Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/

Research Article

Training Models on Sparse Data and Labels in Recommendation Systems: Overcoming Supervision and Feedback Limitations

Abhishek Kumar
Meta, USA

ARTICLE INFO	ABSTRACT
Received: 01 Oct 2025 Revised: 22 Nov 2025 Accepted: 02 Dec 2025	Large-scale recommendation systems consistently grapple with extreme data and label sparsity, where the vast majority of potential user-item interactions remain unobserved and unlabeled. This fundamental constraint severely limits model training, introducing biased gradients, degrading personalization quality, and amplifying popularity bias toward frequently interacted items while neglecting long-tail content. The challenge intensifies as platforms scale to serve diverse user populations across massive item catalogs, where traditional supervised learning approaches prove inadequate. This article provides a comprehensive examination of contemporary strategies designed to train robust recommendation models under severe supervision constraints. The article encompasses self-supervised learning techniques that leverage unlabeled interaction data through contrastive and pretext objectives, implicit feedback modeling approaches that extract weak supervision from behavioral signals while mitigating inherent biases, and transfer learning frameworks that incorporate knowledge from auxiliary tasks and cross-domain sources. Critical analysis reveals the practical trade-offs between sample efficiency, computational scalability, and recommendation fidelity across different sparsity regimes. By synthesizing theoretical foundations with real-world deployment considerations, this work offers actionable guidance for researchers and practitioners seeking to improve recommendation accuracy, enhance cold-start robustness, and expand long-tail coverage in production environments where explicit feedback remains scarce yet personalization expectations continue rising. The article bridges academic innovation with industrial implementation realities facing modern recommendation systems. **Keywords:** Recommendation Systems Sparsity, Self-Supervised Contrastive Learning, Implicit Feedback Debiasing, Transfer Learning, Multi-Task, Cold-Start Personalization

1. Introduction

Modern recommendation systems face a fundamental challenge: the vast majority of potential user-item interactions remain unobserved, creating extreme data sparsity that undermines model training and personalization quality. In typical production environments, explicit feedback such as ratings or purchases accounts for less than 0.1% of all possible interactions, leaving recommendation algorithms to infer user preferences from an overwhelmingly sparse signal landscape. This scarcity of labeled data becomes particularly acute as platforms scale to millions of users and items, where traditional supervised learning approaches struggle to generalize effectively.

The implications extend beyond simple data availability. Sparse feedback amplifies existing biases toward popular items, degrades cold-start performance for new users and content, and limits the discovery of long-tail recommendations that could enhance user satisfaction. While implicit signals like clicks, views, and dwell time offer more abundant data, they introduce noise, selection bias, and ambiguous intent that complicate model training. Recent advances in self-supervised learning,

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

contrastive objectives, and transfer learning have opened new pathways to leverage unlabeled data and auxiliary tasks, yet systematic integration of these techniques into production systems remains challenging.
This article examines practical strategies for training robust recommendation models under severe supervision constraints. The discussion synthesizes approaches across self-supervised representation learning, implicit feedback modeling with debiasing, and multi-task transfer frameworks. By analyzing trade-offs between sample efficiency, computational cost, and recommendation quality, this work guides researchers and practitioners navigating the sparse data regimes that define real-world personalization systems [1].

2. Background and Related Work

2.1 Fundamentals of Recommendation Systems

Recommendation systems rely on three primary paradigms: collaborative filtering, which identifies patterns from user-item interaction matrices; content-based methods, which match item attributes to user profiles; and hybrid approaches that combine both strategies to mitigate individual weaknesses. Collaborative filtering dominates industrial applications due to its ability to capture latent preference structures without requiring detailed content metadata [2]. However, the distinction between explicit feedback (ratings, likes) and implicit feedback (clicks, views) fundamentally shapes algorithm design. Explicit signals provide clear preference indicators but suffer from scarcity, while implicit signals offer volume at the cost of ambiguity and noise. Large-scale deployments face additional challenges: real-time inference requirements, computational constraints, and the need to balance exploration with exploitation across millions of concurrent users.

2.2 The Sparsity Problem in Recommendation

Data sparsity can be formalized as the ratio of observed interactions to the total user-item matrix size, often exceeding 99.9% in production systems. This creates two critical failure modes: the cold-start problem, where new users or items lack sufficient interaction history for accurate recommendations, and long-tail bias, where algorithms overfit to popular items while neglecting niche content. Sparsity directly impacts model convergence by reducing gradient signal quality and limiting the effective training data available for learning user-item representations. The Matthew effect compounds this issue, as popular items receive disproportionate exposure, further starving the tail of content of feedback needed for discovery.

2.3 Traditional Approaches to Sparsity

Matrix factorization techniques like Singular Value Decomposition (SVD) and Non-negative Matrix Factorization (NMF) dominated early sparsity mitigation efforts by decomposing interaction matrices into lower-dimensional latent factor spaces. These methods employ regularization to prevent overfitting on sparse observations and use negative sampling to approximate unobserved interactions as implicit negatives. However, linear factorization struggles to model complex, non-linear user-item relationships that characterize modern recommendation tasks. The transition to deep learning exposed fundamental limitations: shallow embeddings lack representational capacity, and hand-crafted negative sampling heuristics fail to capture the nuanced structure of user preferences across diverse contexts.

2.4 Evolution of Deep Learning in Recommendations

Neural collaborative filtering revolutionized recommendation modeling by replacing inner products with multi-layer perceptrons capable of learning arbitrary user-item interaction functions. Deep matrix factorization extended this by integrating neural architectures with traditional factorization objectives, while attention mechanisms enabled selective focus on relevant historical interactions in sequential recommendation tasks. Transformer architectures have recently gained traction for modeling long-range dependencies in user behavior sequences. Graph neural networks represent the latest frontier, encoding user-item bipartite graphs and social networks to propagate information

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

across connected entities, particularly valuable for addressing cold-start scenarios through neighborhood aggregation [3].

3. Self-Supervised Learning for Sparse Recommendations

3.1 Principles of Self-Supervised Learning

Self-supervised learning generates supervisory signals from unlabeled data through carefully designed pre-text tasks that encourage models to learn meaningful representations. The core principles include invariance—where augmented views of the same entity yield similar embeddings—and equivariance, where specific transformations produce predictable embedding changes. This paradigm connects directly to unsupervised representation learning by eliminating dependence on expensive manual labels while extracting semantic structure from raw interaction data.

3.2 Contrastive Learning Frameworks

Contrastive methods adapted from computer vision (SimCLR, MoCo) have been successfully applied to recommendation by treating user sessions or item co-occurrences as positive pairs while contrasting against randomly sampled negatives. User-user contrastive objectives align the embeddings of users with similar interaction histories, while item-item objectives group functionally similar products. Session-based contrastive learning treats temporally adjacent interactions as semantically related, leveraging sequential structure. Temperature scaling controls the sharpness of similarity distributions, and hard negative mining focuses learning on challenging examples that lie near decision boundaries.

3.3 Data Augmentation Strategies

Effective augmentation for recommendation requires domain-specific techniques: feature masking randomly drops interaction features to prevent overfitting, while temporal augmentation reorders or samples subsequences from behavior histories. Graph structure augmentation perturbs edges in user-item graphs to create alternative views that preserve semantic content. Cross-modal augmentation leverages complementary signals—combining textual item descriptions with visual features—to enrich representations beyond interaction data alone.

Approach	Core Mechanism	Advantages	Limitations	Typical Performance Gain
SimCLR/MoCo Adaptations	User sessions/item co-occurrences as positive pairs with contrastive learning	Leverages unlabeled interaction data; captures semantic similarity	Requires careful negative sampling; computationally intensive	5-15% accuracy improvement on sparse datasets

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

BYOL for Recommendations	Online network predicts target network outputs from augmented views without explicit negatives	Eliminates negative pair dependency; simpler training	Risk of representation collapse; requires careful architecture design	20-30% cold-start improvement for users with <10 interactions
Barlow Twins	Decorrelates embedding dimensions to reduce redundancy	Prevents trivial solutions; maintains representational diversity	Sensitive to batch size; requires dimension balancing	Manageable 1.3-1.8× training time overhead
Session-Based Contrastive Learning	Treats temporally adjacent interactions as semantically related	Captures sequential patterns; exploits temporal structure	Limited to sequential recommendation contexts	Effective for conversion optimization in e-commerce

Table 1: Comparison of Self-Supervised Learning Approaches for Sparse Recommendations [3]

3.4 Self-Supervised Architectures

Siamese networks process user and item inputs through shared encoders, learning representations that maximize similarity for positive pairs. Bootstrap Your Own Latent (BYOL) eliminates explicit negative pairs by training online networks to predict target network outputs from augmented views. Barlow Twins reduces representational redundancy by decorrelating embedding dimensions, preventing collapse to trivial solutions. These architectures integrate seamlessly with existing recommendation backbones, adding auxiliary self-supervised objectives to standard supervised losses.

3.5 Empirical Results and Case Studies

Empirical studies demonstrate that self-supervised pre-training improves recommendation accuracy by 5-15% on sparse datasets compared to purely supervised baselines. Cold-start metrics show particularly dramatic gains, with 20-30% improvements in predicting preferences for users with fewer than ten interactions. Computational overhead remains manageable—typically 1.3-1.8× training time increases—making these techniques viable for production deployment. Industry adoption at platforms handling billions of daily recommendations confirms practical scalability [4].

4. Implicit Feedback Modeling and Debiasing

4.1 Understanding Implicit Signals

Implicit feedback encompasses diverse behavioral signals: clicks indicate initial interest, views measure exposure, dwell time reflects engagement depth, and scroll patterns reveal content consumption. Each signal carries distinct noise characteristics—clicks may be accidental, views don't guarantee attention, and dwell time conflates genuine interest with user distraction. Temporal dynamics further complicate interpretation, as session context shapes behavior: morning browsing differs fundamentally from evening exploration. Inferring user intent requires distinguishing between exploratory clicks, purposeful engagement, and habitual patterns, making implicit signals simultaneously abundant and ambiguous.

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

4.2 Modeling Techniques for Implicit Feedback

Ranking loss functions provide the mathematical foundation for implicit feedback modeling. Pointwise methods treat each interaction independently, pairwise approaches like Bayesian Personalized Ranking (BPR) optimize relative preferences between observed and unobserved items, and listwise losses consider entire recommendation slates. Weighted matrix factorization assigns confidence scores to interactions based on signal strength, while neural implicit feedback models employ deep architectures to capture non-linear relationships between contextual features and user responses, enabling more nuanced preference modeling than traditional linear methods.

4.3 Bias in Implicit Feedback

Position bias systematically inflates engagement with top-ranked items regardless of relevance, as users exhibit strong primacy effects in sequential scanning. Selection bias arises because users only interact with exposed items, creating missing-not-at-random patterns that confound causal inference. Popularity bias perpetuates the Matthew effect—widely consumed items receive disproportionate future exposure, starving niche content of discovery opportunities. Temporal biases reflect shifting user preferences and seasonal patterns, while contextual factors like device type or time-of-day introduce systematic variation in interaction propensities.

4.4 Debiasing Methodologies

Inverse propensity scoring (IPS) reweights training samples by estimated exposure probabilities, correcting for selection bias in observational data. Causal inference frameworks employ techniques like propensity score matching and instrumental variables to identify true preference effects from confounded observations [5]. Unbiased learning-to-rank methods explicitly model position and presentation biases during training, producing rankings optimized for true relevance rather than biased click patterns. Counterfactual reasoning estimates what would have happened under alternative exposures, enabling evaluation of recommendation quality beyond observed interactions.

4.5 Negative Sampling Strategies

Random negative sampling treats unobserved items as implicit negatives but suffers from uninformative easy negatives. Hard negative mining selects challenging examples near decision boundaries to accelerate learning. Popularity-based sampling adjusts negative distributions to match item frequency, preventing models from trivially exploiting popularity signals. Graph-based sampling leverages network structure to identify semantically plausible but unselected alternatives. Dynamic negative sampling adapts sample distributions during training as model predictions evolve, while false negative mitigation techniques identify and downweight likely mislabeled negatives among unobserved interactions.

4.6 Performance Evaluation

Offline evaluation of implicit feedback systems relies on metrics like precision, recall, and normalized discounted cumulative gain (NDCG) computed on held-out interactions, though these poorly predict online performance due to feedback loops. A/B testing remains essential for measuring real-world impact but requires careful experimental design to detect effects amid high variance. Critical trade-offs emerge between short-term engagement metrics (clicks, watch time) and long-term satisfaction measures (retention, subscription renewal). Sophisticated practitioners now track user lifetime value and content diversity exposure as counterbalances to pure engagement optimization.

5. Transfer Learning and Multi-Task Frameworks

5.1 Pre-training Paradigms for Recommendations

Transfer learning addresses data sparsity by leveraging knowledge from related tasks or domains. Domain-specific pre-training objectives include masked item prediction, where models learn to reconstruct randomly obscured items from interaction sequences, and next-item prediction tasks that capture temporal dynamics. Cross-domain knowledge transfer proves particularly valuable when transferring learned representations from data-rich domains (e.g., movies) to sparse domains (e.g.,

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

niche hobbies). Foundation models for recommendations adapt large-scale language model pre-training techniques, treating user behavior sequences as sentences and items as tokens, enabling models to learn generalizable interaction patterns before fine-tuning on specific recommendation tasks [6].

5.2 Multi-Task Learning Architectures

Multi-task learning combats sparsity by sharing representations across related prediction objectives. Shared-bottom architectures employ common lower layers with task-specific prediction heads, enabling parameter efficiency but risking negative transfer when tasks conflict. Multi-gate Mixture-of-Experts (MMoE) addresses this limitation by routing inputs through task-specific combinations of expert networks, allowing selective knowledge sharing. Progressive Layered Extraction (PLE) further refines this approach with task-specific and shared expert groups at multiple network depths. Task relationship learning automatically discovers optimal sharing patterns, while dynamic task weighting adjusts loss contributions based on training progress and task difficulty.

5.3 Auxiliary Task Design

Carefully designed auxiliary tasks provide additional supervision signals. Click-through rate (CTR) prediction offers abundant implicit feedback for learning engagement patterns. Conversion and revenue prediction align recommendations with business objectives while providing sparse but high-value labels. User engagement metrics like dwell time and completion rate capture satisfaction dimensions beyond simple clicks. Content understanding tasks—classifying items by category, predicting tags, or generating descriptions—inject semantic knowledge into item representations, particularly valuable for cold-start items lacking interaction history.

5.4 Cross-Modal Transfer Learning

Cross-modal transfer exploits rich semantic information from multiple modalities. Text-to-recommendation transfer leverages pre-trained language models to encode item descriptions, reviews, and metadata into semantically meaningful embeddings. Vision-to-recommendation transfer applies pre-trained image encoders to product photos or video thumbnails, capturing visual attributes that influence preferences. Multimodal fusion strategies combine text, vision, and interaction signals through attention mechanisms or gated fusion layers. CLIP-style contrastive pre-training aligns items with natural language descriptions, enabling zero-shot recommendation for items without interaction history.

5.5 Meta-Learning for Cold-Start

Meta-learning frameworks train models to rapidly adapt to new users or items with minimal data. Model-Agnostic Meta-Learning (MAML) adaptations optimize for quick fine-tuning, learning initialization parameters that generalize across users. Few-shot learning techniques enable preference prediction from just a handful of interactions by learning similarity metrics that compare new users to existing profiles. Metric learning approaches embed users and items in spaces where distance reflects preference strength. Personalized initialization strategies leverage demographic or contextual features to select appropriate starting parameters for each new user.

5.6 Knowledge Distillation

Knowledge distillation transfers capabilities from large, complex teacher models to efficient student models suitable for production deployment. Teacher-student frameworks train compact networks to mimic the soft predictions of ensemble or over-parameterized teachers, preserving performance while reducing computational costs. This approach proves essential for transferring insights from large-scale pre-trained models to latency-constrained serving environments. Dark knowledge—the information in teacher prediction distributions beyond hard labels—captures nuanced item relationships. Continual learning extensions enable knowledge retention as new data arrives, preventing catastrophic forgetting of previously learned patterns.

6. Hybrid and Ensemble Approaches

6.1 Combining Multiple Learning Paradigms

Hybrid approaches integrate self-supervised, supervised, and semi-supervised objectives within unified training frameworks. Multi-stage training pipelines typically begin with self-supervised pre-training on unlabeled interaction data, followed by supervised fine-tuning on explicit feedback. Joint optimization strategies simultaneously minimize multiple loss functions—combining contrastive losses for representation learning with ranking losses for prediction accuracy. Balancing multiple loss functions requires careful tuning of weighting coefficients, often using uncertainty-based weighting schemes that automatically adjust relative importance based on task-specific gradient magnitudes and learning rates.

6.2 Ensemble Methods for Robustness

Ensemble techniques improve recommendation robustness by combining predictions from diverse models. Model averaging pools outputs from multiple independently trained networks, reducing variance and improving generalization on sparse data. Stacking learns meta-models that optimally weight base model predictions. Diverse architecture ensembles combine fundamentally different model types—matrix factorization, neural networks, and graph models—to capture complementary patterns. Temporal ensemble techniques maintain exponential moving averages of model parameters across training iterations, smoothing predictions and improving stability. Uncertainty quantification through ensembles enables confidence-aware ranking, surfacing high-certainty recommendations for risk-averse scenarios.

6.3 Active Learning and Human-in-the-Loop

Active learning strategically selects samples for manual annotation to maximize information gain per labeling cost. Uncertainty-based acquisition functions prioritize ambiguous examples where model predictions exhibit high variance or low confidence. Interactive recommendation refinement collects explicit feedback on algorithmically selected items, efficiently exploring user preference boundaries. Cost-effective annotation strategies balance the expense of human labels against expected model improvements, focusing annotation budgets on high-impact samples like cold-start users or semantically ambiguous items where automated labels prove unreliable.

7. Evaluation Methodologies and Metrics

7.1 Offline Evaluation Challenges

Offline evaluation of recommendation systems requires careful consideration of temporal dynamics to avoid data leakage. Train-test split strategies for temporal data must respect chronological ordering, using past interactions to predict future behavior rather than random splits that break causality. Cross-validation in sparse settings faces unique challenges, as traditional k-fold approaches may create unrealistically dense training sets or leave insufficient data in validation folds. Standard ranking quality metrics include Normalized Discounted Cumulative Gain (NDCG), which weights relevance by position; Mean Average Precision (MAP), emphasizing precision across rank positions; and Mean Reciprocal Rank (MRR), focusing on the first relevant result. Coverage metrics measure the proportion of catalog items that receive recommendations, while diversity metrics assess variety within recommendation lists, both critical for combating popularity bias in sparse environments [7].

7.2 Online Evaluation Considerations

A/B testing remains the gold standard for online evaluation but requires careful experimental design to achieve adequate statistical power, particularly when measuring subtle quality improvements or rare conversion events. Multi-armed bandit approaches balance exploration and exploitation, dynamically allocating traffic to promising variants while gathering information about alternatives. The tension between long-term metrics (user retention, lifetime value) and short-term metrics (click-through rate, immediate engagement) creates evaluation complexity, as optimizing for clicks may harm long-term satisfaction. Novelty and serendipity assessment captures recommendation systems'

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

ability to surface unexpected yet delightful items, dimensions poorly reflected in accuracy-focused metrics but essential for user experience.

7.3 Bias and Fairness Evaluation

Fairness evaluation encompasses multiple dimensions. Group fairness metrics measure whether different demographic segments receive equitable recommendation quality, detecting systematic disadvantages for minority groups. Individual fairness considerations ensure similar users receive similar treatment, preventing arbitrary discrimination. Filter bubble and echo chamber detection identifies when recommendation systems create ideological isolation by over-personalizing content exposure. Exposure fairness ensures items receive recommendation opportunities proportional to their relevance, while calibration fairness matches recommendation distributions to user preference distributions, preventing over-representation of particular content types [8].

7.4 Cold-Start Performance Assessment

Cold-start evaluation requires specialized metrics that measure system effectiveness with limited data. New user onboarding metrics track how quickly recommendation quality improves as initial interactions accumulate, typically measuring performance after 1, 5, and 10 interactions. New item integration speed assesses how rapidly novel catalog additions receive appropriate exposure, critical for time-sensitive content like news or trending products. Cross-domain transfer effectiveness evaluates whether knowledge learned in data-rich domains successfully transfers to sparse target domains. Few-shot learning benchmarks systematically test model performance across varying data availability levels, providing comprehensive cold-start characterization.

8. Practical Considerations and System Design

8.1 Scalability and Computational Efficiency

Production recommendation systems demand extreme scalability to serve millions of concurrent users. Training pipeline optimization employs techniques like gradient accumulation, mixed-precision training, and efficient data loading to maximize hardware utilization. Distributed training strategies partition models and data across multiple machines using parameter servers or ring-allreduce architectures, enabling training on datasets too large for single-machine memory. Approximate nearest neighbor search algorithms like HNSW or FAISS enable sub-millisecond retrieval from billion-scale item catalogs during inference. Model serving optimization includes quantization, pruning, and graph optimization to meet strict latency budgets—often sub-100ms—required for real-time personalization [9].

8.2 Data Infrastructure Requirements

Robust data infrastructure forms the foundation of production systems. Logging captures comprehensive interaction data, including impressions, clicks, conversions, and contextual features, with minimal performance impact. Feature engineering pipelines transform raw events into model-ready representations, handling missing values, categorical encoding, and temporal aggregations. Real-time processing streams enable immediate incorporation of recent interactions, while batch processing handles computationally intensive feature computations overnight. Feature stores provide centralized repositories with consistent online and offline feature access, ensuring training-serving consistency and enabling feature reuse across models. Data quality validation detects anomalies, schema violations, and distributional shifts that could corrupt model training.

8.3 Model Deployment and Monitoring

Deployment infrastructure orchestrates complex model lifecycle management. A/B testing infrastructure randomly assigns users to treatment groups, tracks metrics, and performs statistical significance testing to validate model improvements before full rollout. Model performance tracking continuously monitors business metrics, prediction quality, and system health through comprehensive dashboards. Concept drift detection identifies when data distributions shift, degrading model accuracy and triggering retraining. Retraining strategies balance model freshness against computational costs, typically employing daily or weekly schedules for main models with more frequent updates for fast-moving components like trending item boosters.

8.4 Privacy and Ethical Considerations

Privacy-preserving recommendation systems employ differential privacy to add calibrated noise during training, protecting individual user data while maintaining aggregate utility—particularly challenging in sparse settings where individual contributions carry more information. Federated learning approaches train models on decentralized user devices without centralizing raw interaction data, though communication costs and device heterogeneity introduce technical complexity. User consent and transparency mechanisms explain why specific items appear in recommendations, providing algorithmic recourse and control. Algorithmic accountability frameworks establish processes for auditing recommendation outcomes, identifying discriminatory patterns, and ensuring human oversight of automated decisions affecting users' information access and opportunities.

9. Case Studies and Industry Applications

9.1 E-commerce Recommendations

E-commerce platforms face extraordinary sparsity challenges with millions of products and infrequent purchase patterns. Product discovery under extreme sparsity requires sophisticated techniques combining collaborative signals with content features like categories, brands, and attributes. Cross-category transfer learning proves essential when users transition from browsing electronics to clothing, leveraging behavioral patterns that generalize across disparate product types. Session-based recommendations capture transient intent during single shopping trips, crucial for converting browsing to purchases. Leading e-commerce platforms report double-digit improvements in conversion rates and average order values when deploying advanced sparse-data techniques, demonstrating substantial business impact beyond traditional accuracy metrics.

9.2 Content Streaming Platforms

Streaming services confront unique challenges where video and music recommendations must satisfy diverse consumption contexts—background listening versus focused viewing, mood-based selection, and genre exploration. Sequential modeling with sparse interactions becomes critical as users exhibit complex temporal patterns: binge-watching series, returning to favorite artists, or exploring new genres. Multi-objective optimization balances immediate engagement metrics like watch time against long-term satisfaction indicators such as subscription retention and content diversity exposure. Real-world deployment insights from major platforms reveal that naive engagement optimization can create filter bubbles that eventually diminish user satisfaction, necessitating careful objective design [10].

9.3 Social Media and News Feeds

Social platforms require real-time personalization at unprecedented scale, processing billions of content items and user interactions daily. Cold-start for viral content presents distinct challenges: new posts lack interaction history yet must surface rapidly to capitalize on trending dynamics. Balancing relevance and diversity prevents echo chambers while maintaining user engagement, a tension particularly acute in news and political content. Platform-specific considerations include algorithmic amplification effects, misinformation risks, and regulatory scrutiny around content curation, demanding technical solutions that incorporate safety constraints alongside personalization objectives.

9.4 Advertising and Sponsored Content

Digital advertising operates under extreme label sparsity, as conversion events occur orders of magnitude less frequently than impressions or clicks. Multi-touch attribution attempts to assign credit across user touchpoints preceding conversion, complicated by sparse observation of complete user journeys. Bidding optimization under uncertainty requires balancing exploration of new ad variants against exploitation of proven performers, typically addressed through contextual bandit algorithms.

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/

Research Article

Regulatory compliance and transparency requirements increasingly mandate explainable recommendations and user consent mechanisms, adding constraints beyond pure performance optimization.

10. Future Directions and Open Challenges

10.1 Emerging Research Directions

Large language models for recommendations represent a paradigm shift, treating recommendations as language generation where user histories and item catalogs are encoded as text prompts. Early experiments demonstrate zero-shot recommendation capabilities and natural language explanation generation, though computational costs and factual grounding remain open challenges. Reinforcement learning with sparse rewards addresses sequential decision-making where recommendations influence future user states and long-term value, requiring stable training despite delayed and infrequent reward signals. Causal recommendation systems move beyond correlational predictions to estimate treatment effects, answering counterfactual questions like "would this user engage if exposed to alternative content?" Explainable AI for sparse models addresses the interpretability gap, generating human-understandable rationales for recommendations despite operating on limited interaction data.

10.2 Technical Challenges

Scalability to billions of users and items demands architectural innovations beyond current distributed systems, particularly as models grow more complex with multimodal inputs and transformer architectures. Real-time adaptation to user preferences requires online learning systems that incorporate fresh interactions within seconds while maintaining model stability and preventing degradation from noisy signals. Cross-platform and cross-device consistency becomes critical as users fragment attention across smartphones, tablets, desktops, and connected TVs, requiring unified identity resolution and preference synchronization. Handling extreme cold-start scenarios—new users providing zero initial interactions or launching services in entirely new markets—pushes current techniques to their limits.

10.3 Ethical and Societal Implications

Algorithmic transparency requirements increasingly mandate that platforms explain recommendation logic to users and regulators, challenging systems optimized purely for prediction accuracy. User control and preference elicitation mechanisms must evolve beyond simple explicit feedback collection to give users meaningful agency over their algorithmic experiences without overwhelming them with complexity. Impact on content creators and platforms creates ecosystem effects where recommendation algorithms shape creative production, potentially homogenizing content toward algorithmically favored patterns. Long-term effects on user behavior remain poorly understood: do personalized recommendations enhance discovery or create dependency, broaden horizons or narrow them, and how do these dynamics evolve across user lifecycles?

10.4 Standardization and Benchmarking

Public datasets with realistic sparsity remain scarce, as most available benchmarks either lack production-scale size or fail to capture real-world sparsity patterns, limiting research reproducibility. Reproducibility in recommendation research suffers from inconsistent evaluation protocols, proprietary datasets, and complex production contexts that academic settings struggle to replicate. Fair comparison frameworks must account for diverse objectives across domains—engagement for streaming, conversion for e-commerce, satisfaction for social platforms—preventing misleading performance claims. Industry-academic collaboration opportunities could address these gaps through data sharing agreements, open-source reference implementations, and shared evaluation platforms that preserve commercial sensitivity while enabling rigorous research advancement.

Conclusion

Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

Addressing data and label sparsity represents one of the most pressing challenges in modern recommendation systems, where the gap between available signals and the scale of personalization demands continues to widen. This article has examined a comprehensive range of techniques—from self-supervised and contrastive learning that extract meaningful patterns from unlabeled interactions, to implicit feedback modeling with sophisticated debiasing methods, to transfer learning frameworks that leverage cross-domain knowledge and auxiliary tasks. The synthesis reveals that no single approach dominates across all contexts; rather, practitioners must carefully balance trade-offs between sample efficiency, computational overhead, and recommendation quality based on their specific sparsity regime and business objectives. Real-world deployments demonstrate that combining multiple paradigms—such as self-supervised pre-training followed by multi-task fine-tuning with debiased implicit feedback—often yields superior results compared to isolated techniques. Looking forward, the field faces both technical frontiers and societal responsibilities: scaling to ever-larger catalogs while maintaining real-time responsiveness, integrating emerging capabilities from large language models, and ensuring algorithmic transparency and fairness even under extreme data constraints. Success in sparse recommendation environments ultimately requires not just algorithmic innovation but also thoughtful system design, rigorous evaluation methodologies, and careful consideration of long-term impacts on users, creators, and platforms alike. As personalization becomes increasingly central to digital experiences, the ability to learn effectively from limited supervision will distinguish leaders from followers in delivering relevant, diverse, and satisfying recommendations.

References

[1] Schahin Tofangchi, et al., "Handling the Efficiency–Personalization Trade-Off in Service Robotics: A Machine-Learning Approach", Journal of Management Information Systems, 02 Apr 2021. https://www.tandfonline.com/doi/citedby/10.1080/07421222.2021.1870391?scroll=top&needAccess=true
[2] Devangam Bangaru Rajesh, Avadhesh Kumar. "Collaborative filtering models an experimental and detailed comparative study", Scientific Reports, 28 August 2025. https://www.nature.com/articles/s41598-025-15096-4
[3] Aarav Patel, et al., "Graph Neural Networks for Multi-modal Recommendation: A Bipartite Heterogeneous Approach", 20 Mar, 2024. https://www.researchsquare.com/article/rs-4135869/v1
[4] Pierre V. Dantas, et al., "A review of state-of-the-art techniques for large language model compression," Complex Intell. Syst. 11, 407, 01 August 2025. https://link.springer.com/article/10.1007/s40747-025-02019-z#citeas
[5] Kaori Narita, et al., "Causal inference with observational data: A tutorial on propensity score analysis", The Leadership Quarterly, Volume 34, Issue 3, June 2023, 101678. https://www.sciencedirect.com/science/article/pii/S1048984323000048
[6] Chengkai Huang, et al., "Foundation Models for Recommender Systems: A Survey and New Perspectives", arXiv:2402.11143v1 [cs.IR] 17 Feb 2024. https://arxiv.org/html/2402.11143v1
[7] Thennakoon Mudiyanselage Anupama Udayangani Gunathilaka, et al., "Addressing sparse data challenges in recommendation systems: A systematic review of rating estimation using sparse rating data and profile enrichment techniques", Intelligent Systems with Applications, Volume 25, March 2025, 200474. https://www.sciencedirect.com/science/article/pii/S2667305324001480
[8] Elizabeth Gómez, et al., "Enhancing recommender systems with provider fairness through preference distribution awareness", Volume 5, Issue 1, June 2025, 100311. https://www.sciencedirect.com/science/article/pii/S2667096824001009
[9] Maxim Naumov, et al., "Deep Learning Recommendation Model for Personalization and Recommendation Systems", [v1] Fri, 31 May 2019. https://arxiv.org/abs/1906.00091
[10] Maurizio Ferrari Dacrema, et al., "Are we really making much progress? A worrying analysis of recent neural recommendation approaches", 10 September 2019. https://dl.acm.org/doi/10.1145/3298689.3347058