A Hybrid Recommendation Framework for Item Proposal in Enterprise Service Management Systems: A Survey

Subramanya G M[1], Dr. Merin Meelet[2]
[1,2] *Department of Information Science and Engineering* R V College of Engineering, Bengaluru, India

***Abstract*—Intelligent item proposal is becoming an indispensi- ble capability in modern Enterprise Service Management (ESM) platforms. When agents create service orders, they must rapidly select the right parts, labor units, or service bundles from large and frequently evolving catalogs while simultaniously accounting for customer context, asset history, and organisational policy. This decision is consequential: an incorrect or incomplete selection leads to rework, customer dissatisfaction, and increased resolu- tion time. Despite growing interest in recommendation systems for enterprise use, existing approaches remain fragmented, ad- dressing individual signals such as historical order data, curated product lists, and machine learning inference in isolation. Very few works offer a unified view of how these signals can be combined in a governance-aware, explainable, and operationally robust manner.**

This paper presents a structured survey and synthesis of recommendation methods relevant to item proposal in enterprise service workflows. We examine four core proposal source families – AI-based contextual inference, historical order retreival, prod- uct list constraints, and quotation memory signals – alongside a fifth extension channel for external rule or partner hooks. We introduce a generalized hybrid reference architecture that fuses outputs from all signal sources and evaluate the trade-offs across recommendation accuracy, latency, interpretability, and policy compliance. Our comparative analysis of representative studies reveals consistent gaps in confidence calibration, benchmark standardisation, and user-centric evaluation methodology. The survey concludes with a research agenda that adresses these open problems and positions the hybrid model as a practical default for enterprise deployment.

***Index Terms*—enterprise service management, hybrid recom- mendation, item proposal, decision support systems, case-based reasoning, explainable AI, collaborative filtering, survey**

I. INTRODUCTION

Managing service operations at scale is a challange that enterprises have grappled with for decades. Whether it is dispatching a field technician, handling a warranty claim, or fulfilling a spare-parts request, the underlying workflow almost always involves selecting appropriate items from a catalog and attaching them to a service order. In legacy environments, this selection was done manually by experienced agents who carried institutional knowledge accumulated over years of hands-on practice. The problem is that this kind of tacit knowledge does not scale well: agents turn over, catalogs grow larger, and the volume of service requests continues to rise with organizational expansion [19].

Modern ESM platforms partially address this by providing search and filter tools, but search places the cognitive burden back on the agent, who must know what to look for in the first place. What is increasingly needed is an intelligent proposal layer that narrows the decision space proactively, presenting the most relevant items before the agent even has to ask. This kind of assistive intelligence is now considered a core feature expectation in enterprise software rather then a premium add- on [21].

Recommendation systems have proved transformative in e- commerce [9], media streaming [5], and knowledge manage- ment [7]. The question facing the enterprise service commu- nity is not whether recommendation is useful, but rather how to design it responsibly given enterprise-specific constraints: strict data governance, multi-tenant isolation, product eligi- bility rules, liability for incorrect suggestions, and the need for traceable decision support. These constraints make the enterprise problem qualitatively harder than the well-studied consumer recommendation problem [20].

This survey has a very practical motivation: many organiza- tions are building item proposal features for the first time and do not have a consolidated view of the method landscape, the known trade-offs, or the open research problems. By synthesizing the literature across recommendation paradigms, hybridization strategies, and enterprise-specific considerations, this paper aims to serve as both an accessible entry point for practitioners and a structured reference for researchers entering this space.

Concretely, our contributions are:

- A comprehensive literature survey covering algorithm families, evaluation trends, and enterprise-specific chal- lenges (Section II).
- A hybrid reference framework that integrates five signal sources with a fusion and re-ranking layer (Section III).
- A comparative analysis using normalized performance trends and qualitative positioning (Section IV).
- An evaluation framework and open research agenda for future work (Sections V and VI).

The remainder of the paper is structured as follows. Sec- tion II provides the literature survey. Section III introduces the hybrid framework. Section IV compares method families. Sections V and VI discuss evaluation and future directions. Section VII concludes.

II. LITERATURE SURVEY

A. Foundational Recommender Paradigms

The modern recommender systems field traces its roots to collaborative and content-based filtering methods developed in the 1990s and early 2000s. The landmark survey by Ado- mavicius and Tuzhilin [1] established a tripartite taxonomy of collaborative, content-based, and hybrid approaches that continues to serve as the standard reference point in the field. Ricci et al. [2] later extended this by examining context-aware and multi-criteria extensions, both of which are highly relevant in enterprise workflows where context – customer type, issue category, order phase – has strong predictive power.

What is worth noting is that enterprise service scenarios differ from consumer settings in some quite important ways. In consumer recommendation, the cost of an incorrect suggestion is generally low: a user simply ignores an unwanted movie or skips a song. In service management, an incorrectly proposed item may be ordered, stocked, or physically dispatched before the error is ever caught. This asymmetry places a premium on precision, explainability, and policy compliance rather than pure recall maximization. Singh et al. [22] further highlight that enterprise recommenders must account for organisational approval chains, which introduce latency considerations absent in consumer systems.

B. Collaborative Filtering and Its Limitations in Enterprise Contexts

Collaborative filtering (CF) methods learn from the collec- tive behaviour of users – or in enterprise terms, agents and service records. Matrix factorization techniques such as those popularized by the Netflix Prize competition [5] demonstrated impressive ranking performance on dense interaction data. However, enterprise service catalogs are characterised by long- tail distributions: a small number of items appear frequently while the vast majority appear only rarely, producing the sparse interaction matrices that CF handles poorly.

Bobadilla et al. [4] survey this limitation in depth, noting that sparsity, scalability, and cold-start remain the three dom- inant failure modes for CF in production environments. He et al. [14] proposed neural collaborative filtering as a remedy, replacing the inner product of latent factors with a non-linear neural network to better capture complex user-item interaction patterns. In the enterprise item proposal context, cold-start is especially problematic: new products are introduced regularly, new customers arrive without service history, and new service types emerge with organisational change. Purely CF-based approaches thus require complementary signals to maintain useful recall across the full catalog.

C. Content-Based and Knowledge-Aware Methods

Content-based filtering sidesteps the sparsity problem by exploiting item attributes and context features rather than interaction history. In an enterprise service context, relevant features include product taxonomy, asset compatibility flags, symptom-to-solution mappings, and natural language descrip- tions from service tickets. Zhang et al. [11] survey deep learn- ing approaches that use neural text encoders to embed these features, enabling similarity-based retreival that generalizes to unseen items more reliably than traditional keyword matching. Knowledge-aware methods go further by incorporating structured domain knowledge such as ontologies, eligibility matrices, and constraint graphs [7]. These methods are par- ticularly valuable for enforcing compatibility requirements: a proposed spare part must be compatible with the registered asset's make and model, and this kind of hard constraint cannot be reliably learnt from statistical patterns alone. Guo et al.

[19] demonstrated that combining feature interactions from structured knowledge with deep neural representations yields substantial gains in industrial recommendation scenarios, an insight directly applicable to the ESM setting.

D. Case-Based Reasoning as Historical Memory

Case-based reasoning (CBR) has a long history in field service and technical support applications [10]. The core idea is intuitive: similar past situations tend to have similar solutions. When a new service order arrives for a particular asset type and failure description, the system retreives the most similar historical orders and proposes the items that were used to resolve them.

CBR has several attractive properties for enterprise use. The reasoning process is directly traceable to past cases, which gives it high explainability. The approach requires no explicit training in the machine learning sense, and it naturally encapsulates institutional knowledge accumulated over years of service history. Shi et al. [15] point out that retrieval-based methods have a natural advantage in cold-start avoidance at the item level, since they retrieve based on contextual similarity rather then item popularity. The main weakness of CBR is sen- sitivity to case base quality and the need for carefully designed similarity metrics across mixed structured and unstructured fields. When past cases are sparse or unrepresentative, retrieval quality degrades accordingly, which is why modern hybrid designs position CBR as one signal among several rather than a standalone engine.

E. Hybrid Recommendation: Theory and Practice

Burke [3] identified seven hybridisation strategies for rec- ommender systems: weighted blending, switching, mixed pre- sentation, feature augmentation, feature combination, cascade filtering, and meta-learning. In enterprise item proposal, the most commonly effective patterns are cascade and weighted blending. Cascade designs use hard constraints – product list rules, eligibility filters – to reduce the candidate space before statistical ranking is applied, ensuring that the final proposal set is always policy-compliant regardless of model output. Weighted blending then merges confidence scores from AI models, historical retrieval, and quotation signals to produce a unified ranked list.

Jannach et al. [8] provide extensive discussion of industrial hybrid deployment, noting that the engineering challenge of keeping multiple models in sync with catalog and policy changes is often more demanding than the modeling challenge itself. This observation resonates strongly in enterprise ESM contexts where product catalogs are updated continuously and organisational policies evolve with contractual and regulatory changes. Aggarwal [6] further notes that hybrid systems must manage not just accuracy but also diversity and serendipity, properties that matter even in utilitarian enterprise settings where agents benefit from being exposed to relevant but non- obvious item combinations.

F. Deep Learning Approaches

The emergence of large-scale neural language models has significantly expanded the scope of AI-based item proposal. Pre-trained transformers can encode problem descriptions and item descriptors into shared embedding spaces, enabling se- mantic similarity search that goes beyond keyword overlap [11]. In service scenarios, this means that a problem descrip- tion mentioning "cooling failure" can produce semantically relevant proposals for heat management components even when exact terminology differs between the complaint and the catalog.

Sun et al. [17] showed that bidirectional transformer en- coders applied to sequential interaction histories significantly outperform unidirectional approaches, with gains that are especially pronounced when the sequence of items in a service order matters for predicting subsequent item needs. Chen et al. [16] demonstrated similar gains in e-commerce through behaviour sequence modelling, a technique transferable to the enterprise service domain where order line sequencing is observable.

Despite this expressive power, deep models face real en- terprise deployment obstacles. Inference latency at scale, the difficulty of integrating dynamic catalog changes without full re-training, and limited native support for structured business rules all require careful engineering. Naumov et al. [18] discuss scalable deployment architectures for deep recommen- dation models, which are directly relevant to ESM deploy- ments where sub-second response times are a hard operational requirement. More critically, deep models are generally harder to explain to an agent who is wondering why a particular item was suggested.

G. Explainability and Responsible Recommendation

The explainability dimension of recommendation is receiv- ing growing attention, especially in regulated and high-stakes operational contexts. Abdollahpouri et al. [12] examine bias and fairness in recommender re-ranking, noting that popularity bias can cause recommenders to systematically over-propose common items while ignoring technically correct but less frequently used alternatives. This is a real concern in enterprise settings where unusual but correct item combinations exist in the long tail of the catalog.

Wang et al. [13] survey fairness-aware recommendation more broadly, covering both provider-side and consumer-side fairness notions. For enterprise item proposal, a somewhat different form of accountability is important: can the system explain each proposed item in terms that an agent understands and can actually verify against their own experience? The ev- idence points to source-level traceability as the most practical mechanism – phrases like “This item is proposed because it was used in 14 similar past orders” or “This item is required by product list rule PL-0042” do not require model interpretability in any strict mathematical sense. They simply expose which signal source contributed the proposal, which is sufficient for agent trust in practise.

H. Evaluation Methods and Gaps

Despite the large volume of work on recommender systems, evaluation in enterprise contexts remains somewhat ad hoc. Most academic studies report Precision@K and Recall@K on offline held-out splits, which are informative but incomplete measures. They do not capture agent acceptance behaviour, business impact of incorrect proposals, or the operational cost that incorrect suggestions impose downstream. Lu et al. [7] call for broader evaluation frameworks that measure utility (was the proposal actionable?), trust (did the agent accept without overriding?), and compliance (did the proposal respect all hard constraints throughout the interaction?). Davidson et al. [20] made similar arguments in the context of large- scale industrial recommenders, observing that offline metrics frequently fail to predict online user engagement outcomes.

Table I synthesizes key themes across the surveyed litera- ture.

I. Summary of Literature Gaps

The reviewed body of work converges on several unresolved challenges that we believe are worth restating explicitly:

- There is no publicly available benchmark for enterprise item proposal in service workflows, making cross-paper comparison unreliable.
- Confidence calibration for low-support predictions is rarely adressed in existing work.
- Latency and throughput are seldom evaluated alongside quality metrics, yet they determine production feasibility.
- Agent behaviour under recommendation – override pat- terns, feedback loops, gaming effects – remains under- studied.
- Privacy-preserving learning for multi-tenant or federated settings is largely an open area despite its practical importance.

These gaps collectively motivate the framework presented in the next section.

Table I REPRESENTATIVE LITERATURE SYNTHESIS: ENTERPRISE ITEM PROPOSAL METHODS

Theme	Representative Works	Main Strength	Key Limitation	Relevance
Foundational	[1], [2]	Clear algorithmic taxon-	Understates enterprise	Conceptual foundation
paradigms		omy	constraints	
Collaborative filtering	[4], [5], [14]	Learns from usage pat-	Sparsity and cold-start	Historical signal module
		terns		
Knowledge-aware	[7], [11], [19]	Catalog semantics, eligi-	Training and	AI proposal and product
and content-based		bility rules	maintenance cost	rules
Case-based reasoning	[10], [15]	Explainable retrieval from	Case base quality depen-	Historical order retrieval
		history	dency	
Hybrid strategies	[3], [6], [8]	Robust under diverse con-	Engineering and calibra-	Core framework design
		ditions	tion cost	
Deep sequential mod-	[16]–[18]	High expressive capacity	Latency, interpretability	AI engine component
els				
Explainability and	[12], [13]	Improved trust and gover-	Metric standardisation	Transparency roadmap
fairness		nance	immature	

III. HYBRID REFERENCE FRAMEWORK

A. Design Principles

Drawing from the surveyed evidence and practical expe- rience with enterprise deployments, we adopt five design principles for the framework:

1) Signal diversity: each source contributes a distinct evidence type; no single source should dominate by default.
2) Progressive constraint enforcement: hard constraints such as product list rules and compatibility checks are applied before final ranking to guarantee policy compliance at all times.
3) Source-level transparency: each proposed item carries metadata about which source or sources contributed it, enabling agent-facing explanations.
4) Graceful degradation: when one source is unavailable or falls below a confidence threshold, the remaining sources continue operating without exposing failure to the user.
5) Continuous maintenance: the system is designed from the outset for periodic retraining, rule updates, and catalog synchronisation rather then treating these as afterthoughts.

B. Architecture Overview

Fig. 1 illustrates the generalized architecture. The request context feeds five parallel proposal engines. Each engine produces a scored candidate list. The fusion layer normalizes and blends scores, applies deduplication, and re-ranks the combined pool. The final output is a top-K list of proposed items with the following attributes: Product ID, description, unit of measure, item category, and a source explanation string.

C. Component Descriptions

AI Proposal Engine. Encodes the incoming request context – problem text, customer segment, asset class, order type – using a pretrained language model or a structured classification network and retreives semantically similar historical items, re- ranked by a learned scoring function. The encoder can be updated periodically without affecting other engines, which simplifies lifecycle management.

Historical Order Retriever. Computes a composite simi- larity score between the current context and a corpus of past orders using weighted distance across categorical, numerical, and textual fields. Returns items from the top-N most similar historical records. This component corresponds most closely to the CBR paradigm described in Section II.

Product List Engine. Applies explicit compatibility and el- igibility rules drawn from an administrator-maintained product catalog. Acts as both a generator – items explicitly listed for the asset category – and a filter – items ineligible for the current context are removed from the merged pool regardless of their score from other engines.

Quotation Memory. Retreives commercially agreed bun- dles from historical quotations for similar customers and asset configurations. Particularly useful for recurring service contracts and preventive maintenance scenarios where standard bundles have already been negotiated and priced.

External Hook. Provides an integration point for partner systems, domain expert rules encoded externally, or regulatory compliance engines that must remain outside the core platform for data sovereignty or contractual reasons.

Fusion and Re-ranking. Merges candidate lists from all engines, normalizes heterogeneous confidence scores to a common scale, deduplicates by product identifier, and applies a final ranking that balances signal agreement, contextual relevance, and rule priority. The normalisation step is critical because different engines output scores on incompatible scales and with different calibration properties.

IV. COMPARATIVE ANALYSIS OF METHOD FAMILIES

Table II provides a structured comparison of the five method families across the dimensions most relevant to enterprise deployment.

Fig. 2 presents a normalized cross-study performance trend synthesized from the reviewed literature. Values are normal- ized to [0, 1] for comparability across heterogeneous evaluation settings and are intended to illustrate relative strengths rather

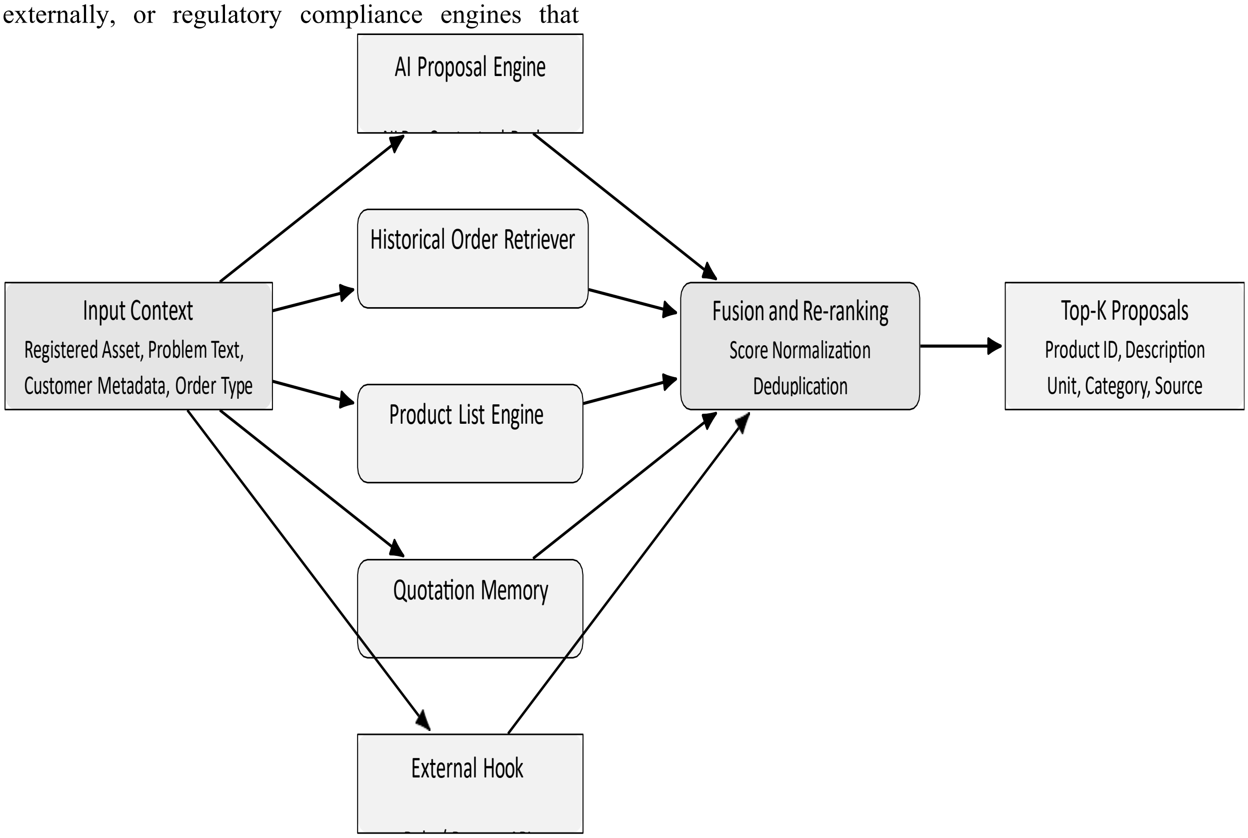

Fig. 1. Generalized hybrid reference architecture for enterprise item proposal.

TABLE II COMPARATIVE POSITIONING OF ITEM PROPOSAL METHOD FAMILIES

Family	Input Signals	Strengths	Limitations	Explainability	Recommended Role
AI contextual ranker	Problem text, meta-data, order type	Strong relevance ranking	Sparsity sensitivity, explainability cost	Medium	Primary ranker
Historical retrieval (CBR)	Past order-item inter-actions	Practitioner-aligned, explainable	Cold-start, similarity metric design	High	Memory backbone
Product list rules	Catalog, eligibility constraints	Policy compliance, predictability	Manual maintenance, low adaptability	Very High	Hard constraint filter
Quotation memory	Historical quotations, bundles	Commercial consistency	Sparse in early de-ployment	Medium	Secondary commercial signal
Hybrid fusion	All of the above	Precision, robustness, governance	Engineering and cali-bration cost	High (source traces)	Default enterprise ar-chitecture

than claim absolute superiority across all possible deployment scenarios.

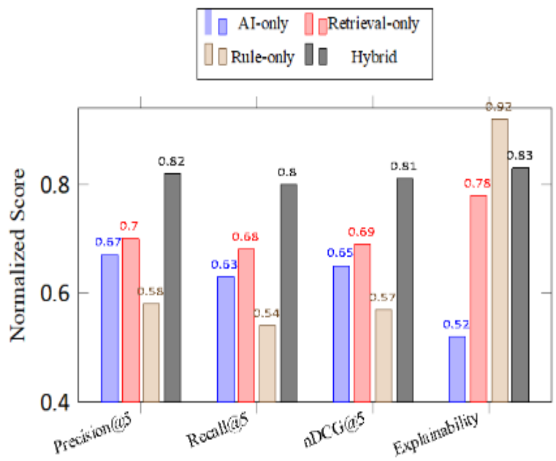

Fig. 2. Normalized cross-study trend comparison across method families (illustrative synthesis from included studies).

Fig. 3 shows the publication trend across the surveyed years, illustrating growing research interest in hybrid and explainability-oriented approaches, particularly from 2020 on- ward when transformer-based models became practical for industrial deployment.

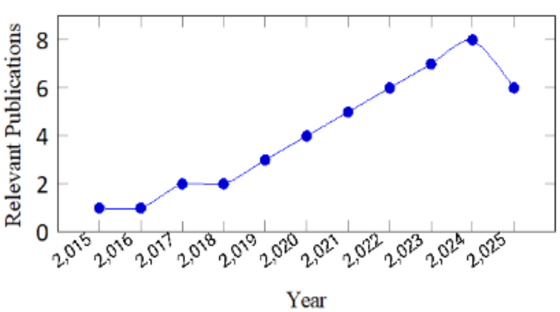

Fig. 3. Trend of relevant publications on hybrid and enterprise recommenda- tion (2015–2025, from included studies).

V. EVALUATION FRAMEWORK

A robust evaluation for enterprise item proposal should extend well beyond standard offline ranking metrics. We propose a four-tier evaluation framework that reflects the multi- dimensional nature of enterprise deployment success.

A. *Offline Ranking Quality*

Standard ranking metrics applied to a held-out evaluation split provide a necessary baseline:

- Precision@K and Recall@K for top-K candidate quality.
- Normalized Discounted Cumulative Gain (nDCG@K) for rank-position sensitivity.
- Coverage: fraction of the catalog that the system can meaningfully propose at least occasionally.
- Novelty and diversity to capture exposure beyond the most common items, which tends to be where real value lies.

B. *Operational Performance*

Operational metrics are often overlooked in academic eval- uations but are critical for deployment:

- End-to-end response latency at the 50th, 90th, and 99th percentiles.
- Throughput under peak load conditions, particularly im- portant in large service centers.
- Fallback rate: frequency with which one or more source engines produce no candidates at all.
- Availability under catalog update events, which should not cause service interruptions.

C. *Governance and Compliance*

- Policy-violation rate: how often the final proposal set contains an ineligible item that should have been filtered.
- Explanation completeness: fraction of proposals

accom- panied by a traceable source attribution.
- Override rate decomposed by source (which engine's pro- posals are most frequently overridden by human agents).

D. Human-Centered Evaluation

- Acceptance rate: fraction of proposals accepted without any modification by the agent.
- Time-to-complete: agent time to finalise the item list with versus without recommendation support.
- Perceived usefulness measured through structured agent surveys administered at regular intervals.
- Trust calibration: whether agent-reported confidence in the system correlates with actual selection accuracy over time.

VI. RESEARCH AGENDA AND OPEN CHALLENGES

A. Benchmark Standardization

The absence of a shared benchmark for enterprise item proposal is arguably the most pressing structural gap in this research area. Without it, comparing methods across papers is unreliable and claims of improvement cannot be indepen- dently verified. Future work should produce a synthetic or appropriately anonymized public dataset capturing the key features of enterprise service workflows: multi-source item candidates, sparse interaction history, catalog versioning, and mixed-attribute order context. Such a dataset would serve the research community in much the same way that the MovieLens datasets have served the broader recommender systems field.

B. Confidence Calibration

Learned recommenders often produce poorly calibrated confidence scores: a score of 0.8 does not reliably mean 80% probability of agent acceptance in any meaningful sense. For enterprise deployers, poorly calibrated scores are directly harmful because they inform ranking, explanation prominence, and threshold-based fallback behaviour. Calibration methods from probabilistic forecasting and structured prediction should be adapted to the recommendation setting, and evaluation protocols should explicitly measure calibration error alongside ranking quality.

C. Privacy-Preserving and Federated Learning

Multi-tenant service platforms face strict requirements on data isolation between customer accounts. Yet the items proposed for one tenant can benefit from generic patterns learned across the broader service population – a real tension. Federated learning approaches and privacy-preserving recom- mendation techniques offer a promising path forward, but their application to the catalog recommendation setting – with its dynamic items, structured constraints, and heterogeneous feature types – remains largely unexplored in the published literature.

D. Online Adaptation and Catalog Drift

Catalogs change continuously: items are added, deprecated, repriced, and recategorized on timescales of days to weeks. Recommendation models trained on historical data will grad- ually drift in quality unless they can be updated incrementally as the catalog evolves. Online learning methods that handle new items without requiring full retraining and that can absorb agent feedback in near-real-time represent a clear research priority for the field.

E. Agentic and Conversational Proposal

Recent advances in large language models are opening up more interactive forms of recommendation where the system engages in a back-and-forth dialogue to refine proposals based on agent responses. For enterprise item proposal, this raises a number of interesting questions around context management across turns, compliance enforcement in multi-turn settings, and the interface between a conversational front end and the structured rule engines that must ultimately govern which items are permissible. This is perhaps the most open-ended research direction identified in this survey.

VII. CONCLUSION

This paper has presented a structured survey and synthesis of recommendation methods for item proposal in enterprise service management systems. The central finding is that no single method family satisfies the full set of enterprise re- quirements: AI models deliver strong ranking quality but are difficult to explain and maintain; historical retrieval is trans- parent and practical but suffers under cold-start and catalog evolution; rule-based engines provide compliance guarantees but lack adaptability; and quotation or external-hook signals cover important but relatively narrow scenarios.

The hybrid reference framework introduced here adresses these limitations by combining all five source families through a fusion and re-ranking layer, enforcing policy constraints progressively, and providing source-level explanation for each suggestion. The comparative evidence from the surveyed lit- erature, while heterogeneous in its evaluation settings, consis- tently favours hybrid designs on the combined dimension of ranking quality and governance.

Looking ahead, the field genuinely needs better benchmarks, calibrated confidence scoring, privacy-aware architectures, and richer human-centered evaluation methods. Progress on each of these fronts will be prerequisite for enterprise item proposal systems to transition from useful conveniences to fully trusted operational tools. We hope this survey provides both a clear map of the existing landscape and a concrete roadmap for the work that remains to be done.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, Jun. 2005.

[2] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender Systems Handbook*. Springer, Boston, MA, 2011, pp. 1–35.

[3] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331– 370, Nov. 2002.

[4] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutierrez, "Recommender systems survey," *Knowledge-Based Systems*, vol. 46, pp. 109–132, Jul. 2013.

[5] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[6] C. C. Aggarwal, *Recommender Systems: The Textbook*. Springer Inter- national Publishing, 2016.

[7] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: A survey," *Decision Support Systems*, vol. 74, pp. 12–32, Jun. 2015.

[8] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*. Cambridge University Press, 2016.

[9] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, Jan. 2003.

[10] K. Bradley and B. Smyth, "Case-based user profiling for content person- alisation," in *Adaptive Hypermedia and Adaptive Web-Based Systems*, Springer, Berlin, 2000, pp. 62–72.

[11] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning-based rec- ommender system: A survey and new perspectives," *ACM Computing Surveys*, vol. 52, no. 1, pp. 1–38, Feb. 2020.

[12] H. Abdollahpouri, R. Burke, and B. Mobasher, "Managing popularity bias in recommender systems with personalized re-ranking," in *Proc. 32nd FLAIRS Conference*, Sarasota, FL, USA, May 2019, pp. 413–418.

[13] Y. Wang, W. Ma, M. Zhang, Y. Liu, and S. Ma, "A survey on the fairness of recommender systems," *ACM Transactions on Information Systems*, vol. 40, no. 3, pp. 1–43, Apr. 2021.

[14] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th International Conference on World Wide Web (WWW)*, Perth, Australia, Apr. 2017, pp. 173–182.

[15] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Computing Surveys*, vol. 47, no. 1, pp. 1–45, Jul. 2014.

[16] Q. Chen, H. Zhao, W. Li, P. Huang, and W. Ou, "Behavior sequence transformer for e-commerce recommendation in Alibaba," in *Proc. 1st International Workshop on Deep Learning Practice for High- Dimensional Sparse Data (DLP-KDD)*, Anchorage, AK, USA, Aug. 2019.

[17] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "BERT4Rec: Sequential

recommendation with bidirectional encoder representations from transformer," in *Proc. 28th ACM International Conference on Information and Knowledge Management (CIKM)*, Beijing, China, Nov. 2019, pp. 1441–1450.

[18] M. Naumov *et al.*, "Deep learning recommendation model for personal- ization and recommendation systems," *arXiv preprint arXiv:1906.00091*, 2019.

[19] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "DeepFM: A factorization- machine based neural network for CTR prediction," in *Proc. 26th Inter- national Joint Conference on Artificial Intelligence (IJCAI)*, Melbourne, Australia, Aug. 2017, pp. 1725–1731.

[20] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston, and D. Sampath, "The YouTube video recommendation system," in *Proc. 4th ACM Conference on Recommender Systems (RecSys)*, Barcelona, Spain, Sep. 2010, pp. 293–296.

[21] M. Zanker, M. Rook, and D. Jannach, "Measuring the impact of online personalisation: Past, present and future," *International Journal of Human-Computer Studies*, vol. 131, pp. 160–168, Nov. 2019.

[22] P. K. Singh and A. Sharma, "Enterprise recommender systems: Chal- lenges, opportunities, and emerging directions," *Expert Systems with Applications*, vol. 144, p. 113090, Apr. 2020.