Silver Screen — Content Discovery & Recommendation Pipeline

Type: Product and Data Case Study
Context: Streaming Discovery and Personalization
Tools: Python, pandas, scikit-learn
Author: Shashwat Chauhan
Institution: BITS Pilani, Goa Campus
Date: December 2025

---

1. Product Context

Large streaming platforms face a recurring discovery challenge. Users often arrive with limited intent, encounter expansive catalogs, and exit sessions before making a confident viewing choice.

From a product standpoint, recommendation systems function as decision-support systems. Their role is to reduce friction in choice-making while maintaining relevance and trust.

Primary product goals: - Reduce time to first play - Increase relevance per scroll - Lower cognitive load during browsing - Improve long-term engagement and retention

This case study explores how collaborative and content-based recommendation strategies translate into practical discovery experiences for the Silver Screen streaming platform.

---

2. Problem Definition

Core product question:
How can Silver Screen help users reach a confident viewing decision faster, with less effort, across a large and diverse content catalog?

Supporting questions: - How effectively do similarity-based models reflect real viewing behavior - Where do recommendation strategies succeed or break down across content types - How should algorithmic decisions influence discovery interface design

The objective is product impact on discovery quality rather than standalone algorithmic performance.

---

3. Data as a Model of Viewing Behavior

To approximate streaming choice behavior, a large-scale user-item interaction dataset was used as a proxy for real viewing patterns.

Dataset summary after consolidation: - Approximately 1.03 million interactions - Around 219,000 unique titles - Primary markets: United States, Canada, Mexico

Captured signals: - Strength of user preferences - Co-consumption patterns - Implicit similarity between titles

These signals generalize directly to discovery behavior on Silver Screen.

---

4. Data Ingestion

```
import pandas as pd

ratings = pd.read_csv('user_ratings.csv')
users = pd.read_csv('users.csv')
items = pd.read_csv('titles.csv')
```

---

5. Product-Driven Data Preparation

The preprocessing objective was to construct a high-signal interaction table that mirrors realistic discovery behavior.

5.1 Merging Interaction Sources

```
df = ratings.merge(users, on='user_id').merge(items, on='item_id')
```

This step unified user profiles, title metadata, and interaction signals into a single analytical dataset.

5.2 Removing Low-Signal Users and Titles

```
active_users = df.groupby('user_id').size()
active_items = df.groupby('item_id').size()

df = df[
    df['user_id'].isin(active_users[active_users > 200].index) &
```

```
    df['item_id'].isin(active_items[active_items > 50].index)
]
```

Filtering rationale: - Removes sparse, noisy interaction patterns - Improves stability of similarity learning - Reflects users with sustained engagement

5.3 Metadata Normalization

```
df['release_year'] = pd.to_numeric(df['release_year'], errors='coerce')
df = df.dropna(subset=['release_year'])
```

Ensures consistent temporal metadata for analysis and downstream filtering.

---

6. Modeling Strategy

An item-based collaborative filtering approach was selected.

Product rationale: - Scales efficiently to large catalogs - Maps cleanly to similar titles discovery patterns - Performs well on genre clusters and franchises

---

7. Interaction Matrix Construction

```
pivot = df.pivot_table(
    index='item_id',
    columns='user_id',
    values='rating'
).fillna(0)
```

Each row represents a title's interaction footprint across the user base.

---

8. Similarity Model Training

```
from scipy.sparse import csr_matrix
from sklearn.neighbors import NearestNeighbors

sparse_matrix = csr_matrix(pivot.values)

model = NearestNeighbors(
    metric='cosine',
```

```
    algorithm='brute'
)
model.fit(sparse_matrix)
```

This configuration mirrors a production-style nearest-neighbor similarity service.

---

9. Product-Oriented Validation

Evaluation focused on real discovery scenarios rather than abstract metrics.

Scenario 1: Dense, Clustered Content

```
distances, indices = model.kneighbors(
    pivot.loc[['Harry_Potter_Azkaban']],
    n_neighbors=5
)
```

Returned neighbors: Goblet of Fire, Chamber of Secrets, Sorcerer's Stone, Order of the Phoenix

Interpretation: Strong intra-series coherence, high precision for franchise content, well-suited for similar titles rows.

Scenario 2: Sparse, Cross-Genre Content

```
distances, indices = model.kneighbors(
    pivot.loc[['1984']],
    n_neighbors=5
)
```

Returned neighbors: A Civil Action, Foucault's Pendulum, No Safe Place

Interpretation: Partial thematic overlap with reduced coherence, highlighting limits of pure collaborative filtering.

---

10. Market Bias and Geographic Signals

```
df['country'].value_counts(normalize=True)
```

Observed distribution: approximately 75 percent United States, 9 percent Canada, less than 1 percent Mexico.

Product insight: Popularity bias is structurally embedded, indicating the need for market-aware ranking and localization strategies.

---

11. Translating Models into Silver Screen UX

Algorithm to Interface Mapping

- Item-based collaborative filtering supports Similar Titles rows
- Hybrid scoring powers the For You feed
- Popularity signals drive Trending and Popular sections
- Content-based features enable genre and attribute filters

Home Feed Scoring Logic

```
home_feed_score = (
    0.5 * collaborative_score +
    0.3 * popularity_score +
    0.2 * content_score
)
```

This balance maintains relevance while preserving discovery and feed stability.

---

12. Collaborative and Content-Based Tradeoffs

Content-based filtering performs well for cold-start users and offers interpretability, but limits serendipity.

Collaborative filtering captures real viewing behavior and excels on popular clusters, but suffers from cold-start issues and popularity bias.

Production systems require hybrid approaches to balance relevance, discovery, fairness, and stability.

---

13. Key Outcomes

This project demonstrates an end-to-end, product-aligned recommendation pipeline including high-signal data preparation, behavior-aware filtering, item-based modeling, scenario-driven validation, and direct UX translation.

---

14. Final Product Takeaway

Recommendation systems are decision-design systems rather than ranking problems.

This study demonstrates that data quality shapes recommendation quality, algorithm selection shapes interface structure, and product impact depends on alignment with real user decision flows.

The highest leverage lies in clean data, thoughtful filtering, and product-aligned evaluation rather than model complexity.

Project: Silver Screen Recommendation System
Author: Shashwat Chauhan
Institution: BITS Pilani, Goa Campus
Date: December 2025