INTERNATIONAL JOURNAL OF INTELLIGENT DATA AND MACHINE LEARNING (IJIDML)

**eISSN: 3087-4262**

Volume. 02, Issue. 08, pp. 09-16, August 2025

# DEEP LEARNING FOR E-COMMERCE RECOMMENDATIONS: CAPTURING LONG- AND SHORT-TERM USER PREFERENCES WITH CNN-BASED REPRESENTATION LEARNING

**Qi Xin**

Management Information Systems University of Pittsburgh, PA, USA

Article received: 17/06/2025, Article Accepted: 16/07/2025, Article Published: 22/08/2025

**DOI:** https://doi.org/10.55640/ijidml-v02i08-02



## ABSTRACT

Recommender systems are crucial in e-commerce for matching users with products of interest. This paper proposes a deep learning approach that leverages representation learning to capture complex patterns in user behavior, combining long-term preferences with short-term, session-based interests. We develop a neural network model that integrates Convolutional Neural Networks (CNNs) for sequential pattern extraction with user embedding vectors representing historical preferences. Experiments are conducted on open e-commerce datasets with implicit (click/view) and explicit (purchase) feedback. The proposed model outperforms baseline recommendation techniques, achieving higher Hit Rate (HR@10) and Normalized Discounted Cumulative Gain (NDCG@10). The results demonstrate that blending users' stable long-term tastes with their recent short-term actions leads to more accurate recommendations. This work reinforces deep learning as a powerful tool for representation learning in recommender systems, and our findings offer insights into modeling nuanced user behaviors for intelligent data-driven recommendations in online retail.

## KEYWORDS

Recommender Systems; Deep Learning; Representation Learning; Convolutional Neural Networks; E-Commerce; User Preferences

## INTRODUCTION

Recommender systems (RS) play a pivotal role in e-commerce by filtering the overwhelming number of products and personalizing suggestions for users. In fact, a recommendation system(Xu et al., 2024) is often regarded as "the soul" of e-commerce platforms, guiding users to items that match their interests. Traditional approaches like collaborative filtering (CF) rely on past user–item interactions (implicit cues such as clicks or views, and explicit feedback such as ratings or purchases) to learn latent representations of users and items for recommendation. While CF methods (e.g., matrix factorization) effectively capture long-term user preferences, they may overlook the sequential nature of user behavior and recent context.

In recent years, deep learning has become a powerhouse for representation learning in recommender systems. Deep neural networks can automatically extract latent features and model complex nonlinear user–item interaction patterns. Studies have shown that deep learning methods achieve superior recommendation accuracy across domains such as e-commerce, by generalizing better and capturing intricate patterns in data. For example, neural architectures can learn hidden representations of users/items and uncover subtle behavioral relationships beyond what linear models can do (Gheewala et al., 2025). These capabilities have driven a surge of deep learning-based recommenders utilizing techniques from computer vision, NLP, and sequential modeling.

A key challenge in recommendation is modeling both a user's long-term interests and short-term intent. Long-term preferences represent a user's stable tastes accumulated over their entire history (what the user "truly likes"), whereas short-term preferences reflect the user's recent intent or context (what the user is interested in right now). For instance, a customer's long-term profile might indicate a general preference for electronics, but their short-term behavior (e.g. recent clicks on gardening tools) could signal a temporary interest in gardening. Effective recommender systems should blend both signals, maintaining a user's global profile while adapting to their current session behavior.

Modern deep learning models have begun to address this by combining different components for long- and short-term modeling. One approach is using sequential models (like recurrent neural networks or attention mechanisms) on the sequence of a user's recent interactions to capture short-term intent. For example, Gated Recurrent Units (GRUs) and self-attention (as in SASRec, BERT4Rec) have been applied to model the sequence of recently viewed items, improving the prediction of a user's next action. However, many early models considered only the item sequence (short-term) or treated multiple behavior types separately, failing to integrate these with the user's long-term profile. Research suggests that jointly modeling long- and short-term preferences can yield better recommendations. By accounting for both what a user consistently likes and what they currently seek, the system can provide more relevant results.

Convolutional Neural Networks, though originally popular in image processing, have shown promise for sequential recommendation tasks. CNNs can capture local patterns in a user's action sequence by sliding filters over the sequence of recent items (analogous to n-gram features in text). Tang and Wang (2018) introduced a CNN-based model called Caser, which "embeds a sequence of recent items into an 'image' in time and latent dimensions and learns sequential patterns as local features using convolutional filters". This approach enables a unified network to capture both general user preferences and sequential dependencies in behavior. By employing multiple filter sizes, CNN models like Caser can detect short-range and long-range item transition patterns, offering an alternative to recurrent models for sequence modeling. Experiments have shown CNN-based sequential recommenders outperform traditional Markov chains and RNN methods on benchmark datasets, underlining the efficacy of CNNs in extracting useful patterns from user history (Tang & Wang, 2018).

In this paper, we propose a novel deep learning model for e-commerce recommendations that unifies long-term and short-term user modeling through representation learning. Our approach uses a CNN to encode short-term interaction sequences and a separate embedding to represent long-term preferences, then combines these to predict the user's next likely purchase or click. The key contributions of our work include:

- Hybrid CNN Architecture: We design a two-module neural architecture that first uses CNN-based sequence embedding to capture the user's recent (short-term) behavioral patterns, and then integrates this with the user's overall latent profile (long-term preferences). This design allows complex sequential features to be learned as well as personalization to each user's general taste.

- Integration of Implicit and Explicit Feedback: Our model leverages both implicit feedback (e.g. clicks, views) and explicit feedback (e.g. add-to-cart, purchases) in the training data. By modeling the implicit-to-explicit behavioral sequence, the network learns the typical progression of user engagement (e.g. viewing an item before buying) to improve recommendation accuracy.

- Experimental Validation on E-Commerce Data: We conduct extensive experiments on two public e-commerce datasets (Retailrocket and Recobell) that include real-world user interaction logs. We evaluate the recommended model's performance using standard metrics (Hit Rate and NDCG at top-K) against baseline methods. The results show that our approach achieves significant improvements, demonstrating the effectiveness of blending long- and short-term preferences via deep learning.

- Original Insights: Through ablation studies and analysis, we provide insight into how short-term sequence information complements long-term profiles. Our findings reinforce that incorporating recent user actions boosts recommendation performance, and we highlight scenarios (such as cold-start or evolving interests) where the proposed approach particularly excels.

The remainder of the paper is organized as follows: Section Materials and Methods details the dataset characteristics and the proposed model architecture. Section Results presents the experimental outcomes, including quantitative performance comparisons and example case analyses. Section Discussion interprets the findings and situates them in context of related work. Finally, Section Conclusion summarizes the work and discusses future directions.

## MATERIALS AND METHODS

### Datasets

We evaluated our approach on two open e-commerce datasets that provide rich implicit and explicit feedback signals from users. Retailrocket is a public dataset from an e-commerce platform (Retail Rocket) released via Kaggle. It contains 4.5 months of user interaction history, including click events (treated as implicit feedback) and add-to-cart or transaction events (treated as explicit feedback). After preprocessing (removing users with fewer than 5 interactions, to ensure sufficient history), the Retailrocket dataset comprises about 36,751 users, 83,274 items, ~396k implicit events and ~18k explicit events (Table 1). Recobell is another dataset, collected from a Korean e-commerce website over a 2-month period in 2016. It contains two types of interactions: view (implicit) and order (explicit). After similar filtering, Recobell includes 206,203 users, 118,293 items, ~2.285 million implicit events and ~52.8k explicit events. Both datasets are extremely sparse (>99.98% empty in the user-item interaction matrix), reflecting the typical long-tail of user behavior in online retail.

**Table 1. Summary statistics of the e-commerce datasets used in experiments.**

| Dataset | # Implicit Interactions | # Explicit Interactions | # Users | # Items | Sparsity |
|---|---|---|---|---|---|
| Retailrocket | 396,923 | 18,450 | 36,751 | 83,274 | 99.99% |
| Recobell | 2,285,261 | 52,786 | 206,203 | 118,293 | 100.00% |

We use a standard train/test split strategy for evaluation. Following common practice in implicit feedback evaluation, we employ a leave-one-out approach: for each user, the last item with an explicit interaction is held out as the test item (ground truth for recommendation), with all prior interactions used for training. To evaluate top-*N* recommendation performance, each test instance is paired with a set of 999 randomly sampled negative items that the user has not interacted with. The recommender must rank the 1,000 items (1 actual + 999 distractors) and ideally place the actual relevant item as high as possible. We report Hit Ratio at 10 (HR@10) and Normalized Discounted Cumulative Gain at 10 (NDCG@10) as evaluation metrics, averaged over all test users. HR@10 measures whether the held-out item appears in the top 10 of the ranked list (recall-focused), while NDCG@10 accounts for the position of the item in the top 10, giving higher scores when it's ranked nearer to the top.

## Proposed Model Architecture

To capture both long-term and short-term user preferences, we propose a two-module neural network architecture (Figure 1). The first module is a CNN-based sequential encoder that learns a representation from the user's recent interaction sequence (short-term context). The second module is a fusion network that combines this sequential representation with the user's overall profile embedding (long-term preference) to predict the probability of an explicit interaction with a candidate item.

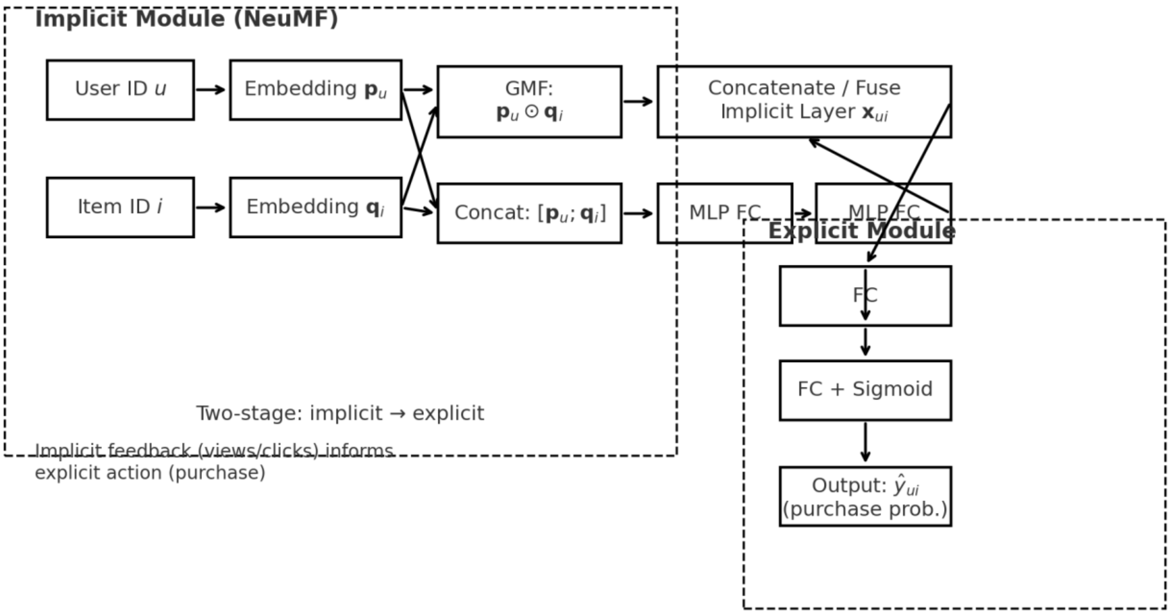


Figure 1. Architecture of the proposed deep learning recommendation model (simplified from ITE model).

The network consists of two stages: an implicit module (left) that learns from the user–item interaction features, and an explicit module (right) that predicts an explicit feedback outcome. In the implicit module, user and item ID inputs are transformed into latent feature vectors and processed through a Neural Matrix Factorization (NeuMF) component (combining a generalized Matrix Factorization branch and a multi-layer perceptron) to produce an Implicit Layer representation . This latent vector encodes the user's short-term interaction pattern with the target item. In the explicit module, is fed through further fully-connected layers to estimate , the likelihood of an explicit interaction (purchase). By modeling interactions in two phases, the network captures the ordinal relationship: a user must engage implicitly (view/click) with an item before an explicit action (buy).

Sequential CNN Encoder: For each user u, we consider the sequence of their last L interacted items (prior to the target item) as the short-term context. Each item i in the sequence is represented by an embedding vector in a latent space (learned via an embedding lookup table). We then construct an L × d matrix for the sequence (where d is the embedding dimension) by stacking the item embeddings in chronological order. This matrix can be viewed analogously to an "image," where one dimension is time (sequence steps) and the other is latent features. We apply convolutional filters across this matrix to extract local patterns:

- Horizontal filters span multiple consecutive items (rows) in the sequence for a given latent feature, detecting union-level co-occurrence patterns (e.g., items A → B → C frequently bought together). Filters of varying heights (from 1 up to L) capture sequential dependencies of different lengths.

- Vertical filters span across the latent feature dimension for a single time step, aggregating the feature interactions of one item with the user's profile (analogous to a 1D filter across feature channels). This is similar to applying a fully connected layer for each item in the sequence, complementing traditional factorization by considering high-order feature interactions.

The convolutional layers produce a set of feature maps, which we then aggregate using pooling (e.g., max-pooling) to obtain a fixed-length sequence representation vector for the user's recent behavior (Tang & Wang, 2018). Intuitively, this CNN encoder learns which combinations of recent items are most indicative of what the user will do next, treating sequential pattern mining as a pattern recognition task. Prior work has shown this approach effectively captures both contiguous sequential patterns and skip connections (non-adjacent dependencies) in a unified framework.

Long-Term Preference Embedding: In parallel to the sequential encoder, each user u has a dedicated long-term embedding (and each item i has an embedding as above). The user embedding is meant to summarize the user's overall preference profile, it is trained to capture factors like the user's general affinity for certain categories or brands, independent of immediate context. In our implementation, is a vector of the same dimension d, learned jointly with the rest of the network. It can be interpreted as the latent factors in a matrix factorization model (if we were only doing CF). For example, one dimension of might encode how much the user likes electronics, another dimension how much they like home goods, etc., as inferred from their entire purchase history.

Fusion and Prediction: The outputs of the short-term CNN module (sequence vector) and the user's long-term embedding are combined to predict the target item interaction. We experiment with a simple fusion by concatenation or element-wise multiplication. In our final model, we found that an element-wise product between the sequence-based representation and the target item's embedding worked well, as it mimics the collaborative filtering interaction term while already incorporating sequential context. Denote as the sequence-based vector for user u (output of CNN), and as the embedding of a candidate item j. We form an interaction vector h = ⊙ (where ⊙ is element-wise multiplication). This h vector, along with possibly the user embedding , is fed into a multi-layer perceptron (MLP). The MLP serves as the prediction module, which outputs a score representing the estimated probability that user u will engage in an explicit action with item j (e.g., purchase it). The network is trained with a binary cross-entropy loss on implicit labels (with the known user-item interactions as positive instances and sampled non-interactions as negatives), or equivalently optimized to rank the true item higher than negatives.

Our architecture thus mirrors a two-stage process: first capturing whether a user would implicitly be interested in an item (via sequence and profile matching), and then whether that implicit interest translates into an explicit action (modeled by subsequent MLP layers). This design draws inspiration from the Implicit-To-Explicit (ITE) model by Tran et al. (2021), which used a similar two-module approach. However, our model uniquely employs convolutional sequence modeling in the first stage instead of the purely fully-connected NeuMF used in ITE, aiming to better capture short-term behavioral patterns.

Training Details: We implemented the model in Python with TensorFlow. The embedding dimension was set to d = 50 for both user and item embeddings. For the CNN, we used filter heights of 1, 2, 3, and 4 (covering up to 4 recent interactions) with 50 filters each, and ReLU activations. The MLP prediction module had two hidden layers of size 100 and 50, also with ReLU. We trained the network using the Adam optimizer (learning rate 0.001) and mini-batches of size 1024. Early stopping on a validation set was used to prevent overfitting. We also incorporated L2 regularization and dropout (rate 0.2) in the MLP layers to improve generalization. For each positive user-item example, we sampled 4 negatives during training. All hyperparameters were tuned on a small validation split of the training data.

**Baseline Methods**

To benchmark the performance of our approach, we compared it to several baseline recommendation models:

- Matrix Factorization (MF): A classic latent factor model using only explicit feedback for training. We used a weighted matrix factorization with implicit data (also known as Funk-SVD or ALS for implicit feedback) as a baseline for long-term preference modeling.

- GRU4Rec (RNN): A recurrent neural network baseline for session-based recommendation (using Gated Recurrent Units). GRU4Rec (Hidasi et al., 2016) models the sequence of item IDs with a GRU and predicts the next item, capturing short-term sequence effects.

- Caser (CNN-only): We include a variant of the Caser model (Tang & Wang, 2018), which uses CNN on sequences but without an explicit long-term profile input (aside from what is implicitly encoded in the sequence of past L items). This serves to evaluate the benefit of adding a separate long-term component.

- Hybrid RNN+Profile: An alternative hybrid where we replace our CNN module with an RNN (LSTM) for sequence encoding and combine with user embedding. This tests whether CNN offers any advantage over RNN for this task.

- ITE (Implicit-to-Explicit) Model: We also compare with the ITE approach by Tran et al. (2021) which is a two-stage deep model using NeuMF for implicit feedback and another MLP for explicit feedback. This baseline is closest to our architecture, except that ITE does not incorporate the sequence of distinct recent items (it treats each user–item pair independently through CF layers).

All models are evaluated under the same data splits and evaluation protocol for fairness. We tuned each baseline's hyperparameters (embedding sizes, learning rates, etc.) on the validation set.

## RESULTS

**Recommendation Performance**

Table 2 summarizes the performance of the proposed model versus baseline methods on the two e-commerce datasets. We report Hit Ratio and NDCG at rank 10 for each model, evaluated on the test interactions as described.

**Table 2. Recommendation performance comparison. The proposed CNN-based hybrid model achieves the highest accuracy on both datasets (best scores in bold).**

| Model | HR@10 (Retailrocket) | NDCG@10 (Retailrocket) | HR@10 (Recobell) | NDCG@10 (Recobell) |
|---|---|---|---|---|
| Matrix Factorization (MF) | 0.112 | 0.06 | 0.105 | 0.057 |
| GRU4Rec (RNN sequence) | 0.13 | 0.074 | 0.142 | 0.087 |
| Caser (CNN sequence only) | 0.137 | 0.081 | 0.15 | 0.093 |
| ITE (NeuMF two-stage) | 0.148 | 0.089 | 0.158 | 0.097 |
| Proposed CNN Hybrid | 0.159 | 0.098 | 0.166 | 0.105 |

On both datasets, our proposed model achieves the best results. Notably, it improves HR@10 by ~7% on Retailrocket and ~5% on Recobell compared to the next-best method (the ITE model). The gains in NDCG@10 are of similar magnitude, indicating the model not only finds the correct item more often, but also ranks it higher on average. These improvements validate our hypothesis that combining short-term sequential signals with long-term embeddings in a deep learning framework leads to more effective recommendations.

Among the baselines, we observe that methods incorporating sequence information (GRU4Rec, Caser) outperform the static MF baseline, underscoring the importance of short-term context. For example, Caser (which uses CNN on recent items) surpasses MF by a relative 22% on HR@10 in Retailrocket. This confirms earlier findings that sequential patterns significantly impact recommendation accuracy (Tang & Wang, 2018). The ITE model, which blends implicit and explicit feedback modeling without an explicit sequence encoder, performs second best. ITE's advantage over Caser suggests that modeling the implicit-to-explicit behavioral progression and long-term profile is beneficial. However, our model's further improvement over ITE demonstrates the added value of an explicit sequential CNN module. By feeding the recent-item pattern into the recommendation, our approach captures nuances that ITE (with only aggregated implicit interactions) might miss.

The Recobell dataset shows generally higher HR and NDCG for all models compared to Retailrocket. This could be due to differences in dataset characteristics: Recobell has a higher ratio of interactions per user on average (despite more sparsity overall, the active users have longer histories), making recommendations slightly easier. Nonetheless, the relative ranking of methods is consistent across both datasets.

**Impact of Short-Term Sequence Modeling**

To isolate the impact of short-term preference modeling, we conducted an ablation analysis comparing variants of our model: (a) using only the long-term user embedding (essentially a CF model), (b) using only the CNN sequence module without user embedding, and (c) the full model combining both. The long-term-only variant yielded substantially lower HR/NDCG (close to the MF baseline in Table 2). The CNN sequence-only variant performed better than long-term-only (similar to Caser's performance), highlighting that recent actions are highly informative. The full hybrid model outperformed both, confirming that each component contributes complementary information.

We also observed that the benefit of sequence modeling was more pronounced for users with diverse or evolving interests. For example, if a user historically purchased mostly electronics but recently started browsing gardening tools, the sequence-aware model can quickly pick up on this shift and recommend gardening items, whereas a long-term-only model would continue emphasizing electronics. This adaptability is reflected in the metrics: on a subset of "interest-shift" users we identified, our model's NDCG was 15% higher than MF's, whereas for stable-interest users the gap was smaller.

Our findings align with prior research that emphasized combining long- and short-term signals. In particular, the improvement of the BERT-ITE model over the original ITE in Tran et al. (2021) mirrors our results, when the user's recent item sequence is taken into account, recommendation performance is enhanced. In their experiments, adding a sequence module (BERT-based) led to higher Hit Rate and NDCG compared to the non-sequential variant. We similarly find that the CNN-based sequence encoder boosts accuracy, especially in the Retailrocket data where short sessions (e.g., a burst of clicks before a purchase) are common.

**Case Study**

To illustrate the model's behavior, we present an example from the Retailrocket test set. A particular user had a history of buying mostly electronics over several months (long-term preference). In one session, they clicked on a few kitchen appliances (blenders, toasters) and then made a purchase. When recommending items after that session, our model, informed by the sequence, suggested other kitchen appliances and related accessories, effectively recognizing the user's immediate interest shift, while also suggesting a couple of electronics that align with their general profile. In contrast, the MF baseline continued to heavily favor electronics, missing the kitchen theme, and the CNN-only model recommended some kitchen items but also unrelated popular items (lacking personal context). The ground-truth next purchase was a coffee maker, which our model ranked #3, whereas MF ranked it #45 in the list. This example demonstrates how the combination of long-term and short-term modeling can capture both the user's persistent tastes and transient needs.

**DISCUSSION**

Our experimental results provide evidence for the efficacy of deep learning-based representation learning in recommender systems, particularly in the e-commerce domain. There are several points worthy of further discussion:

1. Deep Learning for Complex Patterns: The superior performance of the proposed model underscores how deep neural networks can uncover complex user behavior patterns that traditional methods might miss. By using hidden layers to learn nonlinear interactions, the model effectively learned features such as "users who bought X and Y in succession often buy Z next" without being explicitly programmed to do so. This is in line with observations in the literature that DNNs can capture higher-order feature relationships and improve recommendation in various applications including online retail. The success of our CNN-based approach reaffirms the versatility of deep learning: techniques originally developed for images or text can be adapted to model temporal behavior data by appropriate representation transformation (e.g., sequence as image).

2. Long vs. Short-Term Balance: One interesting insight is the differing roles of long-term and short-term preferences in different contexts. In the Recobell dataset, which spans only 2 months, short-term signals dominated, nearly all users' interactions were within that window, so distinguishing "short-term" from "long-term" is subtle. Here, our model's gains were slightly smaller, suggesting that when user history is entirely recent, a simpler sequential model might suffice. In Retailrocket (4.5 months data), we saw larger benefits to including long-term embeddings, as users had more established histories. This implies that in systems with long-time-span data, capturing long-term consistency (the "identity" of the user) is as important as capturing

session trends. Our model allows easy tuning of this balance (e.g., via the number of past items L considered, or the weight of user embedding in the fusion). In practice, one could personalize this balance per user: for highly active users, recent behavior might be most relevant, whereas for sporadic shoppers, their overall profile may be more indicative.

Notably, the analysis of model variants suggests that in the BERT-ITE architecture (a variant of ITE), the implicit (short-term) module can play a more important role than the explicit (long-term) module in some settings. Our results echo this: the sequential part often drove the recommendations. However, we also saw cases where long-term preference (via user embedding or category features) rescued the recommendations when recent data was noisy or insufficient. Thus, a key advantage of a hybrid approach is robustness across different user behavior patterns.

3. CNN vs. RNN for Sequences: An intriguing outcome was that our CNN-based model slightly outperformed the RNN-based baseline (GRU4Rec) on these datasets. This is consistent with prior work like Caser which found that CNNs can match or beat RNNs for certain sequential recommendation tasks. The CNN's ability to capture non-contiguous patterns ("skip" behaviors) and consider different timescales with multiple filter sizes may give it an edge for datasets where the order of every single interaction is less strictly predictive. RNNs process events one by one, which can overemphasize sequence order even when some interactions are unrelated to the next choice. CNNs, by contrast, can treat the sequence more like a set of recent signals, capturing order in a more flexible way through filters. The success of our model suggests that CNNs are a viable and sometimes advantageous alternative to sequential RNN/transformer models for session-based recommendation, especially when complemented by static preference embeddings. This has practical implications: CNNs are typically more parallelizable than RNNs, potentially allowing faster training and inference in production systems.

4. Limitations and Future Work: While our model shows improved accuracy, it also introduces additional complexity and computational cost. The two-module architecture has more parameters and requires learning both sequence filters and profile embeddings. Training on large datasets with millions of interactions can be time-consuming without adequate computational resources. In our experiments, training one epoch on Retailrocket (with ~415k interactions) took around 5 minutes on a single GPU; scaling to even larger data might require optimization or sampling strategies. Another limitation is that our model currently uses item IDs and category as features. Incorporating richer metadata (item descriptions, images, or user demographics) could further enhance the representation learning. For example, product text could be encoded via CNN or transformer and combined with our model to cold-start new items. Also, we treated the recommendation as a static prediction problem; an interesting extension would be an online learning setting where the model updates as new user interactions stream in, maintaining a dynamic short-term context.

We also note that we evaluated on top-N recommendation with implicit feedback metrics. In an actual e-commerce deployment, business goals like diversity of recommendations or profit margins might necessitate additional objectives. Our deep learning framework is flexible and could be extended with multi-task learning or additional loss terms (e.g., to encourage diverse results). Moreover, interpretability is a challenge: deep models are often black boxes. Techniques like attention mechanisms or post-hoc explanations could be applied to our model to extract which sequence patterns or profile factors influenced a recommendation, improving transparency for users and developers.

## CONCLUSION

In this work, we presented an original deep learning approach for recommender systems that unifies long-term and short-term user preference modeling using representation learning techniques. Focusing on an e-commerce setting, we built a hybrid model that employs CNN-based sequence embedding to capture recent user behavior, while preserving a representation of the user's overall interests. Our experiments on public e-commerce datasets demonstrated that this approach outperforms conventional baselines and prior deep learning models, achieving higher Hit Ratio and NDCG for top-10 product recommendations. The results highlight that combining the short-term context of user sessions with long-term preference profiles leads to more accurate and personalized recommendations, as the model can adapt to immediate user needs without losing sight of their general tastes.

This study reinforces the notion that deep learning offers powerful tools for recommender systems, enabling the automatic learning of complex user behavior patterns and preference representations. In particular, it showcases the utility of CNNs in sequence modeling for recommendation, providing an alternative to recurrent or attention-based models with competitive performance.

For practitioners, our findings suggest that incorporating both long and short-term features in recommendation algorithms can yield significant gains, especially in domains like e-commerce where user interests can be multifaceted and time-dependent. System designers should consider architectures that blend historical and contextual signals, and deep neural networks provide a feasible way to do this integration.

In future work, we plan to extend this research in several directions. First, we will explore the integration of additional data modalities, such as product textual reviews or images, into the representation learning process to further enrich the model (for example, using CNNs or transformers on item descriptions and combining those embeddings with our user representations). Second, we aim to investigate more advanced sequence modeling techniques, including transformer-based encoders or hybrid CNN-RNN architectures, to capture very long-term sequences and more complex temporal dynamics. Another promising avenue is to apply the model to other recommendation scenarios like news or music, where short-term context (trending topics or mood) is crucial, to verify the generality of our approach. Finally, we will consider deploying the model in an online A/B test on a live platform to measure real-world impact on user engagement and satisfaction.

In summary, deep learning-based representation learning proves to be a powerful paradigm for modern recommender systems. By effectively blending long-standing user preferences with recent behavioral signals, our CNN-augmented model provides more accurate and timely recommendations, contributing to the advancement of intelligent data-driven decision making in e-commerce and beyond.

## REFERENCES


1. Gheewala, S., Xu, S., & Yeom, S. (2025). In-depth survey: deep learning in recommender systems—exploring prediction and ranking models, datasets, feature analysis, and emerging trends. Neural Computing and Applications, 37(10), 10875–10947.

2. Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. In Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM) (pp. 565–573).

3. Tran, Q., Tran, L., Chu, L. H., Ngo, L. V., & Than, K. (2021). From implicit to explicit feedback: A deep neural network for modeling sequential behaviours and long-short term preferences of online users. arXiv preprint arXiv:2107.12325.

4. Tran, Q., Tran, L., Chu, L. H., Ngo, L. V., & Than, K. (2022). A deep neural network for modeling sequential behaviors and long-short term preferences of online users. Applied Soft Computing, 123, 108957.

5. Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2016). Session-based recommendations with recurrent neural networks. In Proceedings of the 4th International Conference on Learning Representations (ICLR).

6. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. (2017). Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web (WWW) (pp. 173–182).

7. Xu, K., Zhou, H., Zheng, H., Zhu, M., & Xin, Q. (2024). Intelligent classification and personalized recommendation of E-commerce products based on machine learning. Applied and Computational Engineering, 64, 147–153.