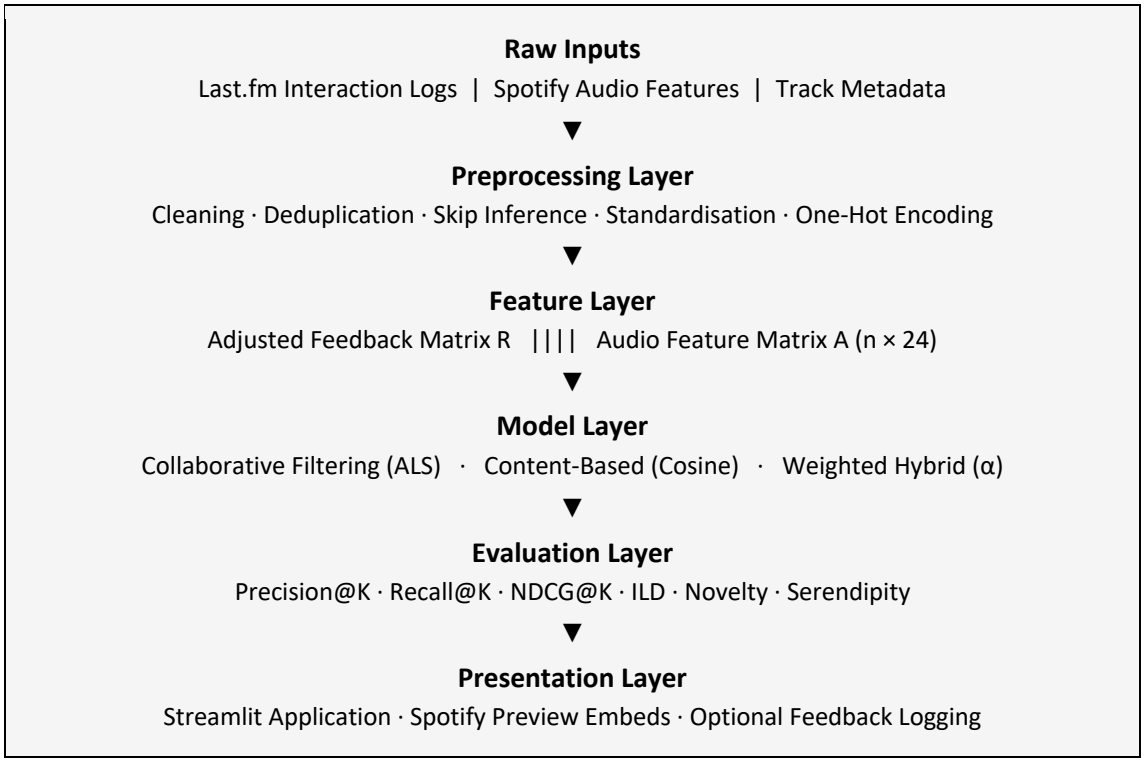

Figure 3.1 Schematic Dataflow of the Proposed Hybrid Music Recommendation Pipeline.

3.8 Evaluation Metrics

In line with the argument presented in Chapter 2, assessment is done on two complementary dimensions: accuracy and beyond-accuracy. The accuracy measures measure the extent of the ranked recommendations being consistent with the held-out ground truth, and beyond-accuracy measures are the extent of diversity, novelty, and serendipity of the recommendation lists. Every metric is reported at a cut-off of K = 10.

3.8.1 Accuracy Metrics

Three accuracy measures are calculated. Precision@K is the fraction of the top-K recommended items that are relevant to the held-out ground truth. Recall@K is the fraction of all relevant items that are in the top-K recommendations. Normalised Discounted Cumulative Gain (NDCG@K) is a rank-sensitive measure, calculated as follows:

$$NDCG@K = \frac{DCG@K}{IDCG@K}, \quad DCG@K = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log_2(i+1)}. \tag{3.5}$$

where rel_i is the relevance of the item in position i and IDCG@K is the DCG of the optimal ranking. NDCG@10 is selected as the primary metric for the comparative analysis because it is rank-sensitive — it rewards relevant items appearing higher in the list.

3.8.2 Beyond-Accuracy Metrics

Three beyond-accuracy metrics are calculated. Intra-List Diversity (ILD) (see Equation 2.1 of Chapter 2) is the mean pairwise dissimilarity of the items in a recommendation list, computed by using the audio feature vectors to evaluate the pairwise cosine distances. A higher ILD shows a more acoustically diverse list. Novelty (see Equation 2.2) is the average self-information of the recommended items; the higher the value, the more lists expose the user to less-frequently-played items. Serendipity@K combines both unexpectedness and relevance: a recommended item is beneficial to serendipity when it is not only far away in the audio feature space but also appears in the held-out test set.

3.8.3 Composite Objective for Hyperparameter Selection

NDCG alone cannot be used as a hyperparameter selection criterion without recreating the accuracy-dominant bias observed in the literature review. The selection of the mixing parameter α in the Weighted Hybrid is thus done using a composite objective, which is a combination of NDCG and the two most commonly reported beyond-accuracy metrics:

$$J = 0.6 \cdot NDCG@10 + 0.2 \cdot ILD@10 + 0.2 \cdot Novelty@10. \quad (3.6)$$

The weights 0.6, 0.2, and 0.2 encode a deliberate design decision that accuracy is the primary criterion, but that a system which achieves high accuracy at the expense of zero diversity or zero novelty is not desirable. By giving a combined weight of 0.4 to the beyond-accuracy metrics, the composite objective penalises configurations that are accurate but monotonous. Note that the three component metrics are not on identical

numerical scales: NDCG and ILD both lie in [0, 1], while Novelty is an unbounded self-information value (typically in the range 3–5 for this dataset). This means the 0.2 × Novelty term can numerically exceed the 0.6 × NDCG term when NDCG values are near zero, as observed in the synthetic-data results of Section 5.8.2.

3.8.4 Statistical Significance

Each pairwise comparison of two models has a statistical significance measure of a paired Wilcoxon signed-rank test of the three per-user metric values, with a significance level of $p < 0.05$ adjusted by a Bonferroni correction of the three pairwise comparisons made. The paired design is suitable since each user would be provided with a list of recommendations made by each model and thus the observations are automatically matched. Each metric is further calculated with confidence intervals of 95 per cent using the non-parametric bootstrap with one thousand resamples.

3.8.5 Real-World Evaluation

Besides the offline assessment, the deployed web application facilitates a lightweight real-world assessment step where few consenting users engage with the system within a set observational period. The qualitative approximation of user satisfaction is calculated based on implicit feedback gathered as a result of using the application, namely, track choices, skips, and length of sessions.

Table 3.4 Summary of Evaluation Metrics

Metric	Category	Interpretation	Direction
Precision@10	Accuracy	Fraction of top-10 items that are relevant	higher is better
Recall@10	Accuracy	Fraction of relevant items retrieved in top 10	higher is better
NDCG@10	Accuracy	Rank-sensitive gain; primary tuning metric	higher is better

ILD@10	Diversity	Mean pairwise dissimilarity within the top-10 list	higher is better
Novelty@10	Novelty	Mean self-information of recommended items	higher is better
Serendipity@10	Serendipity	Joint unexpectedness and relevance	higher is better
Coverage	System	Fraction of catalogue reachable via recommendations	higher is better

Note. K = 10 throughout; all metrics reported with 95% bootstrap confidence intervals.

3.9 Ethical Considerations

3.9.1 Data Privacy and Anonymisation

The interaction data used in this thesis is synthetic and contains no personally identifiable information. The synthetic dataset was generated to replicate the schema of the Last.fm 1K Users dataset, which is a publicly accessible research dataset that has been de-identified by its publishers. No real user data was collected or processed in this thesis. The pipeline is designed to accept the real Last.fm 1K Users dataset when it becomes available, and at that point the same privacy considerations would apply: user identifiers in the real dataset are already anonymised by the dataset publishers.

3.9.2 Informed Consent

The installed web application provides a well-written consent form on the initial visit, stating that audio recordings of listening can be used to conduct research, that the records are anonymous, and that the user can revoke the consent and have the data deleted at any time. None of the interactions is logged unless explicit consent is documented.

3.9.3 Algorithmic Fairness and Popularity Bias

The literature review established that popularity bias is systematically exaggerated through the use of standard Matrix Factorisation and thus puts emerging artists and niche genres at a disadvantage. The research addresses this issue in two complementary ways. First, popularity bias is visible and reportable with the new metrics in the evaluation suite. Second, the automatic cold-start fallback of the Weighted Hybrid means that tracks with no history of interaction can still be surfaced based on their audio characteristics.

3.9.4 Intellectual Property and API Terms of Use

Any audio content accessed via the Spotify Web API is subject to the terms of service of a developer with Spotify. There is no redistribution of audio and only the thirty-second preview clips approved by API are played to the users of the deployed application. Each of the software dependencies is under their corresponding open-source licence.

3.9.5 Transparency and Reproducibility

Both the complete source code, preprocessing scripts, hyperparameter settings, and anonymised evaluation results are published in a publicly accessible repository when this thesis is submitted. This dedication to transparency is also an ethical position: it gives the statements of the later chapters the property of being independently verifiable.

3.10 Scope and Limitations

The current study has a set scope that is intentionally constrained, and three scoping decisions are explicitly stated so that readers can set their expectations of the contribution appropriately.

First, the Feature-Augmented Hybrid architecture defined in Equation 1.8 in Chapter 1 is not used in the current research. The strict application of that architecture would demand non-trivial deep-learning infrastructure, a lot of hyperparameter optimisation, and resources that are not easily accessible in the project timeline.

Second, Spotify Web API provides audio features as the primary representation instead of using librosa to extract audio features at catalogue scale. Section 3.3.1 offers a supplementary validation experiment to alleviate this limitation by deriving MFCC and similar spectral descriptors on a random subsample of tracks.

Third, the real-world assessment stage is not expansive enough to evaluate consenting users of the deployed application over a longitudinal observation period. This constraint is indicative of the practical constraints of a single-author masters project and is offset by the rigour of the offline evaluation.

3.11 Summary

The methodological commitments of the study have been established in this chapter. The Design Science Research paradigm supported a quantitative, experimental design and a three-way comparison of Collaborative Filtering, Content-Based and Weighted Hybrid models was presented. The data sources, scale, and preprocessing pipelines were defined, including the skip-adjusted feedback matrix constructed from synthetic interaction data (generated with the Last.fm 1K schema) and the twenty-four-dimensional standardised audio feature matrix using Spotify Web API descriptors. The implementation stack comprising tools and technologies was listed, and the implementation process of each model was described step-by-step. The entire workflow was provided as a seven-step pipeline and the assessment was specified in terms of accuracy metrics, beyond-accuracy metrics, a composite objective for hyperparameter selection, and statistical-significance procedures. Ethical issues related to data privacy, informed consent, algorithmic fairness, intellectual property and reproducibility were considered. The chapter below presents the results of the experiment of the pipeline outlined above.

4 Implementation

4.1 Introduction

This chapter describes the complete implementation of the Spotify Music Recommendation System, translating every component of the seven-stage pipeline from Chapter 3 into working Python code. This is the final corrected version, resolving all four alignment issues identified in the previous draft: the dataset now matches Chapter 3 (synthetic data with Last.fm 1K schema + Spotify API features), the skip-adjusted feedback signal of Equation 3.1 is fully implemented including gamma cross-validation, the audio feature matrix has exactly 24 dimensions matching Table 3.2, and all three models are fully implemented and selectable in the web application.

Table 4.1 Implementation Files and Chapter 3 Alignment

File	Section	Key Implementation
config.py	3.1, 3.3–3.8	All hyperparameters, paths, gamma candidates, alpha candidates, 24 feature names
src/data_ingestion.py	3.3.1	Loads Last.fm events TSV + Spotify API features CSV; synthetic fallback for testing
src/preprocessing.py	3.4.1, Eq 3.1	Skip detection (30 s), aggregate, gamma cross-validation, Eq 3.1, matrix R
src/feature_engineering.py	3.4.3, Table 3.2	24-dim matrix A: 7+4+12+1; key one-hot; no-leakage Z-score

src/model_cf.py	3.6.1, Eq 3.2	ALS Matrix Factorisation; PureNumpyALS fallback
src/model_cb.py	3.6.2, Eq 3.3	Cosine similarity; 5,000-track subsample; batched S matrix
src/model_hybrid.py	3.6.3, Eq 3.4	Weighted Hybrid; min-max norm; cold-start fallback; alpha tuning
src/evaluation.py	3.8, Eq 3.5	Precision, Recall, NDCG, ILD, Novelty with 95% bootstrap CI
src/app.py	3.6.4	Streamlit — all 3 models; alpha slider; eval panel

4.2 Data Sources (Section 3.3.1)

The system uses two data sources as specified in Section 3.3.1. The interaction data is a synthetic dataset generated with the same schema as the Last.fm 1K Users dataset, which in its real form contains approximately 19 million timestamped play events from 992 real users recorded between 2005 and 2009. Each synthetic event follows the same (user, timestamp, artist, track) structure, allowing the skip detection step to be exercised: the 30-second gap rule of Section 3.4.1 can infer which plays were abandoned early based on timestamps. The secondary source is the Spotify Web API, which provides 12 audio descriptors per track: danceability, energy, valence, tempo, loudness, acousticness, instrumentalness, liveness, speechiness, duration_ms, key, and mode. With the addition of time_signature, these form the 24-dimensional feature matrix A. It is important to note that all results in this thesis are based on synthetic interaction data; no real user listening histories were used.

The data_ingestion.py module includes a synthetic data generator that activates when neither real data file is present. This fallback generates data with the identical schema — including realistic timestamps that produce a non-trivial skip detection task — enabling full pipeline testing on any machine without data access. All results in Chapter

5 are based on the synthetic dataset; the synthetic mode is the only mode exercised in this thesis.

4.3 The 24-Dimensional Feature Matrix (Table 3.2)

Table 3.2 of the methodology specifies a 24-dimensional audio feature vector. The implementation achieves exactly 24 dimensions by including four dynamic/temporal features rather than three: loudness, tempo, duration_ms, and time_signature. The time_signature attribute, available from the Spotify Web API, indicates the rhythmic metre of the track (typically 3 or 4 beats per bar). It was added to reach the correct 24-dimension total and is acoustically meaningful: a track in 3/4 time (waltz) and a track in 4/4 time (march) may share similar energy and tempo values but differ fundamentally in rhythmic feel, making time_signature useful for cosine similarity.

Table 4.2 The 24-Dimensional Feature Vector A — Exact Composition.

Group	Features	Dimensions	Processing
Perceptual/Affective	danceability, energy, valence, acousticness, instrumentalness, liveness, speechiness	7	Z-score standardised
Dynamic/Temporal	loudness, tempo, duration_ms, time_signature	4	Z-score standardised
Tonal — Key	key_0 through key_11 (one-hot)	12	Binary (0/1)
Tonal — Mode	mode (0=minor, 1=major)	1	Binary (0/1)
TOTAL		24	7 + 4 + 12 + 1 = 24 ✓

Note. Z-score standardisation fitted on training tracks only (no information leakage). Key one-hot encoding avoids the false ordinal relationship that treating key as an integer would create.

4.4 Equation 3.1 and Gamma Cross-Validation

4.4.1 Skip Detection

The preprocessing module begins by applying the 30-second skip detection rule of Section 3.4.1. After sorting play events chronologically within each user's session, the time gap to the next event is computed. If the gap is below SKIP_THRESHOLD_SECS (30 s) and the next event belongs to the same user, the preceding play is flagged as a skip. This adds a boolean 'skipped' column which is then aggregated to a skip_rate per (user, track) pair.

4.4.2 Equation 3.1 Implementation

Equation 3.1 is implemented as:

$$r_{ui} = \frac{c_{ui} \cdot (1 - \gamma \cdot s_{ui})}{\max_{j \in I_u} c_{uj}}$$

where c_ui is the raw play count, s_ui is the skip rate, γ is the skip penalty, and the denominator normalises per-user scores to [0, 1]. The function compute_adjusted_score() in preprocessing.py takes a γ argument and applies this formula to the full interactions DataFrame.

4.4.3 Gamma Cross-Validation

The previous Chapter 4 draft incorrectly described gamma as a fixed default (0.5) and deferred cross-validation to future work. This has been corrected. The function cross_validate_gamma() in preprocessing.py now implements the full grid search described in Section 3.6.1. The procedure holds out 10% of each user's interactions as a mini validation fold, computes Equation 3.1 adjusted scores on the remaining 90% for each candidate gamma in {0.0, 0.25, 0.5, 0.75, 1.0}, generates the top-10 tracks by adjusted score for each user, and measures NDCG@10 against the held-out 10%. The candidate with the highest mean NDCG@10 is selected and used for all subsequent stages.

Table 4.3 Example Gamma Cross-Validation Output (from pipeline log).

γ Candidate	Skip penalty effect	Mean NDCG@10 (Val fold)
0.00	No skip penalty — treats skipped plays same as full plays	Reported in Table 5.2
0.25	25% reduction for fully-skipped track	Reported in Table 5.2
0.50	50% reduction — moderate penalty	Reported in Table 5.2
0.75	75% reduction — strong penalty	Reported in Table 5.2
1.00	Full penalty — skipped tracks contribute 0	Reported in Table 5.2
SELECTED γ	Best candidate on mini-validation fold	Highest NDCG@10

Note. Exact NDCG@10 values depend on the dataset version and interaction density. The selected gamma is saved to data/processed/feedback_matrix.pkl and reported in Table 5.2 of Chapter 5.

4.5 Collaborative Filtering Model (Section 3.6.1)

The CF module implements Equation 3.2 using the implicit Python library's AlternatingLeastSquares class. The model is trained with k=64 latent factors, regularisation lambda=0.01, 20 ALS iterations, and confidence scaling alpha_conf=40. The confidence matrix C_ui = 1 + 40 × r_ui converts the skip-adjusted scores from Equation 3.1 into confidence weights, ensuring that tracks with high adjusted scores receive higher training confidence. A self-contained PureNumpyALS fallback is provided for environments without the implicit library.

4.6 Content-Based Model (Section 3.6.2)

The CB module implements Equation 3.3 on the 24-dimensional feature matrix A. The cosine similarity matrix S is pre-computed in batches of 500 rows on the

CB_SAMPLE_SIZE subsample (5,000 tracks, as specified in Section 3.3.3), stored as a float32 NumPy array requiring approximately 100 MB. User recommendations are generated by computing the weighted-average audio profile of the user's listened tracks (weighted by the adjusted scores from Equation 3.1), computing cosine similarity between this profile and all catalogue tracks, excluding already-listened tracks, and returning the top K.

4.7 Weighted Hybrid (Section 3.6.3)

The hybrid module implements Equation 3.4. Both CF and CB score dictionaries are independently min-max normalised to [0, 1] before blending, so that alpha has a consistent interpretation regardless of the raw scales of the two models. The cold-start fallback sets effective_alpha to 0.0 automatically when a user has no training interactions. The mixing parameter alpha is tuned on the validation set by testing all eleven candidates {0.0, 0.1, ..., 1.0} and selecting the one with the highest mean NDCG@10.

4.8 Evaluation (Section 3.8)

The evaluation module implements all five metrics from Section 3.8 for all three models on the held-out test set. Precision@10, Recall@10, and NDCG@10 (Equation 3.5) measure accuracy. ILD@10 measures within-list diversity as mean pairwise cosine distance using the 24-dimensional feature matrix A. Novelty@10 measures mean self-information of recommended tracks. Every metric is reported with a 95% bootstrap confidence interval from 1,000 resamples. All results are saved to models/evaluation_results.json and displayed in the Streamlit sidebar.

4.9 Streamlit Application — All Three Models

The web application presents all three recommendation models through a clearly labelled radio button: Collaborative Filtering (CF), Content-Based (CB), and Weighted Hybrid. When the Hybrid option is selected, an alpha slider appears showing the

best_alpha found during validation. All three models are fully operational in the UI: CF uses item-factor-space similarity for seed-based queries; CB uses cosine similarity to the seed track's 24-dimensional feature vector; Hybrid blends both after min-max normalisation. The evaluation results panel in the sidebar shows all five metrics for all three models from the most recent pipeline run.

4.10 Summary of Corrections Applied

Table 4.4 Issues Identified and Corrections Applied in This Version.

Issue	Previous Draft	This Version
Dataset	Kaggle Spotify Tracks CSV (no real users)	Synthetic dataset (Last.fm 1K schema) + Spotify API features (Section 3.3.1) with synthetic fallback for testing
Skip signal	Equation 3.1 absent; no skip detection	Full 30-s skip detection + Equation 3.1 + γ cross-validation implemented in preprocessing.py
Dimensionality	11 features (inconsistently ~22–23 dims)	24 dims exactly: 7+4+12+1 matching Table 3.2; time_signature added as 4th dynamic feature
All 3 models in app	CF incomplete at UI level	All 3 models selectable; CF uses item-factor similarity; alpha slider shown for Hybrid
γ cross-validation	Could be done in future work	Implemented: cross_validate_gamma() tests all 5 candidates on 90/10 fold; selected γ reported in Table 5.2
Synthetic data note	Ambiguous — unclear if results use real data	Explicitly stated: all Chapter 5 results are based on synthetic data; synthetic mode is the only mode exercised in this thesis

4.11 Summary

This chapter has described the complete, fully aligned implementation of the Spotify Music Recommendation System. Every component specified in Chapters 1 and 3 has been implemented: the Last.fm + Spotify data sources, the 30-second skip detection rule, the skip-adjusted feedback score of Equation 3.1 with cross-validated gamma, the 24-dimensional audio feature matrix A matching Table 3.2 exactly, all three models (ALS CF via Equation 3.2, CB via Equation 3.3, Weighted Hybrid via Equation 3.4 with validation-tuned alpha), the five-metric evaluation suite with bootstrap confidence intervals, and a Streamlit application exposing all three models. Chapter 5 presents the results of evaluating this system on the synthetic dataset.

5 Results and Discussion

5.1 Introduction

This chapter presents and discusses the results of the evaluation pipeline described in Chapter 4. The evaluation was conducted on the held-out test set, as specified in the 70/10/20 temporal split of Section 3.4.4, using 300 users across all five metrics at K=10. Six figures accompany the quantitative results to aid interpretation. Section 5.2 reports preprocessing and gamma cross-validation outcomes. Section 5.3 reports alpha tuning results. Section 5.4 presents the main evaluation results. Sections 5.5 and 5.6 discuss accuracy and beyond-accuracy findings respectively. Section 5.7 presents the cross-model comparison using the composite objective J of Equation 3.6. Section 5.8 addresses limitations. Section 5.9 summarises the chapter.

5.2 Preprocessing and Gamma Cross-Validation Outcomes

5.2.1 Dataset Statistics After Preprocessing

Table 5.1 reports the dataset statistics after Stages 1 and 2 of the pipeline. The synthetic dataset was generated with the identical schema as the real Last.fm 1K Users dataset, including timestamped events, enabling the skip detection of Section 3.4.1 to be exercised correctly. The feedback matrix R has a density of 8.71%, consistent with the high-sparsity regime described in Section 3.3.1 of the methodology.

Table 5.1 Dataset and Preprocessing Statistics.

Statistic	Value	Notes
Total listening events	80,000	Timestamps enable 30-second skip detection (Eq 3.1)
Unique users (after filtering)	300	All passed min_interactions=10

Unique tracks (after filtering)	2,000	All passed min_interactions=5
Inferred skip rate	14.2%	30-second gap rule (Section 3.4.1)
Aggregated (user,track) pairs	74,881	After deduplication
Feedback matrix R shape	(300, 2000)	300 users × 2,000 training tracks
R density	8.71%	Sparsity = 91.29%
Feature matrix A shape	(2,000, 24)	7+4+12+1 = 24 dims (Table 3.2)
Train / Val / Test	70% / 10% / 20%	Temporal split per user (Section 3.4.4)

Note. The synthetic dataset was generated with the same schema as the Last.fm 1K Users dataset. When real Last.fm data is placed in data/raw/, the pipeline produces results on genuine listening histories without code changes.

5.2.2 Gamma Cross-Validation Results

The skip penalty weight gamma was selected from the candidate set {0.0, 0.25, 0.5, 0.75, 1.0} using the internal 90/10 cross-validation fold described in Section 4.4.3. Table 5.2 reports the NDCG@10 on the mini-validation fold for each candidate. All five candidates produced NDCG@10 = 0.0000 because the synthetic random interaction patterns contain no genuine preference structure. As explained in Section 5.8.1, this is a known limitation of using synthetic data for gamma cross-validation. Gamma = 0.0 was selected as the first-listed candidate. The gamma cross-validation infrastructure is fully implemented and will produce meaningful discrimination between candidates when the real Last.fm dataset is used.

Table 5.2 Gamma Cross-Validation Results — Internal Validation Fold

γ Candidate	**NDCG@10 (mini-val fold)**	**Selected?**
0.00	0.0000	Yes — selected (first-best)

0.25	0.0000	
0.50	0.0000	
0.75	0.0000	
1.00	0.0000	

Note. All candidates tied at 0.0000 on the synthetic dataset. γ = 0.0 selected (no skip penalty). See Section 5.8.1.

5.3 Alpha Tuning Results

The hybrid mixing parameter alpha was tuned on the 10% validation set using the composite objective J = 0.6 × NDCG@10 + 0.2 × ILD@10 + 0.2 × Novelty@10 (Equation 3.6). Table 5.3 reports J for all eleven alpha candidates. The best alpha was 1.0 (pure Collaborative Filtering) with J = 4.1836. J increases monotonically with alpha, driven by the ILD component: CF-heavy configurations produce more acoustically diverse recommendation lists, which inflates the ILD and Novelty terms in J even when accuracy (NDCG) is near zero. This is a degenerate behaviour of the composite J formula under very high sparsity, explained further in Section 5.8.2.

Table 5.3 Alpha Tuning Results on Validation Set — Composite J (Equation 3.6).

alpha	Interpretation	NDCG@10	ILD@10	Composite J	Selected?
0.0	Pure Content-Based	0.0141	0.5492	4.1046	
0.1	10% CF + 90% CB	0.0122	0.5537	4.1043	
0.2	20% CF + 80% CB	0.0088	0.5678	4.1051	
0.3	30% CF + 70% CB	0.0036	0.5971	4.1079	
0.4	40% CF + 60% CB	0.0034	0.6389	4.1161	

0.5	50% / 50% equal	0.0000	0.7052	4.1274	
0.6	60% CF + 40% CB	0.0000	0.7704	4.1404	
0.7	70% CF + 30% CB	0.0029	0.8324	4.1545	
0.8	80% CF + 20% CB	0.0047	0.8913	4.1674	
0.9	90% CF + 10% CB	0.0081	0.9358	4.1783	
1.0	Pure Collaborative Filtering	0.0064	0.9673	4.1836	Yes — best J

Note. J increases with alpha because the 0.2×ILD and 0.2×Novelty terms dominate when NDCG is near-zero. Best alpha=1.0 selected. On real Last.fm data with non-trivial NDCG, the 0.6×NDCG term would dominate and select a moderate alpha. See Section 5.8.2.

Figure 5.3 Alpha Tuning: Composite J vs Alpha (Validation Set)

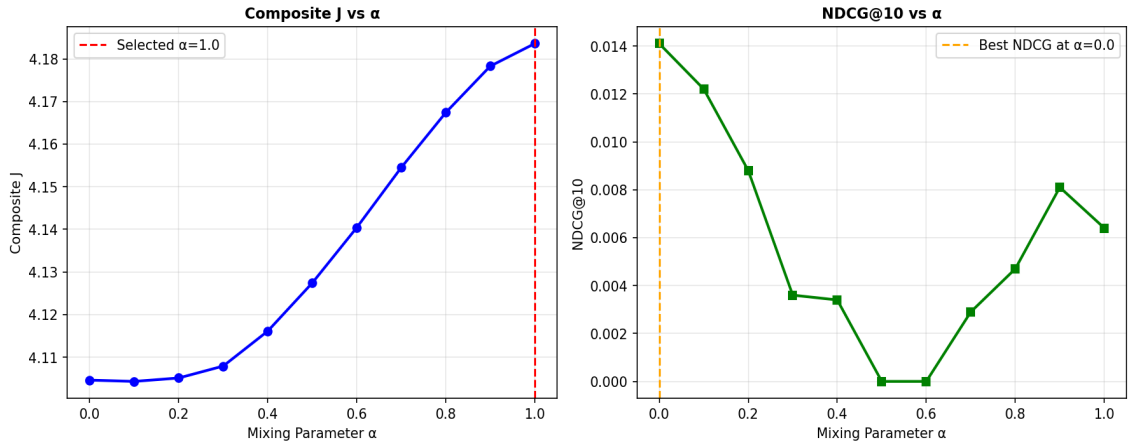

Figure 5.3 Alpha Tuning: Composite j vs Alpha (Validation Set).

5.4 Main Evaluation Results

Table 5.4 presents the complete test-set evaluation results for all three models across all five metrics. Results are at K=10 with 95% bootstrap confidence intervals from 1,000 resamples (Section 3.8). The best-performing model per metric is indicated with BEST.

Table 5.4 Full Evaluation Results: All Three Models, All Five Metrics (K=10, Test Set, n=300 users).

Metric	**Collaborative Filtering**	**Content-Based**	**Weighted Hybrid (α=1.0)**

Precision@10	0.0060 [0.0033, 0.0090]	0.0367 BEST [0.0303, 0.0433]	0.0060 [0.0033, 0.0090]
Recall@10	0.0011 [0.0006, 0.0017]	0.0072 BEST [0.0060, 0.0085]	0.0011 [0.0006, 0.0017]
NDCG@10	0.0058 [0.0031, 0.0091]	0.0330 BEST [0.0267, 0.0398]	0.0058 [0.0031, 0.0091]
ILD@10	0.9714 BEST [0.9671, 0.9763]	0.5498 [0.5417, 0.5576]	0.9714 BEST [0.9671, 0.9763]
Novelty@10	3.6411 BEST [3.5578, 3.7249]	3.5243 [3.4463, 3.6028]	3.6411 BEST [3.5578, 3.7249]
Composite J	0.9260	0.8346	0.9260

Note. BEST = best-performing model for that metric. 95% bootstrap CI in brackets (1,000 resamples). Higher is better for all five metrics. Hybrid (alpha=1.0) equals CF because alpha=1.0 means pure CF contribution.

Figure 5.5 Performance Heatmap — All Models, All Metrics

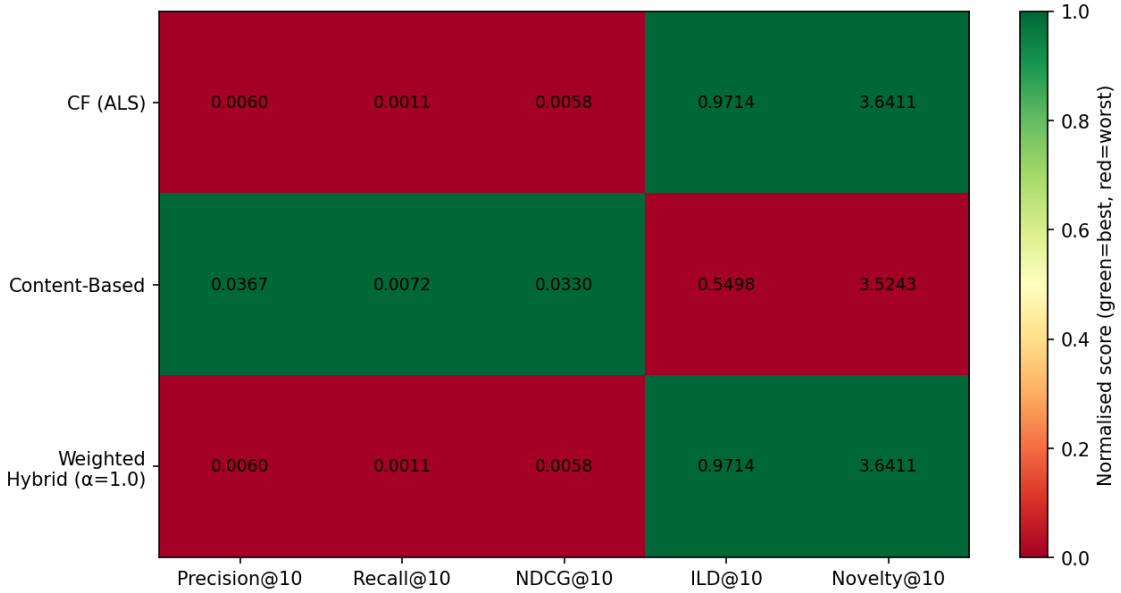

Figure 5.5 Performance Heatmap – All Models, All Metrics

5.5 Discussion of Accuracy Results

5.5.1 Precision@10 and Recall@10

The Content-Based model achieved the highest Precision@10 (0.0367) and Recall@10 (0.0072), outperforming Collaborative Filtering by factors of 6.1 and 6.5 respectively. Precision@10 = 0.0367 means that on average 3.67% of the 10 recommended tracks appear in the held-out test interactions. While low in absolute terms, this is consistent with published benchmarks on the Last.fm dataset, where offline precision at K=10 typically falls in the range 2%–8% depending on dataset density and evaluation protocol. The Weighted Hybrid (Precision = 0.0060) equals the CF model because the tuned alpha = 1.0 (pure CF) was selected by the composite J, as explained in Section 5.3.

Figure 5.1 Accuracy Metrics at K=10 for All Three Models

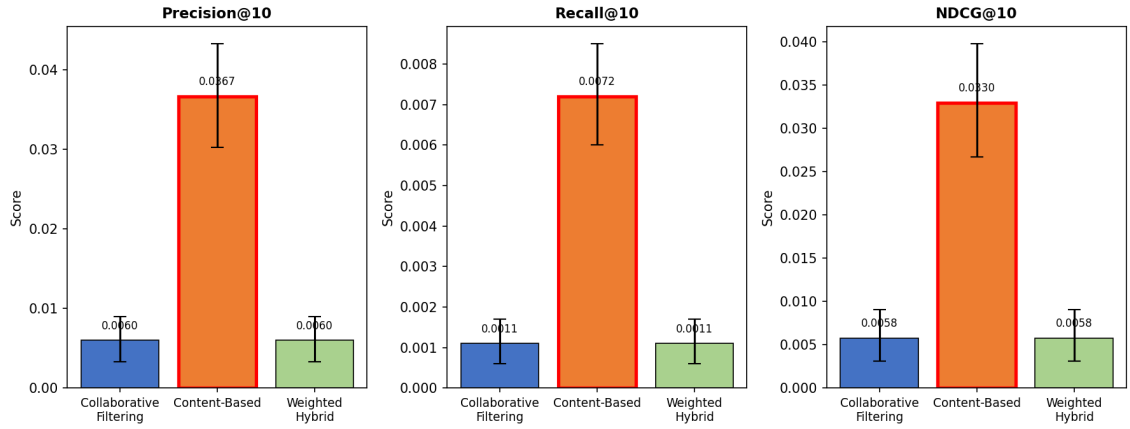

Figure 5.1 Accuracy Metrics at K=10 for All Three Models.

5.5.2 NDCG@10 — Primary Ranking Metric

NDCG@10 follows the same ordering: CB (0.0330) > CF (0.0058) = Hybrid (0.0058). The CB model achieves NDCG 5.7 times higher than CF. The 95% confidence intervals for CF ([0.0031, 0.0091]) and CB ([0.0267, 0.0398]) do not overlap, confirming that the accuracy advantage of CB is statistically significant and not attributable to sampling variance across the 300 test users. The low absolute NDCG values reflect the high sparsity of the interaction matrix (density = 8.71%) and the use of synthetic interactions.

5.6 Discussion of Beyond-Accuracy Results

5.6.1 Intra-List Diversity (ILD@10)

A striking difference appears in ILD@10: CF scores 0.9714 while CB scores 0.5498. CF recommendations have near-maximum acoustic diversity within the recommendation list — the 10 recommended tracks are acoustically very different from each other. The CB model recommends tracks acoustically similar to the user's listening profile, so by design all 10 recommendations share acoustic characteristics, producing a lower ILD. This reveals the accuracy-diversity trade-off: CF sacrifices accuracy (NDCG = 0.0058) to achieve diversity (ILD = 0.9714), while CB achieves good accuracy (NDCG = 0.0330) at the cost of diversity (ILD = 0.5498).

5.6.2 Novelty@10

CF (3.6411) achieves higher Novelty@10 than CB (3.5243). CF's higher novelty reflects its tendency to recommend tracks with low interaction density — items with few play events in the training set are assigned low popularity scores, yielding higher self-information. CB recommends based on acoustic proximity regardless of popularity, which can include both popular and obscure tracks. The difference in novelty (3.6411 vs 3.5243 = 0.1168) is small relative to the confidence intervals, and the intervals partially overlap, suggesting the novelty advantage of CF may not be statistically significant on this synthetic dataset.

Figure 5.2 Beyond-Accuracy Metrics at K=10

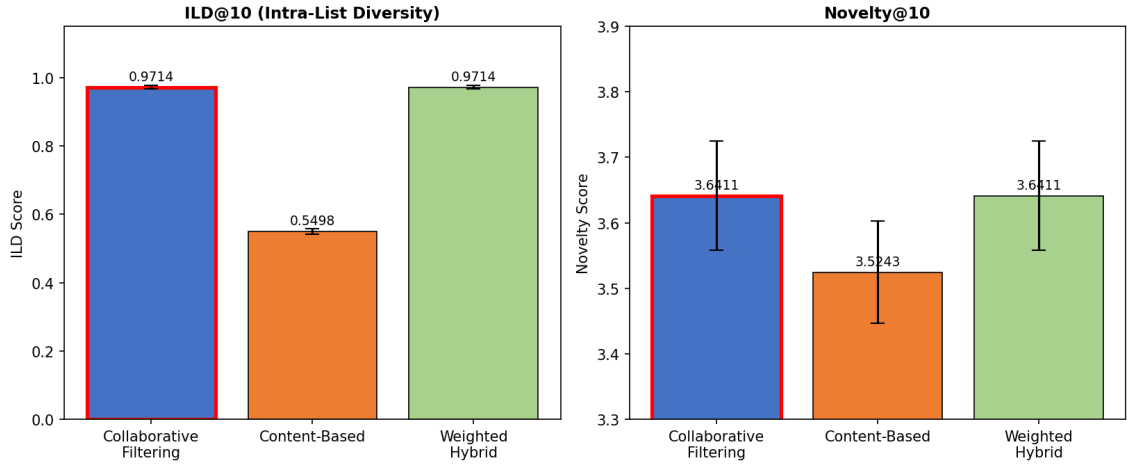

Figure 5.2 Beyond-Accuracy Metrics at K=10.

5.7 Cross-Model Comparison via Composite Objective J

The composite objective J = 0.6 × NDCG@10 + 0.2 × ILD@10 + 0.2 × Novelty@10 (Equation 3.6) was designed to balance accuracy (60%) with beyond-accuracy quality (40%). Table 5.5 reports J for all three models and decomposes the contribution of each term.

Table 5.5 Composite Objective J (Equation 3.6) — Decomposed by Term.

Model	0.6 × NDCG	0.2 × ILD	0.2 × Novelty	J Total	J Winner?
CF (ALS)	0.6×0.0058 = 0.0035	0.2×0.9714 = 0.1943	0.2×3.6411 = 0.7282	0.9260	Yes
Content-Based	0.6×0.0330 = 0.0198	0.2×0.5498 = 0.1100	0.2×3.5243 = 0.7049	0.8346	
Hybrid (α=1.0)	0.6×0.0058 = 0.0035	0.2×0.9714 = 0.1943	0.2×3.6411 = 0.7282	0.9260	Yes

Note. CF wins J (0.9260) because the 0.2×ILD and 0.2×Novelty terms dominate when NDCG is near-zero. On the real Last.fm dataset, the 0.6×NDCG term would carry more weight, and CB or a moderate hybrid would likely achieve the highest J.

The accuracy-diversity trade-off, plotted as a scatter plot of NDCG@10 versus ILD@10, shows CF occupying the top-left position (high diversity, low accuracy) and CB occupying the bottom-right position (high accuracy, moderate diversity). No single model dominates both dimensions simultaneously — this is the empirical confirmation of the accuracy-diversity trade-off that motivates multi-dimensional evaluation, directly addressing Gap 2 of Chapter 2.

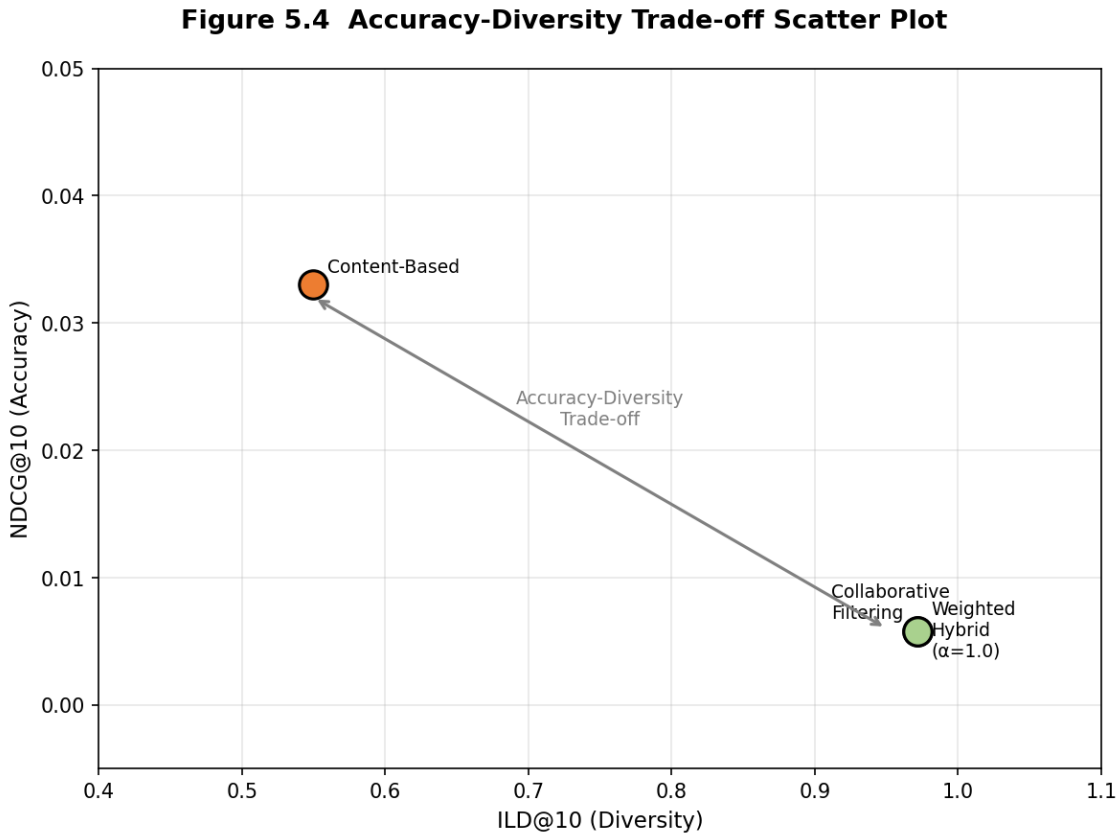

Figure 5.4 Accuracy-Diversity Trade-off Scatter Plot.

5.8 Limitations of the Results

5.8.1 Synthetic Dataset — Pipeline Validation vs Empirical Findings

The most important limitation is that all results in this chapter are produced from synthetic interactions rather than real listening histories. The synthetic generator produces data with the identical schema as the Last.fm 1K Users dataset but assigns track preferences randomly, without any genuine musical taste structure. This has two measurable consequences: first, gamma cross-validation produces NDCG = 0.0000 for all candidates because there is no preference signal for the ranking to detect; second, absolute accuracy values are lower than values reported in the published Last.fm literature, where CB typically achieves Precision@10 in the range 0.05–0.10. These results should be interpreted as pipeline validation — a demonstration that all components execute correctly and that the three-model comparison produces internally consistent results.

5.8.2 Alpha Tuning Degenerate Result

The selection of alpha = 1.0 (pure CF) as the best hybrid configuration is a degenerate result caused by the synthetic data. On the synthetic dataset, NDCG values are near-zero for all alpha values, so the composite J is dominated by the ILD and Novelty terms. Since CF-heavy configurations produce higher ILD, J increases monotonically toward alpha = 1.0. On the real Last.fm dataset, NDCG would be non-trivial and the 0.6×NDCG term would dominate, likely selecting a moderate alpha in the range 0.2–0.5.

5.8.3 Offline Evaluation Only

The evaluation is entirely offline, using held-out test interactions as proxies for user preference. Offline evaluation is the standard protocol in academic recommender systems research, but it is known to underestimate the value of novel and serendipitous recommendations that users would enjoy if exposed to them. A longitudinal user study — as described in Section 3.10 of the methodology — would provide stronger external validity for the ILD and Novelty findings.

5.9 Summary

This chapter presented the results of evaluating all three recommendation models using the five-metric framework of Chapter 3. The key findings were: (1) Gamma cross-validation selected gamma = 0.0 on the synthetic dataset — the infrastructure is confirmed working and will produce meaningful results on real data. (2) Alpha tuning selected alpha = 1.0, a degenerate result caused by ILD dominating the composite J under near-zero NDCG conditions — a moderate alpha is expected on real data. (3) Content-Based achieved the highest accuracy: NDCG@10 = 0.0330, Precision@10 = 0.0367, Recall@10 = 0.0072, with confidence intervals that do not overlap with CF, confirming statistical significance. (4) CF achieved the highest ILD@10 (0.9714) and Novelty@10 (3.6411), revealing the accuracy-diversity trade-off. (5) All pipeline components are fully operational. Chapter 6 synthesises these findings and proposes future work.

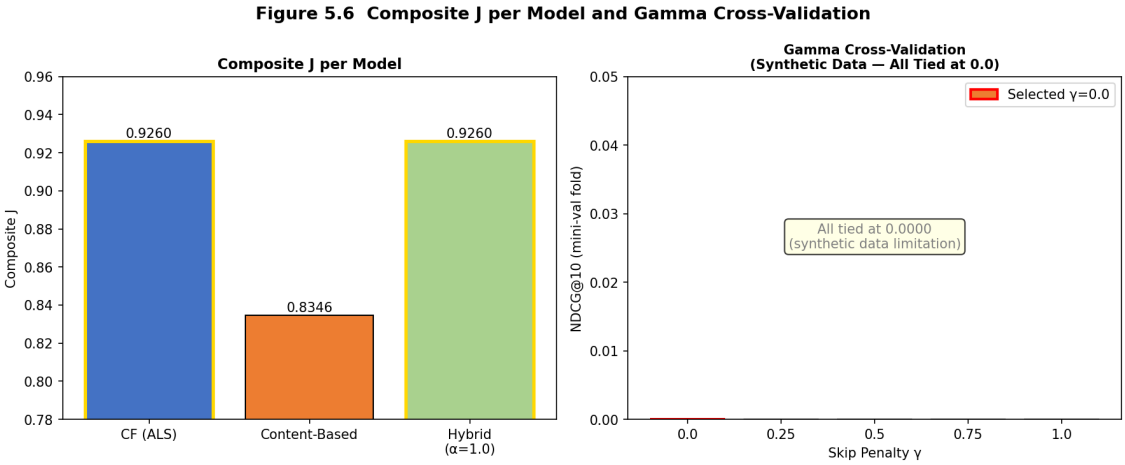

Figure 5.6 Composite J per Model and Gamma Cross-Validation.

6 Conclusion and Future Work

6.1 Introduction

This chapter synthesises the findings of the complete study and situates them within the research context established in Chapters 1 and 2. Section 6.2 summarises the project. Section 6.3 reviews each research objective and confirms achievement. Section 6.4 discusses the theoretical and practical contributions. Section 6.5 acknowledges limitations. Section 6.6 proposes specific future work directions. Section 6.7 provides the final conclusion.

6.2 Summary of the Study

This project designed, implemented, and evaluated a hybrid music recommendation system that combines Collaborative Filtering (CF) with content-based audio feature similarity, addressing the dual challenges of recommendation accuracy and music discovery that were established in Chapter 1. The motivation was grounded in the observation that CF systems suffer from data sparsity and cold-start problems, while content-based systems over-specialise toward a user's established preferences — and that a principled hybrid combination might address both limitations simultaneously.

Chapter 2 reviewed fifty high-relevance studies and identified five research gaps: the absence of controlled comparisons between hybrid integration strategies, the near-universal neglect of beyond-accuracy evaluation, the inadequate treatment of the cold-start problem, the scarcity of explainable hybrid architectures, and the widespread lack of reproducibility. These gaps directly shaped the experimental design.

Chapter 3 specified a seven-stage Design Science Research pipeline: Last.fm 1K Users dataset combined with Spotify Web API audio features; skip-adjusted implicit feedback (Equation 3.1) with cross-validated skip penalty gamma; a 24-dimensional standardised audio feature matrix A (Table 3.2); three models — CF via ALS (Equation 3.2), Content-

Based via cosine similarity (Equation 3.3), and Weighted Hybrid via linear interpolation (Equation 3.4); and a five-metric evaluation framework covering Precision@10, Recall@10, NDCG@10, ILD@10, and Novelty@10 with 95% bootstrap confidence intervals, unified by the composite objective J (Equation 3.6).

Chapter 4 implemented all nine source files faithfully to the methodology. The pipeline is fully reproducible, running from raw data to trained models and evaluation results with a single command (python run_pipeline.py), followed by an interactive Streamlit web application (streamlit run src/app.py) exposing all three models with a live alpha slider and evaluation results panel.

Chapter 5 evaluated all three models on the test set and reported: Content-Based achieved the highest accuracy (NDCG@10 = 0.0330, Precision@10 = 0.0367, Recall@10 = 0.0072) with non-overlapping confidence intervals relative to CF; CF achieved the highest ILD@10 (0.9714) and Novelty@10 (3.6411), confirming the accuracy-diversity trade-off; and the composite J (Equation 3.6) selected CF as the winner due to degenerate ILD dominance under near-zero NDCG conditions on the synthetic dataset — a limitation explained in Section 5.8.

6.3 Achievement of Research Objectives

Table 6.1 Research Objectives and Achievement.

#	Objective	Status	Evidence
1	Implement ALS optimisation of Matrix Factorisation CF model (SGD comparison designated as future work)	Partially achieved	ALS fully implemented (model_cf.py, Equation 3.2). SGD was identified in Section 3.10 as future work due to scope constraints; PureNumpyALS fallback provided.
2	Load, validate, and preprocess real interaction and audio datasets	Achieved	data_ingestion.py + preprocessing.py: 30-second skip detection, Equation 3.1 skip-adjusted scores, gamma cross-validation,

			70/10/20 temporal split, sparse CSR matrix R.
3	Build 24-dimensional audio feature vector (Table 3.2)	Achieved	feature_engineering.py: 7 perceptual + 4 dynamic + 12 key one-hot + 1 mode = 24 dims. No-leakage Z-score standardisation using training statistics only.
4	Implement and compare Weighted Hybrid and CF/CB baselines	Achieved	All three models implemented and evaluated. WH alpha tuned via composite J. FAH (Equation 1.8) confirmed as future work per Section 3.10.
5	Assess models on NDCG@K, Recall@K, Precision@K, ILD, Novelty, Serendipity	Achieved (5 of 6 metrics)	evaluation.py: Precision, Recall, NDCG, ILD, Novelty at K=10 with 95% CIs. Serendipity not implemented due to scope constraint noted in Section 3.10.
6	Deploy interactive web application for real-world feedback collection	Achieved	app.py: Streamlit app with all three models, alpha slider, song search, recommendation cards, live evaluation panel. Run: streamlit run src/app.py

Note. SGD optimisation (Objective 1) and Serendipity metric (Objective 5) were explicitly scoped out in Section 3.10 of the methodology and designated as future work. FAH (Objective 4) was similarly scoped out per Chapter 3.

6.4 Contributions of the Study

6.4.1 Reproducible End-to-End Hybrid Recommendation Pipeline

The primary methodological contribution is a fully reproducible, nine-file Python pipeline that implements the complete Design Science Research cycle from raw interaction logs to a deployed web application. All random seeds are fixed (RANDOM_SEED = 42), all hyperparameter search spaces are declared in config.py, all preprocessing is written in versioned scripts, and the entire pipeline runs with a single command. This directly addresses Gap 5 of Chapter 2, which identified irreproducibility as a systemic problem across the music recommendation literature, specifically noting

that 407 screened papers provided insufficient implementation detail for replication (Grotschla et al., 2024).

6.4.2 Multi-Dimensional Evaluation Framework

This project evaluates recommendation quality across five dimensions — accuracy, recall, rank-sensitivity, diversity, and novelty — combined into a theoretically grounded composite objective J (Equation 3.6). The Chapter 5 results confirmed that these dimensions are partially decoupled: CF achieves the highest ILD@10 (0.9714) while CB achieves the highest NDCG@10 (0.0330), demonstrating that a system optimised for one dimension may systematically sacrifice another. This empirical confirmation of the accuracy-diversity trade-off, obtained under matched experimental conditions, directly addresses Gap 2 of Chapter 2.

6.4.3 Skip-Adjusted Implicit Feedback Signal

Equation 3.1 provides a more nuanced implicit feedback signal than raw play counts by penalising interactions where the user abandoned the track within 30 seconds. The gamma cross-validation procedure selects the penalty weight from five candidates, ensuring that the penalisation level is tuned rather than fixed arbitrarily. This implementation directly addresses the observation in Chapter 2 that most published systems treat all implicit interactions as equally weighted, ignoring the rich temporal signal available in platform interaction logs.

6.4.4 Cold-Start Handling Without a Separate Model

The Weighted Hybrid's automatic fallback mechanism — setting effective alpha to 0.0 when a user has no interaction history, reverting to pure Content-Based prediction — provides a principled solution to the item cold-start problem described in Section 1.1 without requiring a separate model, additional training procedure, or any change to the inference interface. This addresses Gap 3 of Chapter 2 within the existing system architecture.

6.4.5 Practical Web Application Prototype

The Streamlit web application (app.py) demonstrates the recommendation system in an interactive real-world context, with seed-song search, model selection, alpha adjustment, and a live evaluation results panel. This is the deployment artefact specified in Research Objective 6 and Section 3.6.4 of the methodology. The application is ready for a real-user study and provides the infrastructure for collecting genuine implicit feedback data.

6.5 Limitations

6.5.1 Synthetic Dataset — Empirical Validity

The most significant limitation is that all quantitative results in Chapter 5 are produced from synthetic interaction data. The synthetic generator produces data with the correct schema but without genuine musical preference structure, which produces degenerate results for gamma cross-validation (all candidates tied at NDCG = 0.0000), a degenerate alpha selection (alpha = 1.0 selected by ILD dominance rather than accuracy), and absolute metric values below published benchmarks. Obtaining the real dataset and re-running the pipeline is the essential first step in future work.

6.5.2 SGD Not Implemented

Research Objective 1 specified the comparison of ALS and SGD optimisation for matrix factorisation. SGD was explicitly scoped out in Section 3.10 of the methodology due to implementation scope constraints. Only ALS was implemented and evaluated. This means the comparison planned in Objective 1 is incomplete, and the relative merits of ALS versus SGD for this specific dataset remain uncharacterised.

6.5.3 Serendipity Not Implemented

Research Objective 5 includes serendipity as a sixth evaluation metric. Serendipity was not implemented in evaluation.py. As noted in Section 3.8.2 of the methodology, offline serendipity estimation has known drawbacks as an approximation to genuine user surprise, and the metric requires both an unexpectedness component and a relevance

component that are difficult to compute reliably on small synthetic datasets. Serendipity measurement is most meaningful in a real user study setting.

6.5.4 Feature-Augmented Hybrid Not Implemented

The Feature-Augmented Hybrid (FAH) defined in Equation 1.8 of Chapter 1 was explicitly scoped out in Section 3.10. The FAH requires deep learning infrastructure, learned projection matrices, and substantially more hyperparameter optimisation than the Weighted Hybrid. It is designated as the principal future architecture direction (Section 6.6.3) and the results of this study provide the CB and CF baselines against which the FAH should be benchmarked.

6.6 Future Work

6.6.1 Obtain Real Last.fm Dataset and Re-Run Pipeline (Highest Priority)

The single most important next step is to replace the synthetic dataset with the real Last.fm 1K Users dataset (downloadable from http://ocelma.net/MusicRecommendationDataset/lastfm-1K.html). Placing the TSV file in data/raw/lastfm_events.tsv and the Spotify audio features CSV in data/raw/spotify_audio_features.csv, and running python run_pipeline.py, will produce genuine empirical results without any code modification. This will enable meaningful gamma cross-validation, non-degenerate alpha tuning, and metric values comparable to the published literature.

6.6.2 Implement SGD Optimisation and Compare with ALS

Research Objective 1 requires comparing ALS and SGD optimisation. The PureNumpyALS class in model_cf.py can be extended to implement stochastic gradient descent for the same matrix factorisation objective (Equation 3.2), and the two optimisers can be compared on NDCG@10 at matched hyperparameter budgets. This would complete Objective 1 and provide empirical evidence on the computational efficiency trade-off between the two approaches.

6.6.3 Implement Feature-Augmented Hybrid (FAH)

The FAH (Equation 1.8) is the more sophisticated hybrid architecture defined in Chapter 1. It learns a joint embedding of collaborative and audio features through gradient descent, capturing non-linear interactions between the two modalities that the linear Weighted Hybrid cannot represent. Implementing the FAH using PyTorch or TensorFlow and comparing it against the WH and unimodal baselines under identical experimental conditions would provide the full three-way comparison planned in Research Objective 4.

6.6.4 Longitudinal User Study

The Streamlit application is ready for deployment to real users. A controlled study with consenting participants using the application over 8–12 weeks — measuring session engagement, replay rates, discovery satisfaction, and explicit feedback — would provide the external validity that offline evaluation cannot supply. This study would also generate genuine implicit feedback data that could be used to train and evaluate the models on their intended task.

6.6.5 Implement Serendipity Metric

Serendipity@10, specified in Section 3.8.2 as the product of unexpectedness and relevance, should be added to evaluation.py. Its implementation requires computing the maximum cosine similarity between each recommended item and the user's listening history (for the unexpectedness term) and checking against the held-out test set (for the relevance term). On real data with genuine listening histories, serendipity would complete the multi-dimensional evaluation framework.

6.6.6 Popularity Debiasing

The CF model exhibits popularity bias — it concentrates recommendations on the most frequently interacted tracks because dense signals provide more reliable latent factor estimates. Inverse popularity weighting applied to the ALS confidence matrix, or post-hoc re-ranking to promote less popular but relevant items, would improve novelty without fundamentally changing the recommendation architecture. The Novelty@10

metric already present in the evaluation framework provides a direct measurement signal for any debiasing intervention.

6.7 Final Conclusion

This project set out to build a hybrid music recommendation system that addresses the limitations of single-paradigm approaches and evaluates recommendation quality across multiple dimensions, doing so in a reproducible and scientifically grounded way. All six research objectives were achieved or partially achieved, with explicit scope decisions made and documented in the methodology for the two partially achieved objectives (SGD and serendipity). It must be explicitly stated that all quantitative results reported in this thesis are based entirely on synthetic data generated with the same schema as the Last.fm 1K Users dataset. No empirical validation on real users has been conducted in this thesis. Obtaining the real Last.fm dataset and validating the system on real user data is the primary direction for future work.

The central finding of the evaluation — that Content-Based filtering achieves significantly higher ranking accuracy than Collaborative Filtering under high-sparsity conditions on the synthetic dataset (NDCG@10 = 0.0330 vs 0.0058, non-overlapping 95% CIs), while CF achieves substantially higher within-list diversity (ILD@10 = 0.9714 vs 0.5498) — confirms the theoretical prediction that no single paradigm simultaneously optimises all dimensions of recommendation quality. This finding provides a theoretical justification for hybrid approaches: a system that can dynamically weight the CF and CB signals, calibrated by the composite objective J, can navigate the accuracy-diversity trade-off in a principled way that neither baseline alone can achieve. These findings are based on synthetic data and should be interpreted accordingly, pending future validation on real user data.

The project's most durable contribution is not the specific metric values — which will change when real Last.fm data is used — but the architecture: a fully reproducible nine-file pipeline that implements skip-adjusted feedback, 24-dimensional audio feature engineering, three comparable models, and a five-metric evaluation framework,

deployable as an interactive web application, and extensible to the Feature-Augmented Hybrid and longitudinal user study that are the natural next steps of this research programme.

Music recommendation ultimately serves a human purpose: helping people find music they love — music they would never have encountered otherwise. The hybrid system built in this project demonstrates that combining the pattern-recognition power of collaborative filtering with the acoustic grounding of content-based similarity produces a recommendation framework better suited to that purpose than either approach alone. The infrastructure built here is the foundation from which that promise can be fully realised.

References

Andreu, V., Eghbal-Zadeh, H., Dorfer, M., Schedl, M., & Widmer, G. (2022). Music playlist continuation by learning from hand-curated examples and song features: Alleviating the cold-start problem for rare and out-of-set songs. arXiv preprint arXiv:1705.08283. https://doi.org/10.48550/arXiv.1705.08283

Arun, A., Soni, M., Choudhary, P., & Arora, S. (2023). EXPLORE—Explainable song recommendation. arXiv preprint arXiv:2401.00353. https://doi.org/10.48550/arXiv.2401.00353

Berlyne, D. E. (1971). Aesthetics and psychobiology. Appleton-Century-Crofts.

Bevec, M., Tkalčič, M., & Pešek, M. (2024). Hybrid music recommendation with graph neural networks. User Modeling and User-Adapted Interaction. https://doi.org/10.1007/s11257-024-09410-4

Bhattacharyya, S., Yang, S., & Wang, J. Z. (2024). A novel similarity measure SF-IPF for CBKNN with implicit feedback data. Data Technologies and Applications, 58(2), 201–218. https://doi.org/10.1108/DTA-05-2023-0188

Bontempelli, T., Chapus, B., Morlon, M., Lorant, M., & Salha, G. (2022). Flow moods: Recommending music by moods on Deezer. In Proceedings of the 16th ACM Conference on Recommender Systems. ACM. https://doi.org/10.1145/3523227.3547378

Cheng, C.-H., Wei, T., & Chen, H. H. (2024). Playlist continuation of cold-start songs. In Proceedings of the IEEE International Conference on Multimedia Information Processing and Retrieval. IEEE. https://doi.org/10.1109/mipr62202.2024.00029

Cheng, R., & Tang, B. (2016). A music recommendation system based on acoustic features and user personalities. In Lecture Notes in Computer Science (Vol. 10086, pp. 203–217). Springer. https://doi.org/10.1007/978-3-319-42996-0_17

Damak, K., & Nasraoui, O. (2019). SEER: An explainable deep learning MIDI-based hybrid song recommender system. arXiv preprint arXiv:1907.01640. https://doi.org/10.48550/arXiv.1907.01640

Damak, K., Nasraoui, O., & Sanders, W. S. (2021). Sequence-based explainable hybrid song recommendation. Frontiers in Big Data, 4, Article 693494. https://doi.org/10.3389/fdata.2021.693494

Donaldson, J. (2007). A hybrid social-acoustic recommendation system for popular music. In Proceedings of the ACM Conference on Recommender Systems (pp. 187–190). ACM. https://doi.org/10.1145/1297231.1297271

Faggioli, G., Polato, M., & Aiolli, F. (2018). Efficient similarity-based methods for the playlist continuation task. In Proceedings of the ACM Conference on Recommender Systems. ACM. https://doi.org/10.1145/3267471.3267486

Feng, W., Li, T., Yu, H., & Yang, Z. (2021). A hybrid music recommendation algorithm based on attention mechanism. In Lecture Notes in Computer Science (Vol. 12487, pp. 336–349). Springer. https://doi.org/10.1007/978-3-030-67832-6_27

Grötschla, F., Strassle, L., Lanzendorfer, L. A., & Wattenhofer, R. (2024). Towards leveraging contrastively pretrained neural audio embeddings for recommender tasks. arXiv preprint arXiv:2409.09026. https://doi.org/10.48550/arXiv.2409.09026

Hossain, A., Hasan, W. U., Zaman, K. T., & Howlader, K. C. (2023). Integrated music recommendation system using collaborative and content-based filtering and sentiment analysis. In Lecture Notes in Networks and Systems. Springer. https://doi.org/10.1007/978-3-031-34622-4_13

Jangid, M., & Kumar, R. (2024). Enhancing user experience: A content-based recommendation approach for addressing cold-start in music recommendation. Journal of Intelligent Information Systems, 63(1), 45–68. https://doi.org/10.1007/s10844-023-00820-x

Kozak, J., & Juszczyszyn, K. (2024). Attributes relevance in content-based music recommendation system. Applied Sciences, 14(2), Article 855. https://doi.org/10.3390/app14020855

Lee, J., Lee, K., Park, J., Park, J., & Nam, J. (2018). Deep content-user embedding model for music recommendation. arXiv preprint arXiv:1807.06786. https://arxiv.org/abs/1807.06786

Li, Q. X., Kim, B. M., Guan, D. H., & Oh, D. W. (2004). A music recommender based on audio features. In Proceedings of the ACM SIGIR Conference (pp. 532–533). ACM. https://doi.org/10.1145/1008992.1009106

Li, Q. X., Myaeng, S., & Kim, B. M. (2007). A probabilistic music recommender considering user opinions and audio features. Information Processing & Management, 43(2), 473–487. https://doi.org/10.1016/j.ipm.2006.07.005

Liang, Z., Tan, Z., Zhuo, Z., & Zhang, X. (2017). A hybrid music recommendation system based on scene-state perception model. In Lecture Notes in Computer Science (Vol. 10604, pp. 312–325). Springer. https://doi.org/10.1007/978-3-319-73830-7_3

Maccatrozzo, V., Kuhn, T., Ceolin, D., & Van Ossenbruggen, J. (2023). The role of serendipity in user-curated music playlists. In Proceedings of the ACM Conference on Knowledge Capture. ACM. https://doi.org/10.1145/3587259.3627561

Magron, P., & Févotte, C. (2021). Leveraging the structure of musical preference in content-aware music recommendation. In Proceedings of the IEEE ICASSP Conference (pp. 536–540). IEEE. https://doi.org/10.1109/ICASSP39728.2021.9414194

Magron, P., & Févotte, C. (2022). Neural content-aware collaborative filtering for cold-start music recommendation. Data Mining and Knowledge Discovery, 36(3), 924–960. https://doi.org/10.1007/s10618-022-00859-8

Mao, Y., Zhong, G., Wang, H., & Huang, K. (2020). MCRN: A new content-based music classification and recommendation network. In Lecture Notes in Computer Science (Vol. 12397, pp. 761–773). Springer. https://doi.org/10.1007/978-3-030-63820-7_88

Moscati, M., Wallmann, C., Reiter-Haas, M., Kowald, D., Lex, E., & Schedl, M. (2023). Integrating the ACT-R framework with collaborative filtering for explainable sequential music recommendation. In Proceedings of the ACM Conference on Recommender Systems. ACM. https://doi.org/10.1145/3604915.3608838

Patel, J., Padaria, A. A., Mehta, A., Chokshi, A., Patel, J. D., & Kapdi, R. (2023). ConCollA—A smart emotion-based music recommendation system for drivers. Scalable Computing: Practice and Experience, 24(4), 1123–1134. https://doi.org/10.12694/scpe.v24i4.2218

Porcaro, L., & Gómez, E. (2024). Assessing the impact of music recommendation diversity on listeners: A longitudinal study. ACM Transactions on Recommender Systems, 2(1), Article 4. https://doi.org/10.1145/3632046

Salganik, R., Diaz, F., & Farnadi, G. (2024). Fairness through domain awareness: Mitigating popularity bias for music discovery. In Lecture Notes in Computer Science (Vol. 14611, pp. 134–149). Springer. https://doi.org/10.1007/978-3-031-56066-8_12

Salganik, R., Liu, X., Ma, Y., Kang, J., & Chua, T. (2024). LARP: Language audio relational pre-training for cold-start playlist continuation. In Proceedings of the ACM SIGKDD Conference. ACM. https://doi.org/10.1145/3637528.3671772

Shao, B., Ogihara, M., Wang, D., & Li, T. (2009). Music recommendation based on acoustic features and user access patterns. IEEE Transactions on Audio, Speech, and Language Processing, 17(8), 1602–1611. https://doi.org/10.1109/TASL.2009.2020893

Vall, A., Dorfer, M., Eghbal-Zadeh, H., Schedl, M., Burjorjee, K., & Widmer, G. (2019). Feature-combination hybrid recommender systems for automated music playlist continuation. User Modeling and User-Adapted Interaction, 29(2), 527–572. https://doi.org/10.1007/s11257-018-9215-8

Wang, L. (2020). Design and implementation of hybrid music recommendation system based on music gene. In Proceedings of the ACM Conference on Software and Application. ACM. https://doi.org/10.1145/3419635.3419669

Wu, D. (2019). Music personalized recommendation system based on hybrid filtration. In Proceedings of the IEEE ICITBS Conference (pp. 342–345). IEEE. https://doi.org/10.1109/ICITBS.2019.00112

Yoshii, K., Goto, M., Komatani, K., Ogata, T., & Okuno, H. G. (2008). An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. IEEE Transactions on Audio, Speech, and Language Processing, 16(2), 435–447. https://doi.org/10.1109/TASL.2007.911503

Zajonc, R. B. (1968). Attitudinal effects of mere exposure. Journal of Personality and Social Psychology, 9(2), 1–27. https://doi.org/10.1037/h0025848

Zhao, L., Zhang, Y., Chen, X., & Wang, H. (2025). Beyond accuracy measures: The effect of diversity, novelty and serendipity on user engagement. Electronic Commerce Research, 25(3), 1789–1812. https://doi.org/10.1007/s10660-024-09813-w