APRIL 2024

PRESENTATION

MUSIC RECOMMENDATION SYSTEM

TABLE OF CONTENT

KEY TAKEAWAYS

The objective was to develop a music recommendation system to recommend relevant songs to users in order to improve user satisfaction and retention.

ALGORITHMS

Considered collaborative filtering techniques, including user-user, item-item, SVD, co-clustering models, along with content-based and popularity recommendations, to create a recommendation system.

KEY FINDINGS & INSIGHTS

Identified the SVD model as the most effective, achieving high precision, recall, and F1 score, as well low RMSE.

RECOMMENDATION

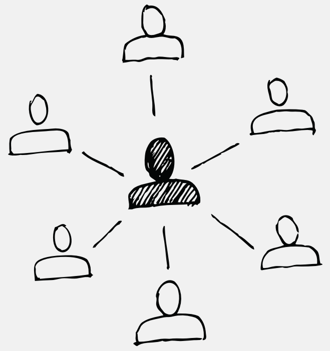
Recommended to implement the SVD model to optimize the music recommendation system and consider hybrid model to further improve to outcomes.

OVERVIEW OF THE PROBLEM

In today's competitive digital landscape, where user engagement is vital for business success, personalized recommendations have become indispensable. Platforms like Spotify must differentiate themselves by accurately predicting user preferences and suggesting relevant songs.

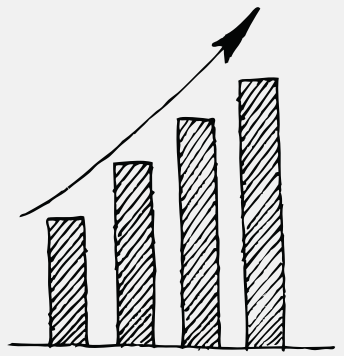
GOAL

USER RETENTION & SATISFACTION



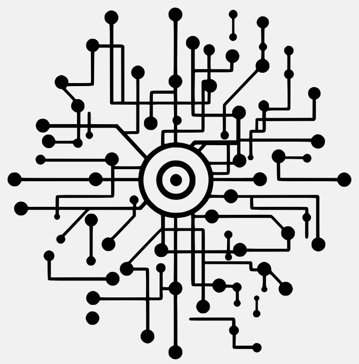
AD
&
SUBSCRIPTION FEE

REVENUE GROWTH



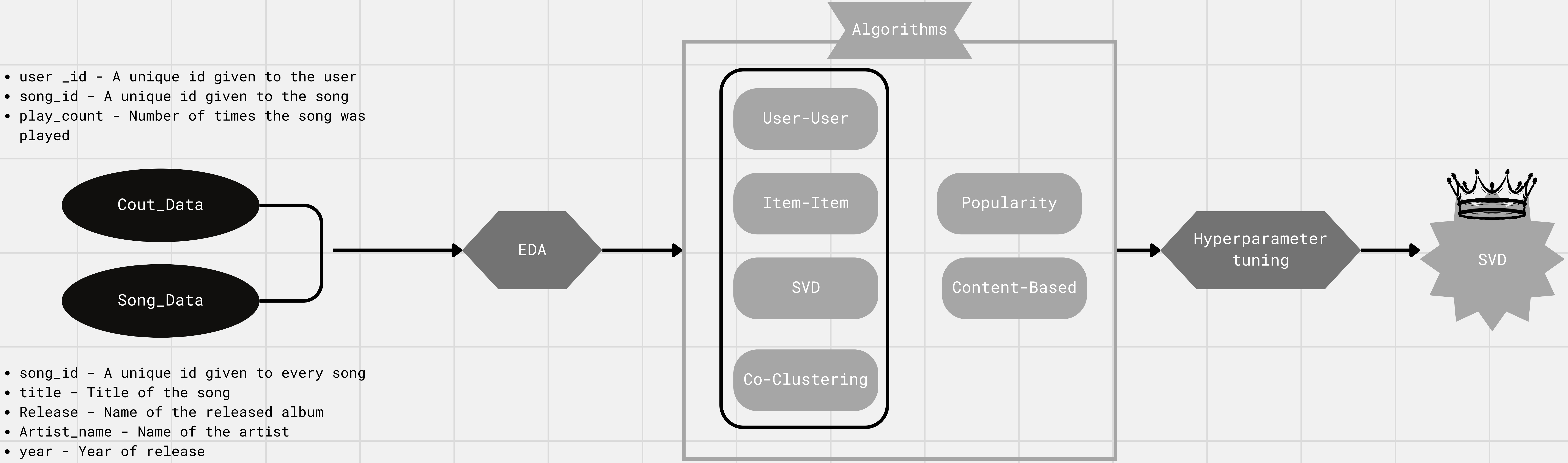
CHALLENGE

CURATE SONG RECOMMENDATIONS TAILORED TO EACH USER'S TASTES



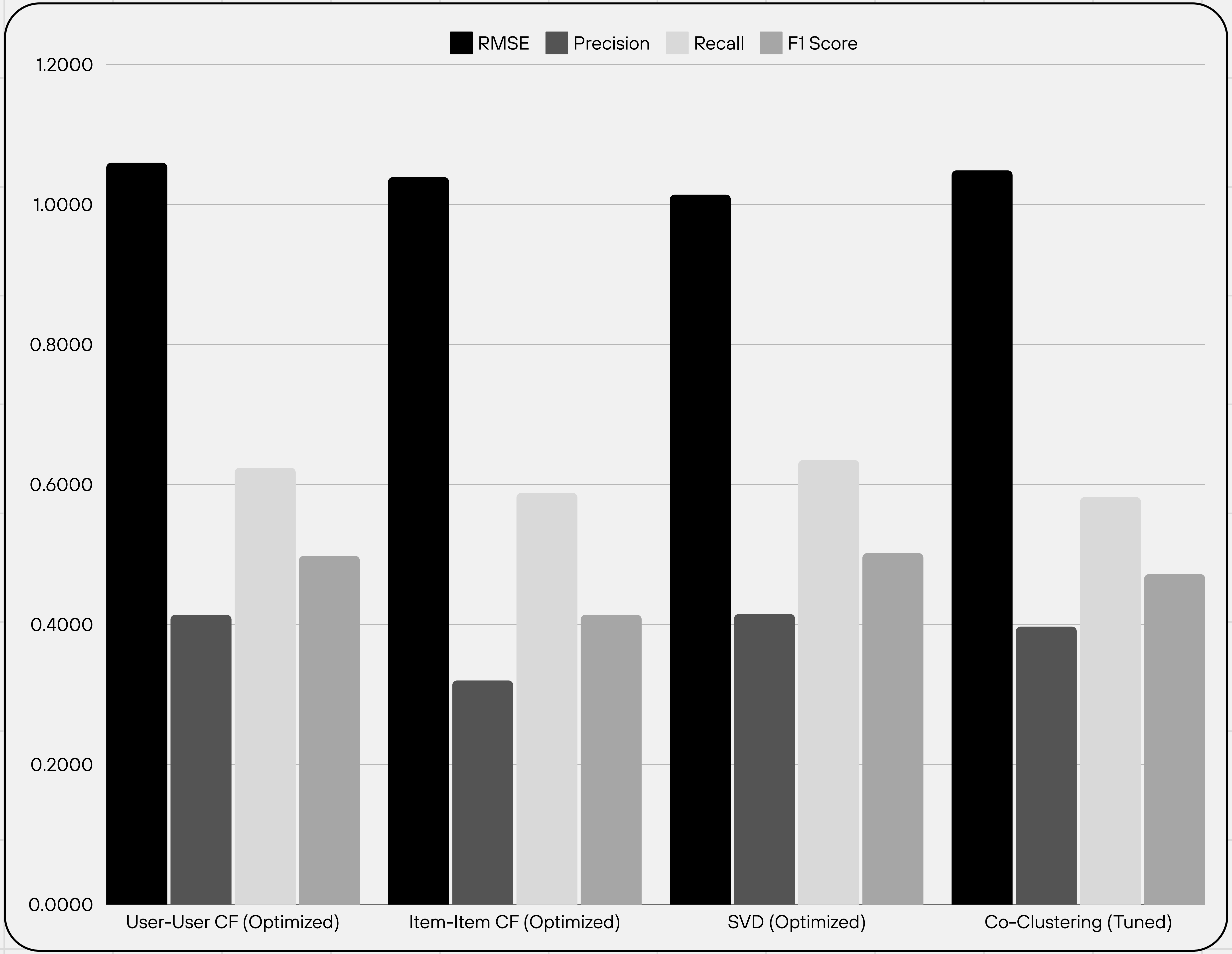
INCREASE USER RETENTION & SATISFACTION

METHODOLOGY

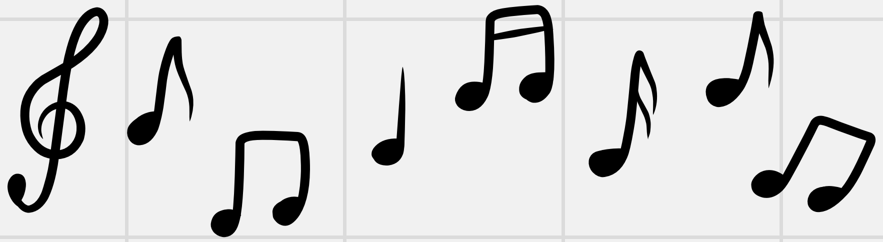


KEY FINDINGS & INSIGHTS

The SVD model consistently outperformed its counterparts in terms of precision, recall, F1 score, as well as RMSE, indicating the model's ability to accurately predict user preferences and recommend relevant songs tailored to individual tastes, as well as the accuracy in predicting play counts for songs by users.



SOLUTION FOR COLD START PROBLEM



Popularity-based

Content-based

BENEFITS OF IMPLEMENTING MUSIC RECOMMENDATION SYSTEM

For platforms like Spotify, if users are more satisfied and more likely to subscribe or be active on the music platform if the songs recommended to them are relevant and matches their tastes.

For platforms like Spotify, the 2 main contributors to the revenue are

1. ad-supported revenue
2. premium subscription fee

A good recommendation system can foster the connection between creators and users, allowing less known creators to gain exposure to their audience.

RECOMMENDATION & NEXT STEPS

- A/B TESTING
- HYPER-PARAMETER TUNING
- HYBRID MODELS
- DIVERSE DATA SOURCE
- USER FEEDBACK

Q&A

	User-User Collaborative Filtering	Item-Item Collaborative Filtering	Co-Clustering	Singular Value Decomposition (SVD)	Popularity-Based	Content-Based Filtering
Logic	Recommends items by finding similar users. If user A has similar taste to user B, suggests items liked by B to A and vice versa.	Recommends items similar to those a user has liked before, based on similarities among items calculated from user ratings.	Groups users and items into co-clusters based on their rating patterns.	Factorizes the rating matrix into lower-dimensional matrices, capturing latent factors associated with users and items.	Recommends items based on their popularity or rating across the user base, disregarding user preferences.	Recommends items similar to those a user has liked before, based on content features of the items.
Strengths	• Personalization, can recommend new or unpopular items.	• More scalable than user-user, as items often less than users. • Stability: Item profiles change less frequently.	• Efficiency: Reduces dimensionality, more scalable. • Accuracy: Improves recommendation by capturing deeper patterns.	• Accuracy: Provides high accuracy by capturing deep patterns. • Reduces dimensionality, helps with sparse data.	• Simplicity: Easy to implement. • Effective for new users without personalized data.	• Personalization: Tailored to user's past preferences. • Can recommend new items based on content.
Limitations	• Scalability: Computationally expensive with large user base. • Cold start: Struggles with new users.	• Cold start for items: Hard to recommend new items. • Sparsity: Requires sufficient ratings.	• Complexity: More complex to implement and fine-tune. • Cold start: Struggles with new users and items.	• Interpretability: Latent factors not always clear. • Cold start: Doesn't handle new users/items without ratings.	• Lack of personalization. • May ignore niche items.	• Limited scope: Relies on available item features. • Over-specialization: May recommend too similar items.