

WHITEPAPER

Personalizing Content using Foundation Model-Based Recommendation Systems

AUTHOR

Jay Chauhan,
Tech Mahindra

Stuart Ure,
Sky New Zealand

Dr. Anshu Premchand
Tech Mahindra

Scale at Speed TM





Industry data suggests that 49% of viewers are willing to cancel a service because they cannot find relevant content quickly.

Executive Summary

Every second a viewer spends searching for content directly translates into lost engagement. Global consumer data highlights a critical issue: the average viewer now spends 14 minutes searching for something to watch, leading nearly half of subscribers to consider cancellation.¹ Existing two-tower recommendation systems are inadequate because isolated machine learning models struggle to transfer learning to other models, maintain context, and infer immediate user intent. By transitioning to a unified preference layer, platforms can solve discovery friction, reduce churn, and protect subscription revenue without undergoing a high-risk, full-system replacement.

Industry data suggests that 49% of viewers are willing to cancel a service because they cannot find relevant content quickly.² Existing recommendation systems act as mirrors, reflecting a user's past behavior but are incapable of inferring the user's future intent. This paper outlines how foundation models can enhance existing recommendation architectures, enable intent-aware discovery while avoiding a disruptive full-system replacement.



Table of Contents

2	Executive Summary	4	Introduction	5	Why Current Systems Fail at Hyper-Personalization
6	The 'Siloed' Brain Problem	7	The Relevance Plateau	8	Shift to Foundation Models: Building the Unified Brain
8	Beyond History: Predicting Immediate Session Intent	9	Retrieval-Augmented Personalization Pattern	10	Netflix Artwork Personalization
10	Tokenization of User Interactions and Cold-Start Resolution	10	Engineering the Transition: Operationalizing the Unified Brain	11	Data Readiness
11	Risk Mitigation	12	Inference Economics	13	The Bottom Line: Quantifying Agentic ROI and Performance
13	The KPI Success Ladder: Fulfilling the Intent Bull's-Eye	14	Responsible Personalization and Governance	14	Conclusion

Introduction

The environment in which streaming platforms operate is one where content grows exponentially while audience attention remains limited. Even though recommendation systems have made it easier to discover content, many viewers still spend a good deal of time looking for something that matches their taste palette and mood. This is a subscriber engagement challenge for content businesses as well as a problem relating to user experience.

The main issue is the use of conventional recommendation systems, which rely heavily on separate machine learning models and signals from users' past behavior. Although these methods have worked well in industry, they usually struggle to handle changing user intentions and contextual preferences. . So, while they may know what a viewer watched, they have less understanding of what the viewer is likely to want during the current session. Instead of replacing existing systems, foundation models offer a chance to improve them by enhancing understanding of user intent and passing this understanding to existing retrieval and ranking systems, thereby increasing recommendation quality.



1. Why Current Systems Fail at Hyper-Personalization

Most recommendation systems in use today rely on two key elements: fast candidate retrieval implemented via a two-tower architecture and short-term memory models that capture recent user behavior. For years, this architecture has been an industry workhorse because it is fast and scalable.

It works by creating two separate neural networks: a user tower and an item tower. Recommendations are then achieved by finding items whose numerical codes (embeddings) are closest through a dot product or similarity calculation.

1.1 The Limits of the Two-Tower Architecture

The two-tower model is highly useful for rapid candidate generation, where a large content catalog is quickly narrowed into a set of relevant recommendations for each user, but it faces practical limits that contribute to the “14-minute leak” problem, namely:

- **Limited Context Awareness:**
These models are effective in identifying recommendations based on a user's recent watch history but fail to identify the immediate reasoning behind a user choosing a particular content at the moment. They convert complex human taste into numerical similarity scores, which is insufficient to accurately predict a user's intent in a particular watch session.
- **Short-Term Memory Constraint:**
Current systems often struggle to look far beyond a user's most recent watch activity because large histories also increase training, serving, and latency costs. As a result, they often miss learning the long-term preferences that define a user's deeper taste profile.

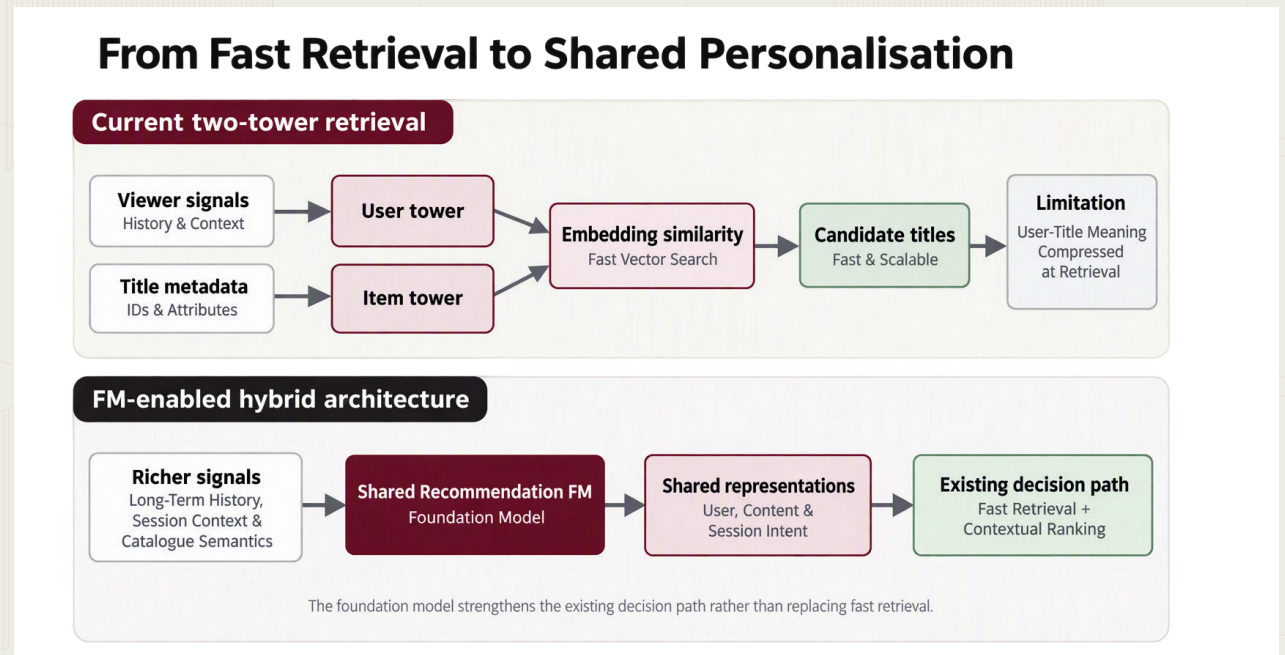


FIGURE 1: Modern FM Enabled architecture vs Legacy Two-Tower retrieval

1.2 The 'Siloed' Brain Problem

While efficient for rapid filtering, the above two-tower architecture suffers from what can be termed the 'siloed brain' problem. Streaming platforms often lack a single reusable preference layer; they rely on multiple separate ML models that are independently trained, maintained, and refreshed. These silos ultimately cause:

- **The Fragmented Reality:** If a user scrolls through any streaming app, each recommendation row often relies on a different ML model. "Trending Now" may use one model, "Because You Watched..." another, and "You May Like" a third.
- **The Communication Gap:** If the ML model behind "You May Like" learns that a user is now a sci-fi enthusiast, the ML model powering "We Think You'll Love These" may not adapt to this new insight for days.
- **The Operational Result:** When updates from one ML model don't seamlessly flow into others, the result is an uncoordinated, resource-heavy, and stale user experience.

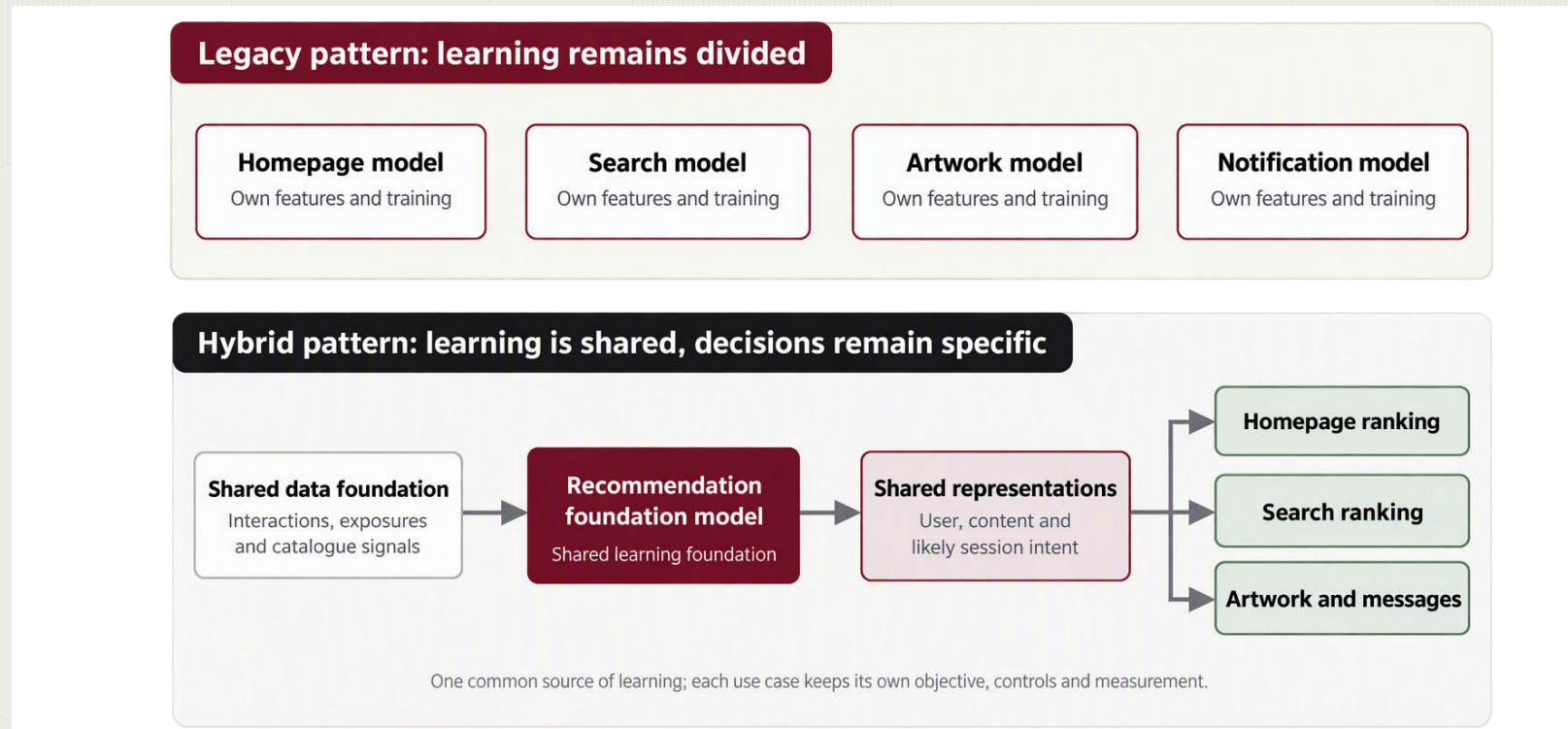


FIGURE 2: Agentic (Unified FM Layer) vs Legacy (Siloed) Architecture

1.3 The Relevance Plateau

Legacy two-tower architectures are good at rapid candidate generation but tend to remain limited in their understanding of the current user session.

Legacy algorithms can become passive when they only wait for the next click and optimize on short-term behavioral signals. On the other hand, if a streaming platform employs simplistic content genre tags like "action" or "drama," it cannot capture nuanced viewing contexts. For example, a late-night movie session, a family co-viewing session, or a quick mobile browse, when combined, can yield irrelevant content recommendations for users.

To plug the "14-minute leak," platforms need to transition toward a unified foundation model (FM)-enabled architecture. While this is not a replacement for the entire stack, it represents a shared personalization foundation that learns from long-term interaction data and catalog semantics. It then distributes those learnings to downstream retrieval, ranking, search, and experience surfaces.



2. Shift to Foundation Models: Building the Unified Brain

The move to a foundation-model architecture involves a shift from using several task-specific ML models to deploying a shared learning foundation. It is to be noted that this is not a move away from fast retrieval and ranking systems as these systems should continue to be used where they work best. Instead, the foundation model acts as a reusable preference layer, learning from a very wide history of user interactions and rich metadata about the content catalog. It then provides viewer, content and intent representations to downstream applications like retrieval, ranking, search and homepage personalization.

This is what we refer to as the “unified brain.” It is not one large model making all content recommendation decisions but instead is a common source of learning that helps different parts of the recommendation experience work with a unified understanding of the viewer.

2.1 Beyond History: Predicting Immediate Session Intent

Legacy ML models act like mirrors, reflecting past behavior rather than anticipating future intent. But in today's market conditions, simple behavioral matching is inadequate, and platforms need to evolve toward predicting user intent by first:

- **Defining Intent:** Intent is a user's immediate, in-the-moment goal

- **Understanding the FM Advantage:**

A specialized recommendation FM can process a much richer history of interactions through tokenization, compression, and efficient sequence modeling, helping the system learn both long-term preferences and current-session signals.

- **The Bullseye Logic:** Because you watched a few action movies last month, most current systems would likely suggest another action movie today. An FM-enabled system adds valuable context: are you on a 4K TV, is it a Friday night, are you continuing a series, browsing for a movie-length experience, or looking for something newly released? This moves the system from simple historical matching toward intent-aware discovery.

A critical correction is that intent should not be positioned as the model guessing a user's mood. A stronger, more measurable framing is to treat it as a set of vectors that can be learned from interaction metadata (e.g., discovery vs. continue-watching, genre preference, movie vs. show preference, and freshness preference).

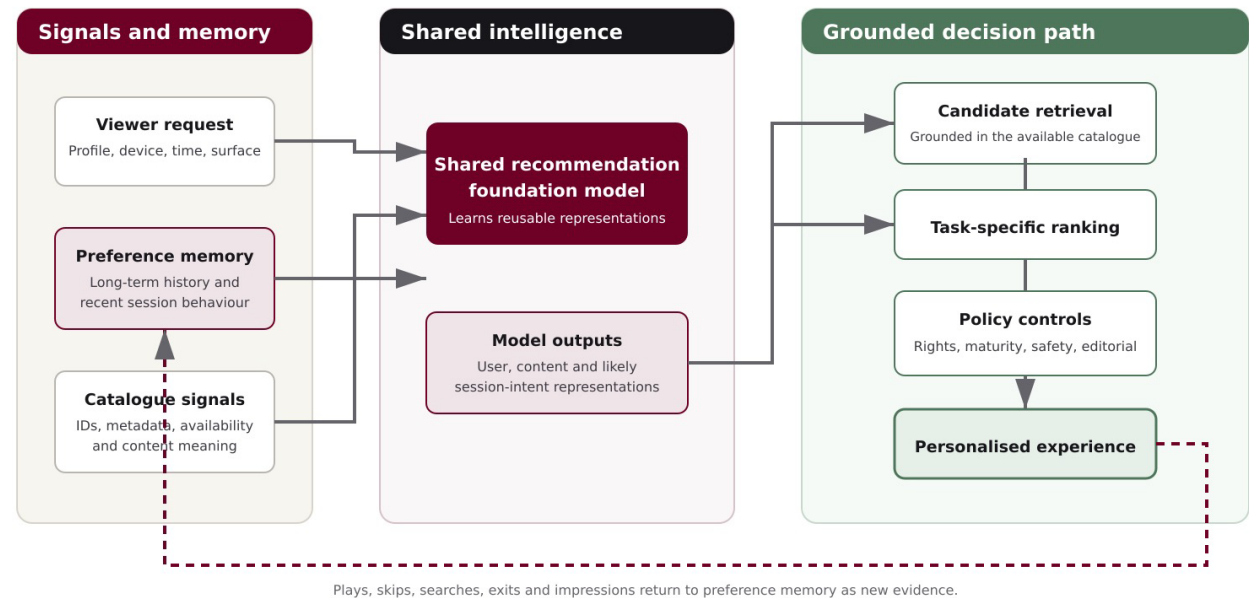
Metric Layer	Metric Examples	What Success Looks Like
Offline Relevance	NDCG@K, Recall@K, Hit Rate, MRR	Better-ranked candidate lists in offline evaluation
Discovery Quality	Time-to-First-Play, search exits, row engagement	Users find something relevant faster
Catalog Health	Coverage, diversity, novelty, cold-start exposure	More of the paid-for catalog is discoverable
Member Value	Retention, reactivation, viewing depth, fatigue signals	Personalization improves durable engagement
Operational Efficiency	P95 latency, CPU/RPS, serving cost per request	Quality gains remain affordable at scale

2.2 Retrieval-Augmented Personalization Pattern

To turn this shared personalization layer into a production-ready asset, we propose a retrieval-augmented personalization pattern consisting of:

- **Memory (Collaborative Filtering):** Retains deep-seated, long-term user habits to avoid genres, themes, or formats the user genuinely dislikes.

FM-Enabled Personalisation Decision Flow



Proposed reference architecture

- **Logic (Foundation Model Reasoning):** A recommendation FM learns from long-term behavior, session context, and catalog semantics to infer the likely intent behind a request. This reasoning does not claim to read user emotions directly.
- **Catalog Bridge (Retrieval):** Ensures content suggestions are grounded in the actual content library and continue to respect availability, rights, safety, editorial, and business constraints.

FIGURE 3: FM-Enabled Personalisation Decision Flow

2.3 Netflix Artwork Personalization

A recent industry proof point is Netflix's latest research on evidence personalization, which uses LLM post-training to select the optimal artwork for a title after it has been recommended.

- **The Strategic Lesson:** Personalization doesn't stop when a title has been chosen to be shown to a viewer. Artwork, synopses and trailers help a viewer decide whether to press play and improving this layer makes the final recommendation more persuasive and relevant.
- **The Measured Result:** Netflix trained its models on 110,000 examples and evaluated them on 5,000 held-out user-title pairs. The post-trained models improved the paper's artwork-selection metric by 3-5% over the Netflix production model used for comparison.^[5]

2.4 Tokenization of User Interactions and Cold-Start Resolution

User histories in the form of play button presses, searches, pauses, rewatches, abandons, device context, time, and catalog metadata need to be compressed into meaningful tokens. This is because a foundation model cannot learn from a user's history unless that history is organized into a useful sequence. A viewer's raw viewing data contains repeated actions including partial plays, searches, browsing activity, timestamps, devices and information about the titles involved.

If each of these actions is treated separately, the sequence becomes too long and expensive to process whereas if too many actions are combined, then important signals may disappear. The task is therefore to compress the viewers' history without losing the nuances that help explain user preference.

Netflix addresses this through interaction tokenization. Related actions involving the same title can be merged into a richer interaction token while preserving information such as watch duration, type of engagement, time, device and content metadata.^[3]

This allows the foundation model to learn from a longer period of user behaviour while keeping training and inference practical. This implies that tokenization must be tested carefully because poor compression can muddy the understanding of the difference between a title that was deliberately watched, briefly sampled or quickly abandoned by a viewer.

A recommendation FM must also address the cold start problem i.e new titles and/or new users to the streaming platform. A traditional ID-based recommendation model has little behavioral evidence for titles newly added to the catalog. But Metadata- such as genre, storyline tone, language, cast and country of origin can be combined with the title's ID-based embedding. This is what Netflix describes as an age-aware approach in which a new title relies more heavily on metadata. As interaction data grows, the foundation model can place more weight on what viewers actually do. This gives new and pre-launch titles a reasonable starting representation without assuming that metadata can replace real audience behavior.^[3]

Cold start problems are thus reduced, with the quality of the initial recommendation highly dependent on the quality of the metadata and on how quickly the model learns from viewer engagement.

3.Engineering the Transition: Operationalizing the Unified Brain

Transitioning to an FM architecture is about bridging the gap between cutting-edge research and ROI. Platforms need to balance data readiness, risk mitigation, cold-start handling, and the economics of token and embedding inference.

3.1 Data Readiness: A foundation model no matter how good it is, cannot compensate for poor interaction data. If user identities, viewing events and catalog metadata are inconsistent, the model will learn an equally inconsistent view of the viewing audience.

Four areas need attention before model training begins:

- **Build a Data Refinery:** Bring viewing, search, browsing and presentation events into a consistent event model. The exact same viewer, title and session must not carry different identities across systems.
- **Record What Was Shown:** A play or click has little meaning unless the platform also knows which titles were shown, where they appeared and what the viewer chose not to select. This helps reduce presentation and selection bias.
- **Improve Catalog Meaning:** Basic genre tags should be supported by richer information such as storyline, tone, themes, language, cast and release context. Generated metadata must be checked rather than accepted automatically.
- **Control Model Refresh:** Interaction data and catalog information should be refreshed according to measured drift and business need. Daily retraining should not become a fixed rule if it adds cost without improving recommendations.

3.2 Risk Mitigation

A full launch that puts the subscriber experience at risk is not advisable. The new layer should prove itself in stages.

- **Offline Evaluation:** Test the model against representative historical sessions using ranking, coverage, diversity and calibration measures.
- **Shadow Testing:** Run the new system alongside the live system without changing what viewers see. This exposes latency, stability and data problems under real traffic.
- **Controlled Experiment:** Send a small, randomly selected audience to the new experience and measure the difference against a control group.
- **Staged Rollout:** Expand only while recommendation quality, latency, cost and safety remain within agreed limits. Keep the current production system available as a fallback.

Historical replay can show whether a model ranks recorded choices more effectively. It cannot prove future retention because viewers never saw alternatives. Retention impact requires a controlled live experiment..

3.3 Inference Economics

A recommendation foundation model can be expensive to run at the scale of a streaming platform with millions of subscribers. The answer is not merely to make the model smaller, rather it is to avoid repeating expensive work on every request.

- **Precompute Stable Outputs:** User and title representations that do not change immediately can be produced in batch and stored for reuse.
- **Keep Sessions Fresh:** Recent activity and current context can be added online without rebuilding the viewer's complete history.
- **Use the Right Model:** A larger foundation model can train shared representations or supervise another model, while smaller models handle lower-cost serving where they are sufficient.
- **Reduce Serving Work:** Distillation, quantization, caching and specialized inference can reduce cost, but their effect on recommendation quality must be measured.
- **Protect the Experience:** A proven fallback should take over when the new route is unavailable or exceeds its latency budget.

Netflix's GenRec uses cost-constrained, prefill-only inference for an LLM-backed ranker, as an example of efficient serving.

Another useful optimization is CPU vectorization through the JDK Vector API that can also accelerate the dot products used inside ranking services.

The Bottom Line:

Implementing foundation model for recommendations must improve viewer engagement and reduce churn without making the service slower, less reliable or too expensive to operate.

The commercial test for this being: can viewers find suitable content faster, does that lead to more meaningful viewing, and can the platform make better use of its catalog at an acceptable cost?

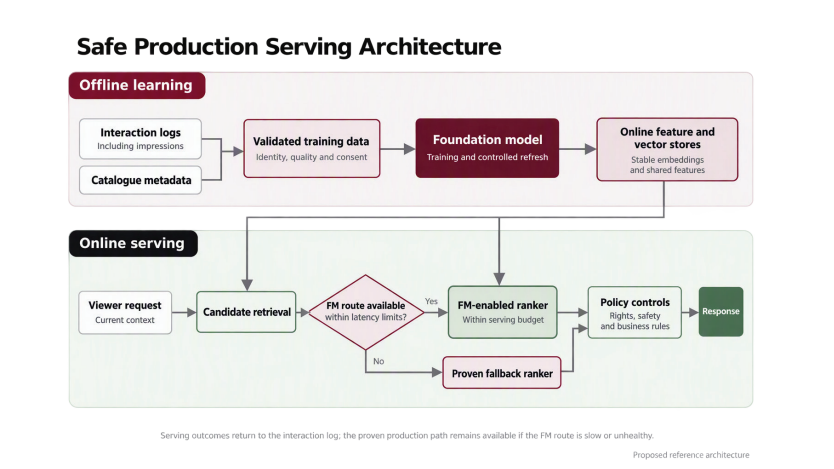


FIGURE 4: Safe Production Serving Architecture

Netflix’s GenPage provides a useful production example. In an online A/B test against the existing homepage recommender, it reported a 0.24% lift in the core engagement metric and a 20% reduction in end-to-end latency.^[9]

The KPI Success Ladder: Fulfilling the Intent Bullseye

The success of an FM-based RecSys is measured by its ability to reduce the 14-minute search window into a 60-second bull’s-eye. This reflects a shift from simple clicks to deep relevance and evidence personalization.

Stage	Metric examples	Decision question
Model quality	NDCG@K, Recall@K, MRR,	Does the model rank relevant titles
Discovery quality	Time-to-first-play, search exits, browse abandonment	Do viewers find something suitable faster?
Viewing quality	Qualified starts, viewing depth, early abandonment	Do viewers watch rather than merely click?
Catalog health	Coverage, diversity, novelty, cold-start exposure	Is more of the catalog discovered without losing relevance?
Member value	Satisfaction, return frequency, retention	Does the improvement last beyond one session?
Economics	P95 latency, cost per request, infrastructure load	Can the improvement be served reliably and affordably?

ROI on the Streaming Content Catalog

Legacy recommendation systems

systems can repeatedly recommend popular titles to viewers because those titles already have the most interaction data. This risks creating a self-reinforcing cycle in which the catalog becomes narrower each time the model learns from its own recommendations.^[10]

While a foundation model does not automatically correct this problem it may reproduce or strengthen popularity bias if the training data and objective reward only immediate engagement.

A streaming content catalog value must therefore be managed very carefully with coverage, diversity, novelty and cold-start exposure measured alongside relevance, with ranking or re-ranking rules applied where needed. The aim being to help each viewer discover a broader set of relevant titles.

Responsible Personalization

A shared preference layer concentrates more behavioral information in one place which makes governance critical.

Platforms should define which data may be used, why it is needed and how long it is retained. They should apply stronger controls to children's profiles, shared households and maturity settings.

Model outputs and policy decisions should be logged and audited so that teams can investigate harmful or unusually narrow recommendations, popularity concentration, uneven catalog exposure and quality differences across user groups.

Viewers should also be able to remove history, correct misleading signals and reset a profile with a continuously learning system also including a practical way to forget viewer preferences.

Conclusion

For a streaming platform, the “14-minute leak” is a very controllable churn vector in today's attention economy. With 49% of viewers ready to cancel due to discovery friction, the passive ML-based recommendation stack used everywhere today is no longer an option. By building a federated, agentic system, a streaming platform can evolve to construct best-in-class entertainment journeys in real time.

For CFOs, the imperative is clear: every second saved in time-to-first-play (TTFP) and every percentage point added to retention directly offset the high cost of content acquisition and user growth.

Any possible risks associated with an overhaul are mitigated by a low-risk, phased adoption strategy. We believe it's time for streaming platforms to evolve now, with news publishers, BFSI and retail media networks, and other ad tech providers soon to follow suit in adopting this new recommendation paradigm.

References

1. Abdollahpouri, H., & Burke, R. (2019). The unfairness of popularity bias in recommendation. arXiv. <https://arxiv.org/abs/1907.13286>
2. Elastic. (2024). Accelerating vector search with SIMD instructions. Elastic Blog. <https://www.elastic.co/blog/accelerating-vector-search-simd-instructions>
3. Gracenote. (2025, November). 2025 State of Play report. Nielsen. <https://www.nielsen.com/news-center/2025/new-gracenote-report-highlights-impact-of-ineffective-content-discovery-on-consumer-happines-s-with-streaming/>
4. Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv. <https://arxiv.org/abs/1503.02531>
5. Nam, H., et al. (2026). Netflix artwork personalization via LLM post-training (arXiv:2601.02764). arXiv. <https://arxiv.org/abs/2601.02764>
6. Netflix Technology Blog. (2025). Thinking fast & slow for a personalized notification system. <https://netflixtechblog.com/thinking-fast-slow-for-a-personalized-notification-system-4d89b26525cd>
7. Netflix Technology Blog. (2025, March). Foundation model for personalized recommendation. <https://netflixtechblog.com/foundation-model-for-personalized-recommendation-1a0bd8e02d39>
8. Netflix Technology Blog. (2025, May). FM-Intent: Predicting user session intent with hierarchical multi-task learning. <https://netflixtechblog.com/fm-intent-predicting-user-session-intent-with-hierarchical-multi-task-learning-94c75e18f4b8>
9. Netflix Technology Blog. (2025, November). Integrating Netflix's foundation model into personalization applications. <https://netflixtechblog.medium.com/integrating-netflixs-foundation-model-into-personalization-applications-cf176b5860eb>
10. Netflix Technology Blog. (2025, October). Post-training generative recommenders with advantage-weighted supervised finetuning. <https://netflixtechblog.com/post-training-generative-recommenders-with-advantage-weighted-supervised-finetuning-61a538d717a9>
11. Netflix Technology Blog. (2026, March). Optimizing recommendation systems with JDK's Vector API. <https://netflixtechblog.com/optimizing-recommendation-systems-with-jdks-vector-api-30d2830401ec>
12. Rajput, S., et al. (2023). Recommender systems with generative retrieval (TIGER). NeurIPS. <https://arxiv.org/abs/2305.05065>
13. Yan, S., et al. (2024). Corrective retrieval augmented generation. arXiv. <https://arxiv.org/abs/2401.15884>

Authors

Jay Chauhan

Group Practice Head, Tech Mahindra

Jay is leading the global M&E business and its AI-led growth strategy at Tech Mahindra. He works across shaping market strategy, driving innovation, and advising media leaders on AI transformation. Jay led the development of Tech Mahindra's M&E service line's first Agentic AI IP for the media industry and is a recognized thought leader through industry research and whitepapers.

Stuart Ure

Strategic Architect, Sky New Zealand

Stuart Ure is a technology and platform architecture leader with over 15 years of experience across digital platforms, online services, and media technology. As Strategic Architect at Sky New Zealand, he helps shape the evolution of broadcast and streaming platforms, with a focus on scalable digital ecosystems, platform strategy, and technology modernization. His expertise spans content delivery, online systems, enterprise architecture, and the design of high-performance platforms that support the future of media and entertainment.

Dr. Anshu Premchand

VP & IBU Head - AI Platforms, Enterprise AI Acceleration & AI Risk, CTO Office, Tech Mahindra

Anshu leads four CTO-direct AI mandates at Tech Mahindra - building the reusable foundations & platforms, governed delivery models and risk architecture needed to scale AI responsibly across the enterprise. She also heads AI Velocity Cells program. With 26 years of experience across AI, cloud, and digital transformation - including her previous role as Global Head of Solutions and Architecture for the Google Business Unit at Tata Consultancy Services - she brings a global lens to enterprise AI adoption. Anshu holds a Ph.D. in Computer Science and continues to publish widely, with 35+ peer-reviewed papers across IEEE, ACM, Springer and JARDCS, contributing to the ongoing research shaping how enterprises build and govern AI at scale.

About **Tech Mahindra**

Tech Mahindra (NSE: TECHM, BSE: 532755) offers technology consulting and digital solutions to global enterprises across industries, enabling transformative scale at unparalleled speed. With 146,000+ professionals across 90 countries, Tech Mahindra provides a full spectrum of services including consulting, information technology, enterprise applications, business process services, engineering services, network services, customer experience & design, AI & analytics, and cloud & infrastructure services. It is the first Indian company in the world to have been awarded the Sustainable Markets Initiative's Terra Carta Seal, which recognizes global companies that are actively leading the charge to create a climate and nature-positive future. Tech Mahindra is part of the Mahindra Group, founded in 1945, one of the largest and most admired multinational federation of companies. For more information on how TechM can partner with you to meet your Scale at Speed™ imperatives, please visit <https://www.techmahindra.com/>.



www.techmahindra.com

www.linkedin.com/company/tech-mahindra

www.x.com/tech_mahindra

Copyright © Tech Mahindra Ltd 2026. All Rights Reserved.

Disclaimer: Brand names, logos, taglines, service marks, tradenames and trademarks used herein remain the property of their respective owners. Any unauthorized use or distribution of this content is strictly prohibited. The information in this document is provided on "as is" basis and Tech Mahindra Ltd. makes no representations or warranties, express or implied, as to the accuracy, completeness or reliability of the information provided in this document. This document is for general informational purposes only and is not intended to be a substitute for detailed research or professional advice and does not constitute an offer solicitation, or recommendation to buy or sell any product, service or solution. Tech Mahindra Ltd. shall not be responsible for any loss whatsoever sustained by any person or entity by reason of access to, use of or reliance on, this material. Information in this document is subject to change without notice.