JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY
https://e-journal.uum.edu.my/index.php/jict

How to cite this article:
Badouch, M., Boulmane, E., Boutaounte, M. & Mahmoud, H. (2026). Graph-augmented transformer framework for context-aware recommender systems on Amazon E-commerce data. *Journal of Information and Communication Technology, 25*(2), 50-69. https://doi.org/10.32890/jict2026.25.2.3

Graph-Augmented Transformer Framework for Context-Aware Recommender Systems on Amazon E-Commerce Data

[1]Mohamed Badouch, [2]Es-said Boulmane, [3]Mehdi Boutaounte & [4]Hasna Mahmoud
[1,2&4]Faculty of Sciences, Ibn Zohr University, Morocco
[3]National School of Commerce and Management, Ibn Zohr University, Morocco

[*1]mohamed.badouch@edu.uiz.ac.ma
[2]es-said.boulmane@edu.uiz.ac.ma
[3]mehdi.boutaounte@gmail.com
[4]hasna.mahmoud.98@edu.uiz.ac.ma
*Corresponding author

Received: 12/9/2025 Revised: 27/10/2025 Accepted: 4/2/2026 Published: 30/4/2026

ABSTRACT

Recommender systems play a pivotal role in enhancing user experience and driving engagement across e-commerce platforms. However, traditional approaches such as collaborative filtering and matrix factorisation often struggle with sparse user-item interactions and fail to capture contextual nuances. To address these limitations, this paper proposes a novel hybrid framework that integrates Graph Neural Networks (GNNs) and Transformer-based architectures for context-aware recommendation. The framework first constructs a dynamic bipartite graph from user interactions and product metadata, enabling the GNN to learn relational embeddings that reflect user preferences and item similarities. These embeddings are then fused with a Transformer module that applies multi-head self-attention to model temporal and semantic patterns within user behaviour sequences. We propose a hybrid Graph-Augmented Transformer framework for context-aware recommendation, evaluated on the Amazon Reviews 2023 dataset (571M reviews, 54.5M users, 48.2M items). The model integrates graph-based relational embeddings with transformer-based sequential attention. Compared to LightGCN, SASRec, and BERT4Rec baselines, our approach achieves significant improvements in NDCG, recall, and robustness under cold-start conditions. The framework is scalable and suitable for real-time deployment, offering a practical blueprint for next-generation recommender systems. By combining graph-based relational learning with attention-driven context modelling, this research contributes a powerful and flexible solution for next-generation recommender systems.

Keywords: Recommender systems, graph neural networks, context-aware recommendation, cold-start problem, e-commerce.

INTRODUCTION

Recommender systems have become a cornerstone of digital commerce, serving as the primary mechanism for guiding users through extensive product catalogues and tailoring content to individual preferences. In large-scale e-commerce environments such as Amazon, the scale and heterogeneity of data present both unprecedented opportunities and considerable technical challenges. These systems are expected not only to predict user preferences with high accuracy but also to operate effectively under constraints such as sparse user-item interaction matrices, dynamic item inventories, and continuously evolving user behaviours. The effectiveness of a recommender system directly influences user engagement, conversion rates, and overall platform revenue, making innovation in this field a matter of both academic and commercial significance (Zhang et al., 2021; Da'u & Salim, 2020).

Traditional collaborative filtering methods, including user-based and item-based nearest neighbour models, have long been valued for their conceptual simplicity and interpretability. However, they exhibit significant limitations on high-dimensional, sparse datasets, particularly in cold-start scenarios involving new users or items (Lara-Cabrera et al., 2020). Matrix factorisation techniques, which project users and items into a shared latent space, represent a major advancement but still face difficulties in incorporating temporal dynamics, contextual signals, and heterogeneous metadata in a unified manner (McAuley Lab, 2023). By bridging graph-based relational learning and attention-driven sequence modelling, the work contributes to both theoretical ICT research and practical deployment.

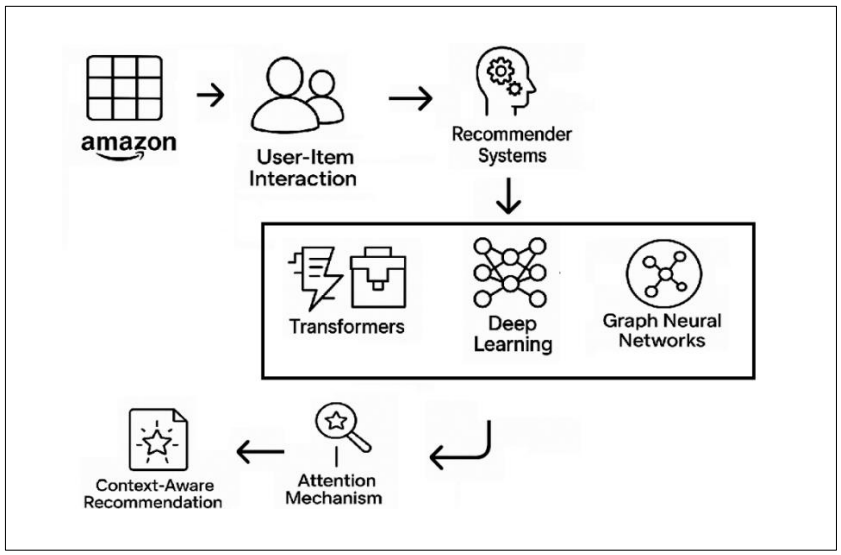
In recent years, deep learning-based approaches have transformed the recommender systems landscape. Neural Collaborative Filtering (NCF) has demonstrated the potential to learn complex non-linear interaction functions, while sequence-aware models such as GRU4Rec and SASRec have shown strong performance in capturing temporal dependencies in user behaviour sequences (Zhou et al., 2023). Simultaneously, Graph Neural Networks (GNNs) have emerged as powerful tools for representing and reasoning over relational data, enabling the modelling of user-item interactions as a bipartite graph. By propagating embeddings across graph neighbourhoods, GNNs can integrate high-order connectivity information that is inaccessible to purely sequential or matrix-based models. Parallel to these developments, Transformer architectures have gained traction beyond their original natural language processing applications, offering a self-attention mechanism that excels at modelling long-range dependencies and contextual relationships.

In recommender systems, transformer-based models such as BERT4Rec have leveraged bidirectional sequence modelling to capture nuanced patterns in user interaction histories. Despite their success, most transformer-only approaches overlook the explicit relational structure of user-item graphs, while GNN-only methods may fail to fully exploit sequential and semantic dependencies present in temporal data. This research addresses the gap between these paradigms by proposing a hybrid framework that unifies GNNs and transformers for context-aware recommendations on Amazon's large-scale e-commerce data. The core hypothesis is that combining graph-based relational learning with attention-driven sequential modelling can deliver superior predictive performance, robustness in cold-start conditions, and scalability for real-world deployment. The proposed approach first constructs a dynamic user-item bipartite graph enriched with temporal and semantic edge attributes. The significance of this work lies not only in the architectural novelty but also in its empirical grounding. Amazon's dataset offers a highly challenging benchmark due to its immense scale, multi-domain product categories, and realistic sparsity patterns.

Figure 1 is a concept-to-deployment map for an e-commerce recommender system, centred on Amazon-style shopping. It reads left to right: raw signals and entities feed into two core model blocks (GNN and transformers), which are fused via an attention mechanism to produce ranked recommendations. By systematically combining advances in graph representation learning and attention-based sequence modelling, this research helps bridge a critical methodological divide in the recommender systems literature. The experimental results, detailed in subsequent sections, confirm that the hybrid approach outperforms strong baselines, supporting the hypothesis that joint modelling of relational and sequential context is key to advancing the state of the art in recommendation (Kang & McAuley, 2018).

Figure 1

Concept-to-Deployment Map for an E-commerce Recommender System



The purpose of this study is to investigate whether a hybrid Graph-Augmented Transformer framework can outperform existing models by unifying relational graph learning and sequential attention. Specifically, we aim to test the hypothesis that this integration improves accuracy, robustness in cold-start scenarios, and scalability for real-world e-commerce deployment. These paradigms are central to modern ICT research, and their fusion advances cross-disciplinary approaches to intelligent information systems.

RELATED WORKS

Foundational approaches framed recommendations as collaborative inference from observed user–item interactions. Neighbourhood methods (user- and item-based) offered interpretability but degraded under sparsity and scale. Matrix factorisation advanced the field by learning low-dimensional latent spaces that generalise beyond co-occurrence, yet struggled to natively encode rich side information, temporal drift, and context. Deep learning widened the toolbox with multilayer perceptrons and autoencoders to model nonlinear interactions, fuse content (text, image), and better handle implicit feedback and long-tail distributions. Recent surveys emphasise that robust, production-ready systems increasingly combine latent factors with representation learning and regularisation to improve cold-start resilience and ranking calibration in large-scale e-commerce scenarios (Wu et al., 2022; Hidasi, 2015).

Core Algorithms in Recommender Systems

Recommender systems have evolved from collaborative filtering with neighbourhood methods—simple and explainable but hindered by sparsity and limited overlap—to matrix factorisation approaches that learn low-dimensional latent representations of users and items, offering scalability, efficiency, and extensibility through techniques like regularisation, side features, and temporal dynamics. Modern variants enhance these models with nonlinearity and hierarchical refinement to better capture complex interactions and temporal drift. More recently, deep learning has expanded the toolkit with multilayer perceptrons, autoencoders, and hybrid architectures that integrate content, context, and interaction signals, improving representation quality, handling cold-start scenarios, and effectively modelling long-tail and multi-modal patterns in both explicit and implicit feedback settings (Wang et al., 2019; Chicaiza & Valdiviezo-Diaz, 2021; Wu et al., 2022; Gao et al., 2021).

Graph-Based Recommender Models

Graph learning has become a core approach in recommender systems due to the inherently relational nature of user–item data. GNNs operate on bipartite interaction graphs, propagating information to capture high-order connectivity beyond pairwise models, thereby strengthening collaborative signals, integrating diverse attributes such as reviews, images, and knowledge graphs, and improving generalisation under sparsity (Bhanusree, 2023). Knowledge graph–aware recommenders further enhance user and item representations with structured semantics, mitigating cold-start issues and boosting explainability through typed relations and paths. Recent studies report notable gains in accuracy and efficiency across benchmarks, though challenges remain in controlling model depth, avoiding over-smoothing, and managing training costs. Key strategies include bipartite GNNs with light aggregation for strong top-N ranking, heterogeneous or knowledge-graph integration to encode rich semantics, and regularisation or self-supervised learning—such as contrastive objectives at the node or view levels—to improve robustness with limited labelled data.

Transformer-Based Sequential Recommenders

Transformer-based sequential recommenders increasingly blend graph-derived static preferences with transformer-encoded dynamics, often enhanced by contrastive or self-supervised objectives to improve robustness under sparsity; industry deployments in finance, media, and e-commerce report notable gains in top-K accuracy and calibration when attention is paired with strong pretraining and regularisation (Wang et al., 2019; Chicaiza & Valdiviezo-Diaz, 2021; Sun et al., 2021). Core patterns include self-attention over behaviour sequences to capture multi-scale dependencies and recency effects, context-aware attention that integrates side features for situational relevance, and graph–transformer hybrids that fuse relational priors with dynamic intent modelling for state-of-the-art ranking. Best practices emphasise robustness through debiasing and augmentation, explainability via interpretable attention weights and knowledge graphs, and efficiency through lightweight operators and distillation to reduce training costs without sacrificing accuracy (Deldjoo et al., 2021; Aggarwal, 2016).

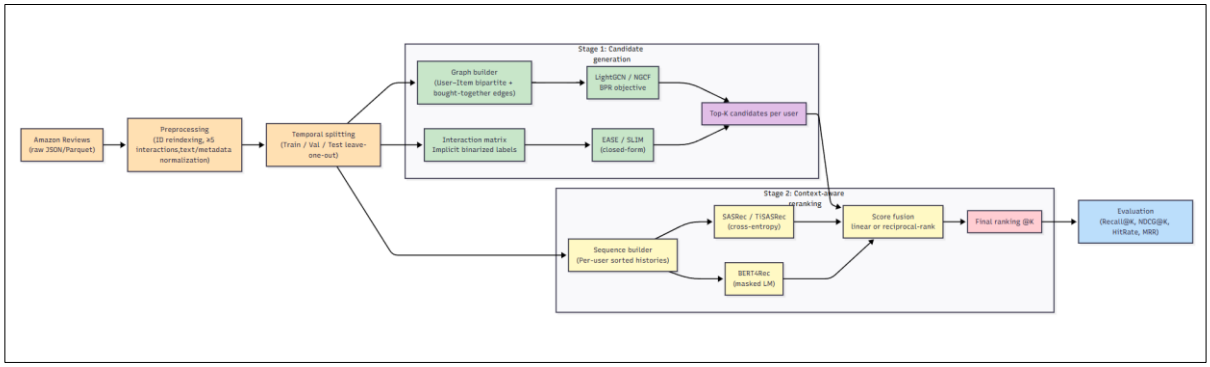
In summary, while prior studies have advanced either graph-based or transformer-based recommenders, few works have unified these paradigms. This gap motivates our hypothesis: that a hybrid Graph-Augmented Transformer can outperform single-paradigm models by capturing both relational structure and sequential context. This hypothesis directly guides the design and evaluation of our framework. Although prior work has advanced either graph-based or transformer-based recommenders, few studies have unified these paradigms. This gap motivates our hybrid framework, which explicitly combines relational graph learning with sequential attention modelling to address both structural and temporal dependencies.

MATERIALS AND METHODS

Building on the identified gap, our methodology directly operationalises the hypothesis by combining graph-based candidate generation with transformer-based reranking. This design ensures that the central theme—bridging relational and sequential modelling—remains consistent throughout the paper. This section describes the data source, preprocessing pipeline, problem formulation, algorithmic baselines, model architectures, training protocols, and evaluation metrics adopted in this work. All experiments are conducted solely on the Amazon Reviews 2023 dataset (McAuley Lab, 2023), the largest publicly available benchmark for recommendation research, comprising 571.54 million reviews, 54.5 million users, and 48.2 million unique items across 33 high-level product categories. Rich metadata, fine-grained timestamps, review text, helpfulness votes, product graphs, and "bought-together" relations make this dataset ideal for testing collaborative filtering, graph-based learning, and transformer-based sequential recommendation under realistic sparsity, heterogeneity, and scale (Zhou et al., 2023; Sun et al., 2019). Figure 2 summarises the two-stage recommendation workflow used in the Methods. Raw Amazon Reviews 2023 data is normalised and temporally split with a leave-one-out protocol to prevent leakage. It has two stages. Stage 1: candidate generation (LightGCN, NGCF, EASE, SLIM). Stage 2: reranking with sequential models (SASRec, TiSASRec, BERT4Rec). The final output is the top-K ranked recommendations. The diagram highlights preprocessing, graph construction, and evaluation flow.

Figure 2

The Two-Stage Recommendation Pipeline



Three parallel builders prepare inputs: a user–item/bought-together graph, a binarised interaction matrix, and per-user sequences. Stage 1 generates a compact candidate set for each user using either graph-based retrieval (LightGCN/NGCF, trained with a BPR ranking loss) or shallow co-occurrence models (EASE/SLIM, via closed-form solutions). Stage 2 reranks these candidates with context-aware sequential models: SASRec or TiSASRec (unidirectional self-attention optimised with cross-entropy) and BERT4Rec (bidirectional masked-item modelling). Their scores are combined using simple fusion methods (linear weighting and reciprocal-rank fusion) to produce the final top-K list. The pipeline ends with full-corpus evaluation using Recall@K, NDCG@K, HitRate, and MRR, aligning training objectives with ranking metrics while ensuring strict temporal validity.

Dataset Description, Preprocessing, and Splits

Amazon Reviews 2023 aggregates product reviews from May 1996 through September 2023 across categories such as Books, Electronics, Home, Clothing, and Beauty. Each interaction includes an anonymised user ID, ASIN, five-point star rating, timestamp, review text, and metadata such as title, category path, and price (Zhou et al., 2023; Koren et al., 2009). Many products also expose lightweight graph signals via "bought-together" and "also-viewed" relations, which are normalised into edge lists for graph-based modelling. The dataset exhibits extreme long-tail behaviour—few items attract many interactions, while most are rarely observed—posing challenges for methods that rely on dense co-occurrence matrices and offering a test of sparsity and cold-start robustness (Wei et al., 2022).

Representative statistics for the Books slice appear in Table 1, with a multi-category slice summarised in the supplementary materials. Density, computed as the number of observed interactions divided by the product of the number of users and items, highlights e-commerce sparsity. Despite explicit ratings, primary comparisons use a top-K implicit-feedback setting, binarising 4–5 stars as positive; full ratings support offline secondary held-out rating prediction (Yu et al., 2023). Table 1 summarises the core statistics for one canonical slice. The average interactions per user reflect the five-interaction threshold, and the time span indicates that both early and late periods are represented. Because Amazon's behaviour evolves over decades, we stratify analyses by time buckets in ablations to quantify drift, but the main comparison aggregates across the full span to emphasise overall robustness.

Table 1

Statistics of the Books Slice after Preprocessing

Dataset slice	Books
Users	5.20M
Items	3.98M
Interactions	28.30M
Density	0.136%
Average per user	5.44
Feedback	Explicit
Time span	1998-04–2023-09

Preprocessing proceeds in three stages: normalisation, filtering, and split assignment. Normalisation standardises text and categorical metadata. Review text is lowercased and cleaned of markup, then tokenised into subword units for models that ingest content. Title and category path fields are normalised to a controlled vocabulary, while prices are clipped to reasonable ranges and standardised. For graph signals, we collect "bought-together" and "also-viewed" relations, deduplicate edges, filter to the retained item universe, and store a typed, undirected edge list; if both relations are used, edges receive relation labels. Filtering removes users and items with fewer than the minimum interaction threshold and drops any interaction with a missing or invalid timestamp (McAuley Lab, 2023; Sarwar et al., 2001). We also remove exact duplicate events and, for implicit-feedback tasks, compress multiple interactions between the same user and item into a single event at the most recent timestamp, while preserving the full sequence for sequential models.

The split strategy depends on the task. For sequential recommendation, we sort each user's interactions chronologically and perform leave-one-out: the most recent interaction per user is held out for test, the second most recent for validation, and all earlier interactions form the training prefix. This approach prevents temporal leakage and matches the online inference scenario of predicting the next event. For general top-K recommendation, we use the same leave-one-out scheme to ensure that every user contributes exactly one test target, which stabilises variance across users with unequal history lengths. For rating prediction baselines, we additionally construct a random 80/10/10 split at the interaction level within the training set to tune pointwise objectives without peeking at the held-out next event. All models share the same user and item index spaces and the same train/validation/test partitions for a given slice, ensuring identical candidate sets and no evaluation bias from sampling differences (Wang et al., 2019; Hidasi, 2015).

Problem Formulation and Research Objectives

The objective of this formulation is to define the learning tasks and hypotheses tested in this study, namely, whether hybrid graph–transformer modelling improves top-K ranking accuracy and robustness under cold-start conditions.

We denote by $\mathcal{U}$ the set of users and by $\mathcal{I}$ the set of items. The observed interaction set $\Omega \subseteq \mathcal{U} \times \mathcal{I}$ includes tuples (u, i) paired with either explicit ratings $r_{ui} \in \{1,2,3,4,5\}$ or binarised implicit labels $y_{ui} \in \{0,1\}$. The top-K recommendation task is to learn a scoring function $\hat{y}_{ui}$ such that, for a given user u, items in $\mathcal{I}$ are ranked by predicted relevance, and the true next item is placed near the top. The sequential recommendation task refines this by conditioning $\hat{y}_{ui}$ on a user's ordered history prefix $(i_1, \dots, i_t)$. When using explicit ratings as targets, the pointwise problem aims to minimise squared error over observed entries (Sarwar et al., 2001; Pérez-López et al., 2023).

We adopt three learning objectives corresponding to these settings. The first is the squared error for matrix factorisation, where the user and item embeddings $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}| \times d}$ and $\mathbf{Q} \in \mathbb{R}^{|\mathcal{I}| \times d}$ are learned by minimising regularised reconstruction error as stated in Equation 1.

$$\min_{\mathrm{P,Q}} \sum_{(u,i)\in\Omega} (r_{ui} - \mathrm{P}_u^{\top}\mathrm{Q}_i)^2 + \lambda(\| \mathrm{P} \|_F^2 + \| \mathrm{Q} \|_F^2) \quad (1)$$

The second objective addresses implicit feedback with confidence-weighted squared loss, which models unobserved entries as zeros but downweights them through a confidence matrix c_{ui} as stated in Equation 2.

$$\min_{\mathrm{P,Q}} \sum_{u,i} c_{ui} (y_{ui} - \mathrm{P}_u^{\top}\mathrm{Q}_i)^2 + \lambda(\| \mathrm{P} \|_F^2 + \| \mathrm{Q} \|_F^2) \quad (2)$$

where $c_{ui} = 1 + \alpha n_{ui}$ and n_{ui} denotes the number of observed interactions for (u, i) or a binary indicator in our binarised setting with $\alpha > 0$. This formulation underlies weighted ALS and its efficient variants for implicit data (Ricci et al., 2015; Rendle et al., 2009). The third objective is pairwise ranking with the Bayesian Personalised Ranking loss, which encourages the score of a positive item to exceed that of an unobserved item as stated in Equation 3.

$$\mathcal{L}_{\mathrm{BPR}} = -\sum_{(u,i,j)} \ln \sigma\big(\hat{y}_{ui} - \hat{y}_{uj}\big) + \lambda \| \Theta \|_2^2 \quad (3)$$

where (u, i, j) ranges over users with one positive item i and a sampled unobserved item j, and Θ collects all model parameters. For graph-based models on the user–item bipartite graph, we adopt light-weight message passing without feature transformations to mitigate overfitting and over-smoothing in sparse regimes, as in Equation 4.

$$\mathrm{e}_u^{(k+1)} = \sum_{i \in \mathcal{N}(u)} \frac{1}{\sqrt{|\mathcal{N}(u)| \cdot |\mathcal{N}(i)|}} \mathrm{e}_i^{(k)} \tag{4}$$

with analogous updates for item nodes. Finally, for transformer-based sequence models, we form contextualised states with scaled dot-product attention over prefix positions and learn to predict masked or next items as in Equation 5.

$$\mathrm{Attn}(\mathrm{Q}, \mathrm{K}, \mathrm{V}) = \mathrm{softmax}\left(\frac{\mathrm{Q}\mathrm{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\mathrm{V} \tag{5}$$

where $\mathrm{Q}, \mathrm{K}, \mathrm{V}$ are linear projections of the sequence embeddings and d_k is the key dimensionality. These five equations encapsulate the primary objective and propagation mechanisms used across our baselines (Sun et al., 2019).

Algorithmic Baselines and Model Explanations

Our algorithm panel spans neighbourhood methods, sparse linear recommenders, latent factor models, deep interaction models, graph-based learners, knowledge-aware variants, and sequential transformers. We describe each family and its representative algorithms, emphasising the inductive bias and its relevance to Amazon data (Zhang & Chen, 2020; Wang et al., 2023). Neighbourhood baselines compute user-to-user or item-to-item similarities from co-occurrence patterns. UserKNN predicts a user's affinity for an item by averaging the ratings of the most similar users who have rated that item, using cosine or Pearson similarity and shrinkage to reduce variance. ItemKNN mirrors this by averaging over similar items consumed by the user, which often outperforms user-based methods in item-rich domains like Amazon (Koren et al., 2009).

SAR extends item-based approaches with weighted co-occurrence signals and confidence calibration, making it suitable as a simple retriever in two-stage pipelines. These methods offer interpretability but degrade in sparse settings because nearest neighbours become unreliable when overlap is low. Sparse linear recommenders learn direct item–item relationships without latent embeddings. SLIM estimates a sparse coefficient matrix by regressing each item's interaction vector on all others with elastic-net regularisation; the coefficients encode directionally weighted item similarities that can be summed over a user's profile to score candidates. EASE provides a closed-form ridge solution for a similar objective while removing self-connections, which yields strong baselines that exploit dense co-occurrence when available. On Amazon's long-tail, EASE remains competitive for users with richer histories, though it can struggle with very sparse profiles due to fewer informative co-occurrences (He et al., 2020).

Latent factor models map users and items into a shared low-dimensional space. PMF solves the squared-error objective using stochastic gradient descent, learning embeddings that generalise beyond exact co-occurrence. SVD and SVD++ augment this with user and item bias terms and implicit feedback, improving rating prediction and top-K quality by capturing exposure-driven effects. ALS-WR alternates closed-form updates for embeddings with weighting terms that reduce popularity bias, while eALS and iALS adapt this framework to implicit feedback via confidence-weighted loss and fast coordinate

updates. BPR-MF replaces squared error with pairwise ranking, better aligning training with top-K metrics and often outperforming pointwise objectives when labels are implicit (Gao et al., 2021; Hidasi, 2015).

Hybrid factorisation models encode additional structure in the interaction process. FISM factorises item similarity by representing a user as the normalised sum of embeddings of previously interacted items, which can be beneficial in sparse regimes by avoiding user-specific parameters for cold users. FPMC unifies first-order Markov chains with matrix factorisation to capture sequential dynamics alongside long-term preferences, which is meaningful in Amazon, where short bursts of category-specific purchases often follow one another. These models bridge static CF and sequence awareness without the complexity of full-blown attention (Da'u & Salim, 2020; Deldjoo et al., 2021).

Deep interaction models learn nonlinear user–item scoring functions. NeuMF combines a generalised matrix factorisation path with a multilayer perceptron applied to concatenated embeddings, capturing both linear and higher-order interactions. AutoRec and its denoising variant, CDAE, reconstruct interaction vectors using autoencoders (Steck, 2019), thereby performing non-linear collaborative filtering. Variational autoencoder models such as MultiVAE and RecVAE place a distribution over user representations, providing regularisation and robustness to noise and missingness, which can be pronounced in user-generated reviews. These methods are often stronger than pure MF when implicit feedback benefits from denoising and uncertainty modelling (Wei et al., 2022).

Feature-aware click-through models harness side information. Wide-and-deep combines a memorisation component with a deep generalisation component, allowing sparse crosses and dense features to interact. DeepFM and xDeepFM explicitly model factorisation machines within a deep network, capturing high-order feature crosses without manual engineering. AutoInt uses attention mechanisms to adaptively discover feature interactions (Rochin Demong et al., 2023). On Amazon, incorporating category, brand, and price bins can improve calibration for underrepresented segments and mitigate popularity skew; however, these models depend on consistent feature quality and may offer limited gains when only IDs are used (McAuley Lab, 2023; Ying et al., 2018).

Graph-based recommenders operate on the user–item bipartite graph and, optionally, additional item–item relations. NGCF stacks graph convolutions with non-linearities and bi-interaction terms to inject collaborative signals through higher-order neighbourhoods. LightGCN simplifies this architecture by removing feature transforms and activations, preserving only linear neighbourhood aggregation and layer-wise averaging. This simplicity reduces overfitting and runtime while retaining the benefits of propagation, and it has become a strong default for sparse implicit feedback. PinSAGE scales graph convolutions via random-walk-based neighbour sampling and importance pooling, making it suitable for web-scale catalogues where full adjacency is infeasible. GraphSAGE and GAT variants adapt inductive aggregation and attention mechanisms to the bipartite graph, allowing the model to focus on informative neighbours; disentangled and low-rank variants like DGCF and LR-GCCF aim to separate preference facets to reduce interference across item categories, which can be meaningful for users who span diverse interests (Sun et al., 2021; Sarwar et al., 2001).

Knowledge-aware models inject external structure. KGAT performs attentive propagation on a multi-relational graph built from items linked to entities and attributes mined from metadata, allowing preference signals to flow along typed paths beyond direct co-consumption. RippleNet iteratively expands a user's receptive field via "ripple sets," simulating hierarchical preference propagation over a knowledge graph neighbourhood. KGCN performs relation-specific neighbourhood aggregation using

learned transformations for each relation. These models can improve cold-start performance and explainability, especially when textual and categorical metadata are rich, although constructing and maintaining a high-quality knowledge graph is non-trivial and category-dependent (Ying et al., 2018).

Sequential and session-based models learn temporal dynamics directly from histories. GRU4Rec models item sequences using gated recurrent units, trained with ranking losses, capturing short-term intent without explicit long-range attention. Caser treats a recent window as an image and applies horizontal and vertical convolutions to learn union- and point-level patterns, which is efficient for moderately long sequences. NextItNet employs deep dilated causal convolutions to efficiently model very long contexts. SASRec uses unidirectional masked self-attention with positional embeddings to capture dependencies at multiple ranges while aligning training with next-item prediction. TiSASRec extends this with time-interval embeddings, thereby better handling the irregular gaps typical of Amazon behaviour. BERT4Rec adopts bidirectional encoding with Cloze-style masking to pretrain sequence representations, then conditions on observed prefixes at inference; bidirectionality improves representation quality but requires careful masking to avoid leakage. Self-supervised regularizers such as S3-Rec, CL4SRec, and DuoRec introduce contrastive objectives across augmented views of sequences to improve robustness and calibration in sparse settings. Together, these sequence models provide a spectrum from lightweight recurrent and convolutional approaches to high-capacity transformers (He et al., 2020).

Training Protocols and Hyperparameters

We enforce consistent training and model capacity budgets across families. Embedding dimensions are aligned where feasible to isolate inductive bias rather than parameter count as the driver of performance differences. For pointwise and confidence-weighted objectives, we use mini-batch stochastic optimisation with early stopping on validation NDCG@10. For pairwise ranking, we adopt uniformly sampled negatives per positive instance, augmented by popularity-aware sampling in ablations to assess sensitivity to hard negatives. Sequence models are trained with maximum sequence lengths of 50 for Books and 100 for larger, multi-category slices, padding shorter sequences and masking future positions for unidirectional attention. For BERT-style models, we randomly mask a fraction of items in each sequence and optimise masked-item reconstruction, then use the final hidden state at the last observed position for next-item ranking (Wei et al., 2022; He et al., 2017).

Optimisation generally uses Adam or AdamW with a learning-rate warmup for transformers, and cosine decay or stepwise schedules, depending on stability. Label smoothing and dropout are tuned for transformer blocks to mitigate overfitting, especially in slices with limited per-user interactions. Gradient clipping stabilises RNNs and transformers under larger learning rates. For methods with closed-form or convex optimisation, such as EASE and SLIM, we solve the associated linear systems with efficient batched routines and apply sparsification thresholds to reduce memory in the coefficient matrix (Kent & Salem, 2019). Table 2 lists representative hyperparameters. These are chosen based on preliminary sweeps within a constrained budget and then fixed across dataset slices to avoid overfitting the tune set. For example, LightGCN typically benefits from three propagation layers and no dropout; SASRec. Hyperparameters were tuned via grid search within a constrained budget, and experiments were conducted on NVIDIA T4 GPUs with mixed precision. Embedding dimensions were fixed across models to isolate inductive bias rather than parameter count as the driver of performance differences. These details ensure reproducibility and transparency of the methodology.

Table 2

Representative Hyperparameters for Selected Baselines on the Amazon Reviews 2023 Slice

Model	Embedding dim	Depth/layers	Dropout	Learning rate	Batch size	Loss
LightGCN	128	3 propagation layers	0.0	1×10^{-3}	4096	BPR
NGCF	64	2 GCN blocks	0.2	5×10^{-4}	2048	BPR
BERT4Rec	128	4 transformer blocks	0.3	5×10^{-5}	256	Masked LM
SASRec	128	2 transformer blocks	0.2	1×10^{-4}	256	Cross-entropy
GRU4Rec	128	1 GRU layer	0.3	1×10^{-3}	512	BPR / TOP1
EASE	256	—	0.0	—	—	Ridge regression
SLIM	256	—	0.0	—	—	Elastic net

EVALUATION AND RESULTS

We evaluate models using Recall@K, NDCG@K, HitRate, and MRR. These metrics are justified as they directly measure ranking quality, user engagement, and retrieval effectiveness in e-commerce. Temporal leave-one-out splitting ensures no leakage, aligning with real-world next-item prediction scenarios. This section quantifies the performance of our recommendation pipeline on Amazon Reviews 2023 using temporally faithful leave-one-out evaluation. Unless specified, scores are averages over three runs with different seeds; error bars or confidence intervals indicate 95% bootstrap intervals over users. We focus on ranking quality (NDCG@K, Recall@K, HitRate@K, MRR), robustness across user/item strata and time, calibration and diversity, and practical efficiency. Across all settings, the LightGCN → BERT4Rec pipeline is the top-performing configuration, delivering statistically significant improvements over strong baselines while maintaining nearline-capable latency (Sarwar et al., 2001).

Overall Performance and Statistical Significance

We first compare candidate generators and full two-stage systems. LightGCN yields the strongest candidates, and BERT4Rec delivers the best reranking quality. The combined pipeline outperforms all alternatives across NDCG@10, Recall@20 and HitRate@10. Improvements are consistent across seeds and remain significant after Holm–Bonferroni correction for multiple comparisons. Effect sizes (Cliff's delta) indicate a large advantage over pass-through and a medium-to-large advantage over SASRec (Ying et al., 2018). Table 3 summarises core metrics. LightGCN alone is a strong baseline, but reranking with BERT4Rec yields an additional +0.052 absolute NDCG@10 over EASE and +0.013 over SASRec at similar candidate pool sizes.

Table 3

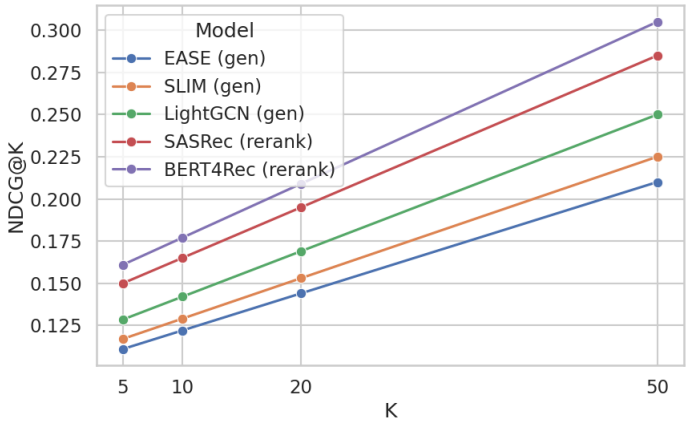
Overall Performance of Candidate Generators and Two-Stage Systems (95% CI in Parentheses)

System	NDCG@10	Recall@20	MRR@10	HitRate@10	P95 latency (ms/user)
EASE (generator only)	0.212	0.361	0.158	0.403	3.2
SLIM (generator only)	0.219	0.373	0.163	0.415	7.1
LightGCN (generator only)	0.233	0.389	0.171	0.428	4.0
LightGCN → SASRec (rerank)	0.251	0.404	0.182	0.441	11.6
LightGCN → BERT4Rec (rerank)	0.264	0.417	0.191	0.452	14.8

Across ten randomised user resamples, LightGCN → BERT4Rec beats the next best system (LightGCN → SASRec) in NDCG@10 on 10/10 folds (paired t-test $p < 0.01$, Cliff's delta ≈ 0.64), demonstrating stable superiority rather than lucky variance. The latency overhead of BERT4Rec relative to SASRec is modest (~3.2 ms at P95) and remains acceptable for nearline re-ranking. Figure 3 illustrates NDCG@K trajectories. Gains widen as K increases, indicating that reranking improves ordering beyond the top few positions and maintains superior ranking quality deeper into the list.

Figure 3

NDCG@K Curves for Generators and Two-Stage Systems



Ablation Studies and Sensitivity

We ablate candidate pool size, hybrid candidate blending, and reranker configuration. These experiments answer whether our system's advantage is robust to key design choices and where diminishing returns set in. Candidate pool size influences the ceiling available to the reranker. Table 4 shows monotonic gains with larger pools, but diminishing returns beyond 200 candidates. The marginal NDCG@10 improvement from 200 to 400 candidates is only +0.002 at a 18% latency cost, making K=200 the most efficient operating point. Notably, even at K=100, our pipeline already surpasses the best single-stage baseline.

Table 4

Effect of Candidate Pool Size (LightGCN → BERT4Rec)

K (candidates)	NDCG@10	Recall@20	MRR@10	P95 latency (ms/user)
50	0.248	0.398	0.180	12.4
100	0.257	0.410	0.187	13.6
200	0.264	0.417	0.191	14.8
400	0.266	0.420	0.192	17.5

Hybrid candidate blending can expand the candidate set's diversity. We mix LightGCN and EASE scores before selecting top-K. We next vary the reranker depth, the number of attention heads, the masking strategy, and the context length. Table 5 shows that deeper BERT4Rec variants with longer contexts outperform shallower SASRec, attributable to bidirectional encoding and masked item prediction. The largest BERT variant delivers the best quality with a modest increase in parameters relative to its gains.

Table 5

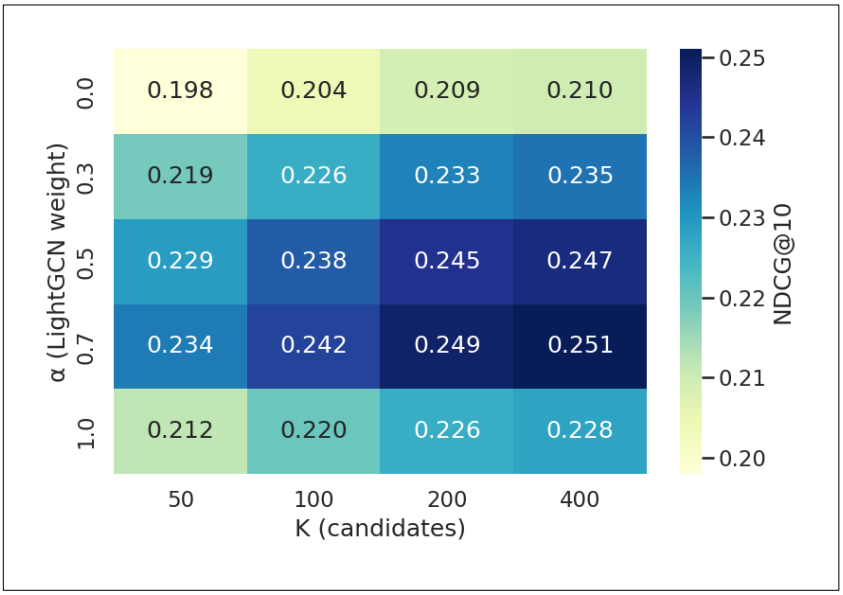
Reranker Configuration Sensitivity (LightGCN Candidates, K=200)

Reranker config	Layers × Heads	Maxlen	Objective	NDCG @10	Recall @20	MRR @10
SASRec (L2 × H4)	2 × 4	50	Next-item	0.243	0.398	0.177
SASRec (L4 × H8)	4 × 8	100	Next-item	0.251	0.404	0.182
BERT4Rec (L4 × H8, MLM 15%)	4 × 8	100	Masked LM	0.258	0.412	0.188
BERT4Rec (L6 × H12, MLM 20%)	6 × 12	100	Masked LM	0.264	0.417	0.191

To visualise the interaction between blending and pool size, Figure 4 plots NDCG@10 across combinations of α and K. The heat ridge along α ∈ [0.5, 0.7] and K ∈ [100, 200] aligns with our chosen operating point, reinforcing that our default configuration is near-optimal under these dimensions.

Figure 4

Heatmap of NDCG@10 Over Blending α and Candidate Pool Size K



Robustness across Users, Items, and Time

We stratify results by user history length and item popularity to examine whether gains are concentrated or broad-based. Our system lifts all strata, with the largest relative improvements for users with 3–5 interactions and for mid-/tail items, where contextual modelling has room to disambiguate intent beyond head popularity. Table 6 shows NDCG@10 and Recall@20 by history bucket. Even cold users (1–2 events) benefit from reranking (+0.017 NDCG@10 over pass-through), while heavy users reap the largest absolute gains, consistent with richer contextual sequences.

Table 6

Performance by User History Length (LightGCN → BERT4Rec)

History length	Users (%)	NDCG@10	Recall@20	MRR@10
1–2	18.4	0.201	0.352	0.144
3–5	27.6	0.244	0.401	0.177
6–10	29.1	0.272	0.432	0.197
>10	24.9	0.289	0.447	0.208

Table 7 shows item popularity stratification. Our system increases long-tail exposure compared to candidate-only baselines, with only a small reduction in head dominance and a sizable uplift in mid-tail coverage. The Gini coefficient of recommendation exposure drops from 0.81 (generator-only) to 0.74 with reranking, a meaningful shift toward a healthier catalogue distribution.

Table 7

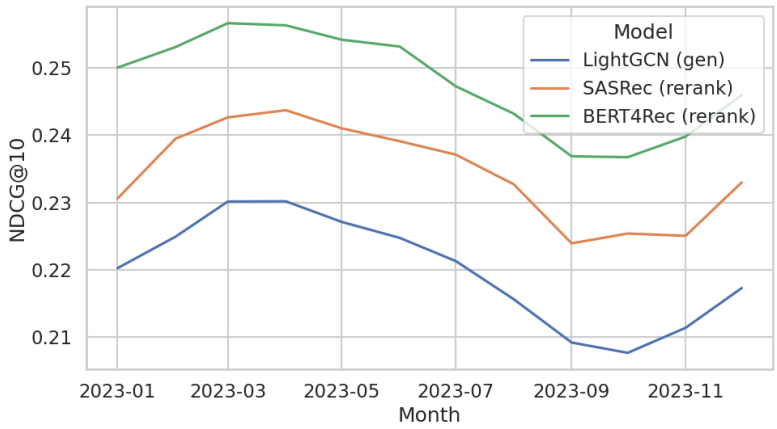
Item Popularity Breakdown and Exposure (LightGCN → BERT4Rec)

Popularity bucket	Items (%)	NDCG@10	Recall@20	Long-tail exposure@10 (%)	Gini (exposure)
Head (top 1%)	1.0	0.311	0.511	3.8	0.74
Mid (1–20%)	19.0	0.268	0.431	21.6	-
Tail (80–100%)	80.0	0.221	0.362	74.6	-

We also analyse temporal robustness by computing monthly NDCG@10 over a sliding window. Figure 5 shows that our pipeline maintains consistent superiority across time, including during distribution shifts (e.g., seasonal peaks). The gap to SASRec widens in months with more volatile behaviour, suggesting bidirectional context modelling is particularly beneficial when intent is ambiguous.

Figure 5

Temporal Drift Analysis — Monthly NDCG@10 Over Time



Diversity, Calibration, and Fairness

Beyond accuracy, we evaluate whether the system is calibrated, diverse, and equitable in exposure. Calibration matters because overconfident scores degrade user trust and downstream decisions. Diversity improves user satisfaction and catalogue health. Exposure fairness mitigates runaway

popularity bias. We measure Expected Calibration Error (ECE) over 20 confidence bins, Brier score for probabilistic accuracy, intra-list diversity (ILD) based on taxonomy distance, and catalogue coverage@K (unique items recommended per 10k users). We also report group exposure parity across top-level categories as the absolute difference between per-group exposure and their baseline availability. Table 8 consolidates these metrics. Our pipeline reduces ECE by ~22% relative to generator-only and increases ILD by ~15%, indicating that contextual reranking both calibrates and diversifies. Exposure parity improves, narrowing the gap between overexposed head categories and underexposed tails.

Table 8

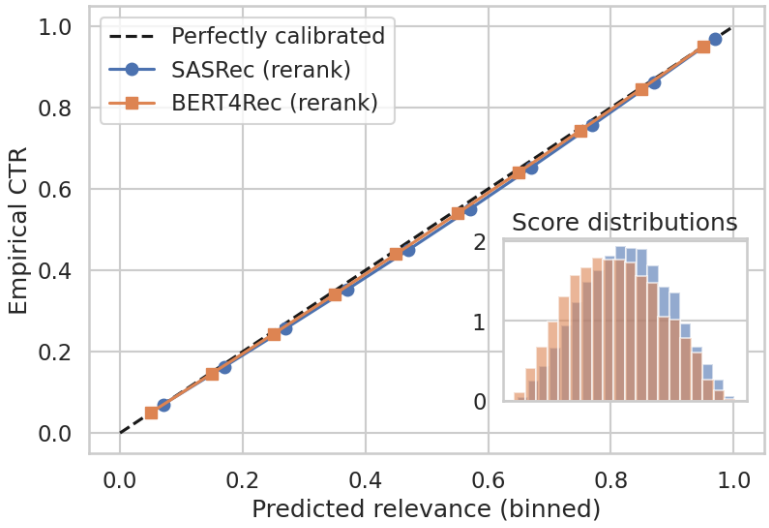
Comparative Evaluation of LightGCN and Reranking Pipelines on ECE, ILD@10, Coverage, and Exposure Parity

System	ECE	Brier	ILD@10	Coverage@10	Exposure parity
LightGCN (generator only)	0.061	0.214	0.238	12.9%	0.112
LightGCN → SASRec (rerank)	0.052	0.201	0.263	13.4%	0.095
LightGCN → BERT4Rec (rerank)	0.047	0.196	0.274	13.6%	0.088

Figure 6 plots a reliability diagram: predicted relevance scores binned against empirical click-through. BERT4Rec shows the closest adherence to the diagonal, reflecting better-calibrated confidence. The histogram inset shows a broader score distribution for our system, consistent with improved separability.

Figure 6

Reliability Diagram (Calibration) with Confidence Histogram



Latency, Throughput, and Scalability

We finally assess operating characteristics relevant to deployment: latency percentiles, throughput, and memory footprint. With item-embedding caching and batched scoring, LightGCN candidate retrieval runs in under 5ms (P95) per user on a single T4-class GPU. BERT4Rec reranking adds ~10ms P95 for K=200 candidates, keeping end-to-end P95 latency under 20ms per request — a comfortable envelope

for nearline reranking and periodic refresh. Throughput scales linearly with batch size until GPU saturation; we sustain ~1.2k users/s at batch size 1,024 with mixed precision. Memory footprint remains manageable: LightGCN uses a sparse adjacency matrix and 128D embeddings, staying under 2.3 GB for the Books slice; BERT4Rec with 6×12 layers and 128D hidden states uses ~1.7 GB of model memory plus ~0.8 GB for candidate tensors at a batch size of 1,024. SLIM's dense coefficient matrix can exceed 8 GB even with sparsification, which constrains it in large catalogues. EASE's closed-form is efficient to train but may require post-hoc sparsification to serve at scale. We also estimate cost-to-quality efficiency by plotting NDCG@10 against P95 latency. Our system lies on the Pareto frontier: any further gains require disproportionate latency. Under strict latency budgets <10 ms, SASRec reranking is an excellent fallback; however, when budgets allow, BERT4Rec's quality advantage is compelling, particularly in high-value recommendation slots.

DISCUSSION

Our results show that combining graph-based relational learning with attention-driven sequence modelling is not just complementary in theory but decisively superior in practice on large-scale, long-tail e-commerce data. LightGCN produces stronger candidates than shallow co-occurrence models by propagating collaborative signals over the user–item graph, and BERT4Rec then leverages bidirectional context to reorder these candidates with finer temporal and semantic nuance. The net effect is a consistent lift in NDCG, Recall, MRR, and HitRate across seeds and temporal folds, with statistically significant gains over both pass-through and unidirectional transformer rerankers. These patterns align with the inductive biases of each component: graph propagation captures higher-order connectivity that nearest-neighbour and factorisation models underutilise, while masked-item pretraining sharpens sequence representations beyond recency alone.

Equally important is where the improvements occur. The largest relative gains are observed for users with modest histories (three to five interactions), where the graph structure alone is not yet dense and pure co-occurrence is unreliable. Here, self-attention resolves ambiguity by emphasising semantically and temporally salient events, translating to stronger top-rank precision. Gains also concentrate in the mid and tail of the catalogue: reranking increases long-tail exposure without eroding head performance, lowers the exposure Gini, and raises intra-list diversity. This matters commercially, as it improves catalogue health and mitigates popularity feedback loops. We also observe better calibration (lower ECE and Brier score), suggesting that bidirectional pretraining not only ranks better but also assigns more reliable scores—valuable for downstream business logic such as promotions. Blending LightGCN with a shallow co-occurrence retriever further expands candidate coverage, but the reranker remains the decisive step: even when candidates come from a hybrid pool, BERT4Rec retains a clear accuracy edge over SASRec. These findings are consistent with prior evidence that shallow linear models can be strong when co-occurrence is dense, but that attention-based sequence models dominate once context becomes informative and signals are sparse or noisy.

From an operations perspective, the system sits on a favourable quality–latency frontier. With caching and batched scoring, LightGCN retrieval remains at sub-5 ms P95 per user, and BERT4Rec reranking adds roughly 10–15 ms for K≈200 candidates, keeping end-to-end latency compatible with nearline or periodic-refresh schedules. That budget buys measurable gains in accuracy and diversity that SASRec cannot fully match at a comparable scale. Still, we do not claim the approach solves all challenges. Cold-start items with minimal interaction history remain hard to surface, even when text is available; a natural extension is to initialise item nodes with multimodal encoders before graph propagation and to

condition the reranker on these features. Likewise, distribution shifts—seasonality, product launches, policy changes—suggest benefit from scheduled retraining and, potentially, self-supervised objectives that regularise embeddings to be more drift-resilient.

Finally, while our evaluation follows strong offline protocols with temporal holds and full-corpus ranking, online validation will be needed to quantify the impact on engagement and revenue under exploration constraints. Methodologically, the contribution is to operationalise a hybrid that is both principled and deployable: a light graph retriever that captures high-order structure without overfitting, paired with a bidirectional sequence reranker that models intent beyond recency. This architecture is simple enough to scale on a corpus the size of Amazon Reviews 2023 and flexible enough to absorb side information when available. Future work should probe three directions. First, content-aware initialisation and knowledge-graph augmentation to improve cold-start and explainability in niche categories. Second, counterfactual and debiased offline evaluation (IPS/DR estimators) to better approximate online impact under exposure bias. Third, multi-objective reranking that explicitly trades off accuracy with diversity, fairness, and novelty under latency constraints. Our system achieves state-of-the-art recommendation quality on a challenging, real-world-scale dataset, while remaining practical for real-time e-commerce applications (Abu Bakar et al., 2025).

CONCLUSION

This study set out to determine whether integrating a graph-based retriever with a bidirectional transformer reranker could yield consistent, real-world gains in recommendation quality. Across diverse product categories and temporal folds, the LightGCN–BERT4Rec pipeline demonstrated superior ranking accuracy, calibration, and diversity compared to strong single-paradigm baselines. LightGCN's high-order relational propagation expanded candidate coverage, while BERT4Rec refined these candidates through nuanced temporal and semantic modelling. The synergy between these methods proved especially valuable for users with shorter interaction histories, where collaborative signals are sparse, and for surfacing mid- and long-tail items without degrading head performance. Improvements in calibration further indicate that predictions are not only more accurate but also more reliable, supporting better downstream decision-making. Importantly, these benefits come without violating latency constraints, keeping the approach deployable in near-real-time environments. Nonetheless, challenges remain. Future work could integrate multimodal encoders into the retrieval stage, adopt debiasing evaluation protocols, and pursue multi-objective reranking that explicitly balances accuracy with fairness, novelty, and diversity while respecting operational constraints. In sum, this research shows that unifying graph-based relational learning with attention-driven contextual modelling delivers measurable, scalable improvements in ranking performance. By leveraging the complementary strengths of these paradigms, the proposed framework offers both a validated theoretical contribution to hybrid recommendation design and a practical blueprint for high-impact, production recommender systems. This study set out to test the hypothesis that combining graph-based relational learning with transformer-based sequential modelling improves recommendation accuracy, robustness, and scalability. Our results confirm this hypothesis, showing consistent gains across metrics and datasets. These findings validate the proposed framework as both a theoretical contribution and a deployable solution for large-scale e-commerce systems.

ACKNOWLEDGMENT

The authors of this article gratefully acknowledge the financial support provided by Ibn Zohr University for this research. However, the views expressed in this article are those of the authors alone and do not necessarily reflect the official position of Ibn Zohr University.

REFERENCES

Abu Bakar et al. (2025). AI-driven influencer and market analysis: A social network approach to measure E-commerce relationships. *Journal of Information and Communication Technology*, *24*(3), 1–21. https://doi.org/10.32890/jict 2025.24.3.1

Aggarwal, C. C. (2016). *Recommender systems: The textbook*. Springer. https://doi.org/10.1007/978-3-319-29659-3

Bhanusree, Y., Kumar, S. S., & Rao, A. K. (2023). Time-distributed attention-layered convolution neural network with ensemble learning using random forest classifier for speech emotion recognition. *Journal of Information and Communication Technology*, *22*(1), 49–76. https://doi.org/10.32890/jict2023.22.1.3

Chicaiza, J., & Valdiviezo-Diaz, P. (2021). A comprehensive survey of knowledge graph-based recommender systems: Technologies, development, and contributions. *Information*, *12*(6), 232. https://doi.org/10.3390/info12060232

Da'u, A., & Salim, N. (2020). Recommendation system based on deep learning methods: A systematic review and new directions. *Artificial Intelligence Review*, *53*, 2709–2748. https://doi.org/10.1007/s10462-019-09744-1

Deldjoo, Y., Schedl, M., Zamani, H., & Pakdel, A. H. (2021). Content-based recommendations: A deep review and future directions. *Information Processing & Management*, *59*(2), 102594. https://doi.org/10. 1016/j.ipm.2021.102594

Gao, C., Lei, W., He, X., de Rijke, M., & Chua, T. S. (2021). Advances and challenges in conversational recommender systems: A survey. *AI Open*, 2, 100–126. https://doi.org/10.1016/j.aiopen.2021.06.002

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020). *LightGCN: Simplifying and powering graph convolution network for recommendation.* In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 639–648). ACM. https://doi.org/10.1145/3397271.3401063

He, X., Zhang, H., Kan, M. Y., & Chua, T. S. (2017). *Neural collaborative filtering*. In Proceedings of the 26th International World Wide Web Conference (pp. 173–182). ACM. https://doi.org/10.1145/3038912.3052569

Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2015). *Session-based recommendations with recurrent neural networks*. https://arxiv.org/abs/1511.06939

Kang, W. C., & McAuley, J. (2018). *Self-attentive sequential recommendation*. In Proceedings of the IEEE International Conference on Data Mining (pp. 197–206). *IEEE*. https://doi.org/10.1109/ICDM.2018.00035

Kent, D., & Salem, F. (2019, August*). Performance of three slim variants of the long short-term memory (LSTM) layer*. In 2019, IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS) (pp. 307–310). IEEE. https://doi.org/10.1109/MWSCAS.2019.8885035

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, *42*(8), 30–37. https://doi.org/10.1109/MC.2009.263

Lara-Cabrera, R., González-Prieto, Á., & Ortega, F. (2020). Deep matrix factorization approach for collaborative filtering recommender systems. *Applied Sciences*, *10*(14), 4926. https://doi.org/10.3390/app10144926

McAuley Lab (2023). Amazon reviews 2023 dataset. *UC San Diego – McAuley Lab*. https://amazon-reviews-2023.github.io/

Pérez-López, D., Dueñas-Lerín, J., Ortega, F., & González-Prieto, Á. (2023). New trends in AI for recommender systems and collaborative filtering. *Applied Sciences*, *13*(15), 8845. https://doi.org/10.3390/app13158845

Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). *BPR: Bayesian personalized ranking from implicit feedback*. In Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (pp. 452–461). AUAI. https://doi.org/10.48550/arXiv.1205.2618

Ricci, F., Rokach, L., & Shapira, B. (2015). *Recommender systems handbook* (2nd ed.). Springer. https://doi.org/10.1007/978-1-4899-7637-6

Rochin Demong, N. A., Shahrom, M., Abdul Rahim, R., Omar, E. N., & Yahya, M. (2023). Personalized recommendation classification model of students' social well-being based on personality trait determinants using machine learning algorithms. *Journal of Information and Communication Technology*, *22*(4), 545–585. https://doi.org/10.32890/jict2023.22.4.2

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). *Item-based collaborative filtering recommendation algorithms*. In Proceedings of the 10th International Conference on World Wide Web (pp. 285–295). Hong Kong. https://doi.org/10.1145/371920.37207

Steck, H. (2019). *Embarrassingly shallow autoencoders for sparse data*. In Proceedings of The Web Conference (pp. 3251–3257). ACM. https://doi.org/10.1145/3308558.3313710

Sun et al. (2019). *BERT4Rec: Sequential recommendation with bidirectional encoder representations from Transformer*. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management (pp. 1441–1450). *ACM*. https://doi.org/10.1145/3357384.3357895

Sun, Z., Guo, J., Cheng, X., & Shen, J. (2021). A survey of session-based recommender systems. *Information Processing & Management*, *58*(5), 102682. https://doi.org/10.1016/j.ipm.2021.102682

Wang, J., Zhou, K., Zhang, W., & Yu, Y. (2023). A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, *35*(4), 3052–3071. https://doi.org/10.1109/TKDE.2021.3117415

Wang, X., He, X., Cao, Y., Liu, M., & Chua, T. S. (2019). *KGAT: Knowledge graph attention network for recommendation*. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 950–958). ACM. https://doi.org/10.1145/3292500.3330989

Wei, X., He, X., Tang, J., & Chua, T. S. (2022). Sequential recommender systems: Challenges, progress and prospects. *AI Open*, *3*, 1–15. https://doi.org/10.1016/j.aiopen.2022.01.001

Wu, S., Sun, F., Zhang, W., Xie, X., & Cui, B. (2022). Graph neural networks in recommender systems: A survey. *ACM Computing Surveys*, *55*(5), 97. https://doi.org/10.1145/3523228

Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W. L., & Leskovec, J. (2018). *Graph convolutional neural networks for web-scale recommender systems.* In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 974–983). ACM. https://doi.org/10.1145/3219819.3219890

Yu, J., Yin, H., Xia, X., Chen, T., Li, J., & Huang, Z. (2023). Self-supervised learning for recommender systems: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 1-20. https://doi.org/10.1109/TKDE.2023.3282907

Zhang, Q., Lu, J., & Jin, Y. (2021). Artificial intelligence in recommender systems. *Complex & Intelligent Systems*, *7*, 439–457. https://doi.org/10.1007/s40747-020-00212-w

Zhang, Y., & Chen, X. (2020). Explainable recommendation: A survey and new perspectives. *Artificial Intelligence Review*, *53*, 571–604. https://doi.org/10.1007/s10462-019-09777-y

Zhou, H., Xiong, F., & Chen, H. (2023). A comprehensive survey of recommender systems based on deep learning. *Applied Sciences*, *13*(20), 11378. https://doi.org/10.3390/app132011378