Indian Journal of Information Sources and Services
ISSN: 2231-6094 (P) Vol.15, No.3, 2025, pp.330-338
© The Research Publication, www.trp.org.in
DOI: https://doi.org/10.51983/ijiss-2025.IJISS.15.3.37

Neural Re-Ranking Models in Research Article Recommendations

Dr.K. Sankara Moorthy[1*], Mohammed Hussein Fallah[2], Manoj Govindaraj[3], Dr.S. Rama Sree[4], S. Kanmani Jebaseeli[5] and Juraev Tokhirjon Mansurali Ugli[6]

[1*]Assistant Professor, Faculty of Management, SRM Institute of Science and Technology, Kattankulathur, India

[2]Department of Computers Techniques Engineering, College of Technical Engineering, Islamic University of Najaf, Najaf, Iraq; Department of Computers Techniques Engineering, College of Technical Engineering, Islamic University of Najaf of Al Diwaniyah, Al Diwaniyah, Iraq

[3]Associate Professor, Department of Management Studies, VelTech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, India

[4]Professor, Department of Computer Science and Engineering, Aditya University, Surampalem, Andhra Pradesh, India

[5]Assistant Professor, Department of IT, New Prince Shri Bhavani College of Engineering and Technology Chennai, Tamil Nadu, India

[6]Faculty of Business Administration, Turan International University, Namangan, Uzbekistan

E-mail: [1]sankarak@srmist.edu.in, [2]eng.iu.comp.mhussien074@gmail.com, [3]manoj.nmcc@gmail.com, [4]ramasree_s@adityauniversity.in, [5]kanmani.s@newprinceshribhavani.com, [6]tohir2001@mail.ru

ORCID: [1]https://orcid.org/0000-0002-6853-6878,[2]https://orcid.org/0009-0001-3128-2907, [3]https://orcid.org/0000-0003-2830-7875, [4]https://orcid.org/0000-0002-8771-6006, [5]https://orcid.org/0009-0002-5985-4388,[6]https://orcid.org/0000-0003-2876-892X

(Received 24 June 2025; Revised 12 August 2025, Accepted 26 August 2025; Available online 30 September 2025)

Abstract - **Refining the rank lists to ensure user satisfaction and experience is achieved through a re-ranking procedure, which is the culmination of the multi-stage recommender system. Gradation and correlation vary with its satisfaction. With the advent of deep learning approaches, neural-based re-ranking is a prominent research topic and is increasingly finding business applications in industry. To bolster comprehensive and effective solutions for future work by contextualizing the algorithms into a broader literature scope focused on re-ranking. It is done by creating a taxonomy of existing heuristics built on neural networks. After that, the terms of their evolution and main aims, which incorporate subdominant systems like network architecture, personalization strategies, and the welfare of computational complexity, are described. Additionally, an analysis of the main benchmark models of neural re-ranking is provided. It is also accompanied by detailed metrics for quantifying the model benchmarks used in the analysis. The final chapter, which closes the review, focuses on the research in question and other directions that can be pursued. The appendix contains cited works and benchmark datasets which is used in the analysis.**

Keywords: **Neural Networks, Re-Ranking, Research Article, Benchmarks, Model, and Computational Complexity**

I. Introduction

The Multistage Recommender Systems (MRS) utilize social networking platforms such as LinkedIn (Geyik et al., 2019), Google (Bello et al., 2018), Taobao (Pei et al., 2019), and YouTube (Wilhelm et al., 2018), which are considered frameworks of their systems. These systems accommodate users and items reaching up to a billion by troubleshooting processing problems associated with the scale. The recommendation problem consists of multiple stages, providing relevancy and speed, by narrowing down with each step, filtering down to a Limited selection using more complex and slower models (Hron et al., 2021).

MRS is split into three parts: recalling or matching, ranking, and re-ranking. The model collects items in a corpus and attempts to retrieve an optimal subset. Candidates are assigned scores along with a rank. The list is done by improving the objectives and constraints on the recommendation or attempting to make it multidimensional, a process known as re-ranking.

The re-ranking component optimizes the best possible overall ranking, which refers to context within the recommendation list and its relational dependencies, including interdependencies from the provided selection. An individual’s interest in an item may be determined by its feature about how that item is organized, and positioned relative to other items in the same list (Pei et al., 2019). Capturing these interactions among items is the reason why ineffective reranking strategies exist (Shi et al., 2024).

The re-ranking concept emerged from the marginal relevance algorithm, initiated by (Carbonell & Goldstein, 1998), which involves the sequential selection of items that are consistent with thresholds for overall diversity and holistic relevance. With the shift in industrial focus towards deep learning, neural re-ranking was distributed, relying heavily on deep learning techniques to replace domain-specific features with learned ones in automated systems, eliminating the need for engineered features. The capacity of neural networks to universally approximate functions (Cybenko, 1989) made these models suitable for modeling complex interdependencies between item lists (Goodfellow et al., 2016; Goldberg, 2017).

During recent years, there has been an increase in the use of neural architectures for re-ranking in both academic and practical settings. This stands as an early example for detailed assessments targeting neural re-ranking techniques in recommender systems, to grow the scope of comprehension and encourage further exploration in this rapidly evolving technology domain.

II. A Taxonomy of Neural Re-Ranking Models

The categories describing neural re-ranking models differ in terms of optimization objectives, which include optimizing for accuracy versus multi-goal approaches, as well as the type of supervision signals employed: user-based interactions versus counterfactual feedback. These two dimensions are represented in a taxonomy with four quadrants shown in Fig 1.

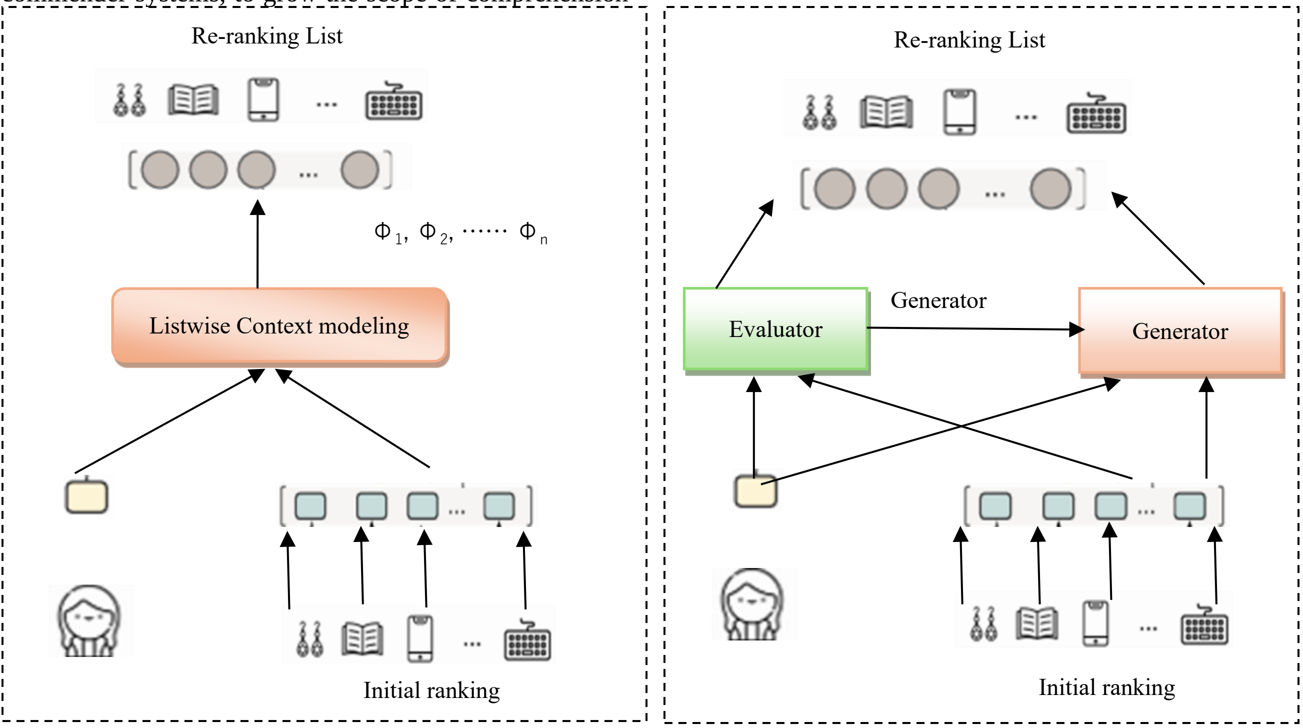

Fig 1. Re-ranking Network Architectures

Most existing work tends to focus on a single objective: achieving high prediction accuracy. This complements the primary purpose of recommender systems, which is to predict a user's preferences accurately. However, in addition to accuracy, modern systems must also address other emerging requirements, such as supporting diverse goals, novelty, and fairness (Kim et al., 2022). Recent works have begun to address how to balance these competing objectives within re-ranking (Kaminskas & Bridge, 2016).

The second differentiating axis describes the origin of supervision signals. For instance, most neural re-ranking models appear to rely on observed data—interactions and feedback from users regarding the preliminary rankings of items provided to them. However, the literature suggests that user preferences are not strictly static and can depend significantly on the ordering and context provided in a list (i.e., listwise context). Therefore, some advanced methods (Conti et al., 2017) utilize counterfactual signals by simulating item ranking permutations that are not directly observable. These signals enable the estimation of preferences based on alternate arrangements of items (Zou et al., 2019).

From this taxonomy, several trends and gaps in current research are identified: most approaches still focus predominantly on precision and accuracy metrics, while using few diversity or fairness metrics in multi-objective approaches. An example of such focus is self-attention (Vaswani et al., 2017), combined with recurrent neural networks (RNNs) (Hochreiter & Schmidhuber, 1997), which has become prevalent due to their ability to capture dependencies between items. Some promising future work involves integrating counterfactual learning within multi-objective research frameworks, which addresses a gap in current research. The following sections will provide a more detailed analysis of specific neural re-ranking methods within this proposed taxonomy.

Re-Ranking Network Architectures

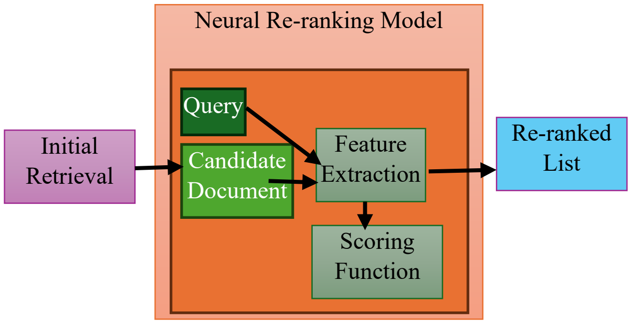

Fig. 2 Neural Re-ranking architecture

Recent improvements in neural networks have improved the performance of many domains, including computer vision, natural language processing, machine translation, and speech recognition. The primary neural components that contributed to the breakthrough in numerous domains are convolutional and recurrent neural networks. Deep neural network models lead to state-of-the-art results in a variety of applications, including information retrieval (IR). Deep neural networks are well-known for their ability to catch complicated patterns throughout both the feature extraction and model construction phases. Because of the benefits of deep neural networks, researchers have concentrated on developing neural ranking models that can learn both features and models simultaneously.

The neural re-ranking component is typically employed as a re-ranking phase with two inputs: candidate documents and processed queries. The candidate documents are retrieved using an unsupervised ranking step, such as BM25, which accepts as inputs the original set of indexed documents and the processed query. During unsupervised ranking, recall is more critical than precision in covering all possible relevant documents and forwarding a group of candidate documents, including both relevant and irrelevant documents, to the neural-based re-ranking step.

Inputs to neural ranking models include inquiries and documents of varying lengths. Neural ranking models that utilize textual input for query and document derive features from the local interactions between the two. For example, a deep neural network model can be used to separately map the query and documents to feature vectors, and then the recovered features are utilized to produce a relevance score. The neural ranking model produces a ranking of documents that are relevant to the user's query.

Neural reranking models tend to fall into one of two categories based on how they supervise the system using observed interactions or counterfactual feedback. The main concepts of the two categories are listed as follows:

(a) Standard Neural Re-Ranking with Observed Signals

In the traditional architecture of neural reranking, the model is trained on user-item interactions, which are based on user sessions within the current ranking lists. In this design, we typically include:

Input Layer: Builds receiving structures for the top N items output by the prior ranking stage, together with their features, embeddings, contextual information, and metadata.

Contextual Modeling Layer: This layer models dependencies among items in the list. Self-attention layers, RNNs, or even Transformers can capture listwise context and Recurrence.

Prediction Layer: Outputs a re-ranked score for each listed item in a way that captures item interdependencies and incorporates listwise interactions.

Loss Function: Based on the feedback received (from clicks, views, ratings, etc.), the rank-wise ordering is refined using learning-to-rank objectives, such as pairwise or listwise loss.

This approach effectively harnesses user feedback to enhance the relevance of item lists. However, it assumes that the ordering of items presented to users is optimal or unbiased.

(b) Evaluator-Generator Framework Using Counterfactual Signals

To transcend the limitations imposed by relying solely on observed interactions, counterfactual signals are incorporated into the Generator-Evaluator framework. This model aims to address the limitations of purely observational learning by enabling learning from unobserved permutations and capturing listwise contextual effects.

Generator: This generator simulates the prospective viewing strands of the items by proposing alternative permutations or reordering the items in different ways to present them to the user (Jelena & Srdan, 2023).

Evaluator: Serves as a proxy for user feedback and the scoring mechanism that provides relevance ratings for the proposed permutations, simulating responses for a pre-computed set of lists.

Training Strategy: The generator and evaluator are trained jointly or alternately. On some occasions, Reinforcement learning or adversarial approaches are used to increase the amount of unobserved data that can be utilized during training.

Objective: The system enhances recommendation quality by refining evaluation processes, where integrated simulation of various ranking orders pre-optimizes re-ordering results to provide deeper insights into nuanced recommendations. This system facilitates understanding of user-item interactions and enhances the capture of preferences and behavior by flexibly adjusting representations of user preferences based on item arrangements and behaviors in environments without direct feedback.

Neural Re-Ranking for Recommendation

Neural re-ranking within recommendation systems emphasizes creating a multivariate scoring function that accepts the complete output from the first ranking phase. This strategy retrieves cross-item interactions with listwise context. It is essential to assign a relevance score to each item, which is not based solely on its features but also on the surrounding items within the list (Liu et al., 2020; Ai et al., 2018). Unlike traditional models, these "linear" approaches employ referential scoring techniques and assume that all items are processed sequentially by a univariate function one at a time, capturing item relationships only through the loss function via pairwise or listwise losses, rather than through direct modeling of the output (Su et al., 2024).

$$\varphi^{*} = \arg\min L\ (\varphi(R), Y) \qquad (1)$$

L (·) is a loss function used to train a re-ranking model. In most cases, the primary focus is maximizing the number of accurate rank positions to achieve ranking accuracy, which is served through measures such as NDCG and MAP. However, reranking could target broader system objectives, such as increasing concept changes through novelty, promoting fairness, or enhancing diversity in recommendations (Angel et al., 2022).

This form of redefining the rerank problem scope can help build the taxonomy we have seen in earlier figures (refer to Fig 1). Models can be placed in a box with these:

The focus of the objective: whether the accuracy loss is leveled out alone, or has multiple objectives alongside it, like accuracy – "+ diversity".

Supervision type: Y, whether derived from user-observed supervision (e.g., clicks, ratings) or evaluated through quasi-observational review of item permutations, items are evaluated using surrogate functions (Ganesh & Sivakumar, 2021).

These categories are termed as 'counterfactual' and 'observational' as illustrated in Fig 2. Models that depend on observed signals use a direct architecture where re-ranking scores are computed based on interactions captured among items in the list. In contrast, counterfactual models employ a generator-evaluator framework (David et al., 2024), where a generator produces re-ranked lists of items as proposals, while an evaluator scores these proposals using contextual information.

In the following sections, neural re-ranking models will be analyzed in detail based on the four quadrants of our taxonomy: single vs multi-objective models and observed vs. counterfactual supervision.

III. Single Objective: Focused on Reranking for Accuracy

As previously discussed, enhancing recommendation accuracy remains a hallmark endeavor for neural re-ranking model frameworks in modern recommender systems (MRS). The performance of the re-ranking model is evaluated by ordering the comments by value and measuring both NDCG and MAP at the list level.

Current accuracy-oriented reranking approaches can be categorized into two types depending on the type of supervision signals guiding the learning process.

- Learning from Observed Signal
- Learning from Counterfactual Signal
- Learning from observed signals

This method maximizes the comprehensive feedback ranking R.

R given the ground truth label Y.

Y, which includes the user's feedback, like clicks or ratings. These labels correspond to the items shown to the user and, therefore, can be considered direct and simple noise with lower supervision.

The standard setup architecture, shown in Fig 2a, begins at the embedding stage, where users and items are represented as dense vectors. Then, the model calculates the cross-item dependencies using listwise context modeling to derive re-ranking scores. This direct learning paradigm benefits from rich, high-quality, explicit relevance signals and is robust in empirical studies (Ai et al., 2018; Pei et al., 2019; Liu et al., 2020; Zou et al., 2019). Depending on which neural model is employed to represent listwise relations, one can categorize observed-signal techniques into:

a. Recurrent Listwise Modeling

Early neural re-ranking models utilized Recurrent Neural Networks (RNNs) to process top-ranked items sequentially, capturing both positional and sequential dependencies. (Ai et al., 2018)'s Deep Listwise Context Model (DLCM) use Gated Recurrent Units (GRUs) to sequentially encode item vectors, allowing them to capture the influence of previously scored items on the current item's score. MiDNN (Bello et al., 2018) uses LSTM networks and global feature augmentation and formulates the re-ranking as a sequence generation task with beam search. Seq2Slate (Abri et al., 2022; Ahmed & Pandey, 2024) builds on MiDNN with a pointer network and attention mechanism to enable it to dynamically choose the next item in the sequence from the original list.

b. Attentive Listwise Modeling

Effort has been made in the application of multi-head self-attention to re-ranking because self-attention mechanisms have prospered in natural language processing. It allows for more effective simulation of all pairwise item interactions than sequential encoding (Vaswani et al., 2017). PRM (Personalized Reranking Model). For instance, a cross-item influence model for every user act as a personalized embedding layer, enhancing user-specific interactions. It is founded on a transformer-like architecture consisting of stacked self-attention blocks and positional embeddings.

- (Huang et al., 2020) model flight itinerary re-ranking as user behavior at multiple levels, capturing long-term preferences and real-time actions using multiple attention heads.
- (Li et al., 2022) incorporate user-specific attention weights in modeling cross-item interactions to personalize modeling further.
- (Li et al., 2022) propose a cross-attention model that considers the items in the initial list and the user's history, including the items they have clicked on. PEAR introduces feature-level interactions and an auxiliary classification task for item attention, further enhancing performance.

Attentive models have been shown to combine flexibility with expressive power. They can enrapture many complex dependencies among candidate items, which can be aligned to user behavior to improve recommendation systems.

c. Mathematical Model

- **Rank Net:** Rank Net is a paired approach that employs a neural network to train a ranking function. Its primary goal is to reduce the number of inversions in the ranked list.
- **Scoring Function:** The neural network's output is a score, si=f(xi), for each document i, where xi is the feature vector of the document.
- **Predicted Probability:** The probability that document i is ranked higher than document j is calculated using a sigmoid function:

$$P_{ij} = \frac{1}{1+e^{-(s_i-s_j)}} \quad (2)$$

- **Loss Function (Cross-Entropy):** RankNet minimizes the cross-entropy loss between the predicted probability and the true probability (which is 1 if document i should be ranked higher than j, and 0 otherwise):

$$L = -\bar{P}_{ij}log(P_{ij}) - (1-\bar{P}_{ij})log(1-P_{ij}) \quad (3)$$

- **Gradient:** The gradient of the loss with respect to the difference in scores is particularly simple

$$\frac{\partial L}{\partial(s_i-s_j)} = P_{ij} - \bar{P}_{ij} \quad (4)$$

- **Deep Relevance Matching Model (DRMM)**

$$S(Q,D) = \sum_{t\in Q} g(t).F(H(M(t,D))) \quad (5)$$

DRMM is an interaction-focused model that computes a matching score between a query and a document. It uses a term gating network to assign importance to each query term. The overall matching score for a document D and query Q is given by the weighted sum of matching signals for each query term.

Algorithm: Neural Re-ranking algorithm

```
-------------------------------------------------------------------------

def neural_re_ranking(query, initial ranking, user interactions, counterfactual_signals=None):

    # Step 1: Preprocess input data

    query_embedding = preprocess_query(query)

    articles_embeddings = preprocess_articles(initial_ranking)

        # Step 2: Generate initial ranking

    initial_ranking_scores = generate_initial_ranking(query, initial ranking)

    # Step 3: Extract features (content, context)

    article features = extract features (articles embeddings, query embedding, user interactions)

    # Step 4: Neural Re-Ranking Model (Observed Signals)

    relevance scores = neural_ranking_model(article features)

    # Step 5: Loss function and model training (pairwise loss / listwise loss)

    loss = calculate loss (relevance scores, user interactions)

    train model(loss)

    # Step 6: Counterfactual Learning (if applicable)

    if counterfactual_signals:

        counterfactual_relevance_scores = counterfactual_ranking(counterfactual_signals)

        refine_model(counterfactual_relevance_scores)

    # Step 7: Refine Ranking based on the learned model

    refined_ranking = refine_ranking(relevance_scores)

    # Step 8: Multi-objective Optimization (Diversity, Fairness)

    final_ranking = apply_multi_objective_optimization(refined_ranking)

    return final_ranking

-------------------------------------------------------------------------
```

IV. Multiple Objectives

As the primary objective, accuracy is considered the most important for recommender systems. Based on existing literature, focusing solely on accuracy tends to increase correlation or redundancy among recommendations, which can limit the overall usefulness of suggestions offered to users and lead to echo chamber phenomena (Ge et al., 2020). To address this hindrance, many studies focus on multi-objective optimization, aiming to improve accuracy while also considering supplementary metrics such as fairness and diversity. Balancing these competing objectives is one of the most significant issues because improvements in diversity and fairness can often be at odds with accuracy (Liu et al., 2019). This study focuses on neural re-ranking techniques that aim to balance multiple objectives simultaneously, with a particular emphasis on those related to diversity and fairness.

Regarding re-ranking, diversity is typically explained as enhancing the dissimilarity of items in the list to ensure broad coverage of the user's domain of interest. Unlike classical non-learning techniques such as Maximal Marginal Relevance (MMR) (Carbonell and Goldstein, 1998), neural models that incorporate diversity at lower levels of the hierarchy employ an end-to-end training paradigm, which eliminates the need for handcrafted features of relevance or diversity. Features in methods to address this problem can be classified into two categories: implicit and explicit. Implicit methods calculate diversity by using some form of inter-item dissimilarity, without requiring other subtopic labels. For example, neural tensor networks (NTN) were designed to learn pairwise dissimilarities, which were then used in relevance and diversity sequential re-ranking (Pasumarthi et al., 2016). MDP-based models, such as MDP-DIV, utilize diversity metrics like α-DCG and optimize them using policy gradient methods. Some enhancements, such as M2DIV, which utilizes recurrent networks and ahead Monte Carlo Tree Search, aim to improve long-term ranking reward outcomes (Su et al., 2024; Zou et al., 2019). More recently, DALETOR developed smooth approximations of diversity metrics and applied self-attention to better model item relationships in the tasks (Yu et al., 2023).

Explicit techniques aim at maximizing coverage over relationships defined a priori on topics or item classes. For instance, DSSA models relevance and diversity through subtopic-level attention, and so does DVGAN by posing the generation of diverse lists as a minimax game between a generator and a discriminator, assessing relevance and diversity simultaneously (Carraro & Bridge, 2024; Liu et al., 2020a; Nejad, 2015). Similarly, DESA utilizes an encoder-decoder self-attention model to correlate items and subtopics (Deng et al., 2020). Airbnb search and other use cases have showcased those metrics designed to evaluate the distance from the ranked list to the suggestion, illustrating the application of diversity-aware re-ranking, as neural nets are used to produce non-redundant recommendations (Abdool et al., 2020). The interplay between precision and diversity or the lack thereof, is usually controlled by optimizing a trade-off parameter defined a priori to weigh the objectives or combining them into a single metric like α-NDCG, which is known to assess both within one framework (Pasumathi et al., 2016; Yu et al., 2023).

V. Applications for Emergencies

Various Applied research with commercial potential cases have recently adopted neural rankers, increasing their scope beyond traditional recommendation contexts. An area of research that is dedicated to the challenge of recommending an eclectic collection from multiple sources or channels is integrated re-ranking (mixed re-ranking). For example, delivering a feed combining articles, videos, and news with diverse characteristics and formats (Xia et al., 2024) is done simultaneously by the recommendation system. The re-ranking model's input is changed from a single list to multiple lists supplied from various channels for boosting relevance. DHANR is a well-known model that addresses cross-channel interactions by using hierarchical self-attention structures with dependencies across many item sources (Calzolari et al., 2022; Xia et al., 2024) proposed the break down the integrated re-ranking problem into two subtasks, such as source selection along with item ranking, and uses hierarchical reinforcement learning to solve it. DEAR uses deep Q-networks to switch between advertisements and organic material, optimizing the balance of different things in the final sorted list (Xu et al., 2023, 2021; Liao et al., 2021) demonstrated that a reinforcement learning model with a cross-channel attention mechanism improves interaction modeling for item categories.

Edge re-ranking is a new method based on edge-to-cloud computing for enhancing suggestion quality using real-time user interactions and modeling on the edge or mobile device. (Gong et al, 2020) Edge-based re-ranking allows real-time user feedback and enhances recommendation accuracy and responsiveness. TheEdgeRec proposed system suggests ranking lists are calculated in the cloud first, whereas re-ranking is done on local mobile devices. This facilitates in-device personalisation and novel prospects in federated learning and privacy-aware recommendation models (Hard et al., 2018). Edge-based re-ranking plays a central role in delivering real-time, context-aware recommendations that are tailored to customers' preferences.

VI. Summary Findings

Greater complexity in recommendation systems has led to the emergence of the new practical application of neural re-ranking methods. Combined re-ranking solves the problem of building a combined list of different content types from multiple heterogeneous sources through hierarchical attention mechanisms and reinforcement learning for simulating cross-channel interactions and enhancing product usability. In parallel, edge re-ranking is an interesting line of future research in bringing some of the computation onto the edge device so that real-time adaptation to changing user preferences can be made and serving up personalized, privacy-critical models through methods like federated learning. All of these novel applications show the scalability and versatility of neural reranking. This is exciting research in efficiency, contextualization, and integration across multiple sources. These regions have the potential to enhance the quality of recommender systems for real-world use. Re-ranking and edge re-ranking are two cutting-edge neural approaches that address part of the issues of recommender systems.

Integrated re-ranking applies item combination, which combines articles, videos, and news feeds from multiple sources and integrates them into one ranked output list. Integrated re-ranking solves ordinal cross-channel interaction complexity using hierarchical attention models and reinforcement learning, and therefore is appropriate for systems combining multiple content types. Edge re-ranking enables responsive, fast real-time re-ranking on a user's preferences by offloading some processing to the user device. This real-time responsiveness is built on more customisation, but at the expense of user privacy. Federated learning paradigms and on-device models promote privacy but raise problems with regard to limited computing powers on edge devices, which result in heavy-duty privacy guards to be negotiated. Re-ranking integrated addresses multi-source diversity and diversification scalability, whereas edge re-ranking emphasizes agility and privacy. These activities supplement each other in the delivery of hybrid, context-sensitive systems that are able to adapt to perform in various environments.

TABLE I MODEL EVALUATION

Model	Threshold	mAP	Recall	Precision	F1	Wikipedia-NQ	MS-MARCO	Hotpot-QA
Cross-Encoder (CE)	0.9	0.64	1	0.36	0.53	0.47	0.91	0.32
Cross-Encoder (CE)	0.7	0.64	1	0.29	0.45	0.45	0.89	0.3
Cross-Encoder (CE)	0.5	0.64	0.79	0.29	0.42	0.44	0.85	0.29
Cross-Encoder (CE)	0.3	0.64	0.71	0.29	0.41	0.42	0.8	0.28
ColBERT	0.9	0.6	1	0.45	0.58	0.49	0.91	0.34
ColBERT	0.7	0.58	0.98	0.4	0.54	0.49	0.9	0.31
ColBERT	0.5	0.56	0.96	0.36	0.5	0.47	0.87	0.28
ColBERT	0.3	0.54	0.94	0.34	0.48	0.46	0.82	0.26
Dense Passage Retrieval (DPR)	0.9	0.63	1	0.48	0.6	0.48	0.89	0.32
Dense Passage Retrieval (DPR)	0.7	0.61	0.98	0.42	0.56	0.48	0.89	0.31
Dense Passage Retrieval (DPR)	0.5	0.59	0.96	0.38	0.52	0.46	0.87	0.3
Dense Passage Retrieval (DPR)	0.3	0.57	0.94	0.34	0.5	0.46	0.85	0.24
EDRR (U≤0.1)	0.9	0.73	1	0.37	0.55	0.52	0.94	0.35
EDRR (U≤0.1)	0.7	0.73	1	0.37	0.54	0.52	0.93	0.34
EDRR (U≤0.1)	0.5	0.73	1	0.37	0.54	0.52	0.92	0.33
EDRR (U≤0.1)	0.3	0.73	1	0.37	0.54	0.5	0.9	0.32

Table I illustrates Evidential Document Re-Ranking (EDRR), a novel three-stage retrieval architecture that combines evidence-based training, uncertainty-driven active learning, and re-ranking using uncertainty filters and relevance scores. Our experiments on the Wikipedia-NQ dataset demonstrate that mAP@10 improves incrementally from 0.62 with evidence-based training to 0.69 with active learning and 0.73 with uncertainty-based re-ranking. On all tested datasets, EDRR outperformed strong baselines such as Cross-Encoder, ColBERT, and DPR.

The addition of uncertainty estimation provides better ranking calibration and accuracy, as well as increased transparency and interpretability, both of which are essential in high-risk or uncertain retrieval applications. Active learning also allowed the model to learn better from underrepresented or difficult data by highlighting informative, high-uncertainty samples. Our findings convincingly validate the use of uncertainty-aware information retrieval methods because they provide more relevance and reliability.

The results show that the union of active learning and uncertainty estimation has extremely high potential to boost the adaptability and reliability of retrieval-augmented question answering generation systems. We noticed that adding evidential learning and uncertainty estimates improves re-ranking performance. However, EDRR is still prone to performance decreases in cases with excessively noisy or sparse data, but to a smaller extent than typical machine learning algorithms. Adopting strong data augmentation techniques as well as more sophisticated preprocessing and data selection tactics can help to mitigate these problems. Notably, the sole difference between the EDRR model and normal classification models is its optimization function. The underlying network architecture and inference techniques remain unaltered. As a result, expanding to larger datasets increases processing time at the same pace as classical classifiers. Incorporating evidentiary deep learning into calibration has made re-ranking performance more consistent across datasets, which in turn has reduced performance variation from dataset shifts and made the retrieval system more robust.

TABLE II PERFORMANCE OF RETRIEVERS

Category	Model	MRR	HasPositives@k
Classical	TF-IDF	0.681	0.593
Classical	LMDirichlet	0.799	0.77
Classical	BM25	0.817	0.785
Neural	sparse docT5query	0.786	0.754
Term-based	ColBERT	0.765	0.708
Document-level	SentenceBERT	0.669	0.592
Neural	DPR	0.624	0.547
Proposed	EDRR	0.73	0.69

Table II shows the findings of the retriever models. We do not give latency performance since the document corpus is too small to detect significant variations across the chosen models. While not approaching BM25 performance, neural retrievers have made significant improvements, surpassing the TF-IDF baseline and performing comparably to the LM Dirichlet model. The sparse model docT5Query is the first runner-up and shows significant improvements over BM25, on which it is based. In other words, we believe that expanding fact-checked documents with artificially generated queries and then indexing them using standard techniques (BM25 in our case) is an effective approach because the generation process clearly extracts the subjects, topics, and/or events, increasing the likelihood of detecting matching queries citing those concepts. Unfortunately, we demonstrate how the query generation technique, which relies on the T5 transformer, is computationally costly and may not be viable in online applications where the document corpus is often changed. ColBERT, on the other hand, obtains impressive results without the need for any preprocessing. Furthermore, the late interaction method it employs between query and document words appears to be efficient enough to scale with millions of corpora. Finally, document-level neural retrievers (SentenceBERT and DPR) lag behind the other approaches, most likely because representing the entire document/query with a single embedding result in a coarse representation that lacks the details required to infer the relationship between the claim and its verified document. Concretely, fact-checked publications are typically longer texts that cite multiple concepts and entities to assess claim truthfulness. In such cases, it is difficult to produce a meaningful representation based on the entire document rather than more granular information (e.g., text terms and/or phrases).

To summarize, recent advances in neural information retrieval appear to be closing the gap with traditional retrieval systems, but we have demonstrated that even the most advanced retrievers cannot replace them in practice. Furthermore, merging the two approaches and developing more efficient interaction methods shows promise for enhancing performance. The suggested EDRR model achieves encouraging results with a mAP@10 of 0.73, leveraging uncertainty-based re-ranking to improve retrieval quality, but it still struggles to handle excessively noisy or sparse datasets.

VII. Conclusion

We discussed how modern recommender systems use neural re-ranking techniques. Accuracy has significantly improved from the simplistic univariate approaches to sparse feature set models due to more advanced interactions modeled in listwise approaches. It is intriguing to observe the shift from single-objective, accuracy-centric models to multi-objective models that prioritize accuracy, diversity, and fairness. This indicates a growing emphasis on both user satisfaction and ethical dimensions. Furthermore, new integrated re-ranking for multi-source content and edge re-ranking for real-time personalized recommendations showcase emerging applications, highlighting industrial relevance and real-time adaptiveness as neural re-ranking techniques are seamlessly integrated into various application domains and respond to user demands in real time. These improvements foster increased user interaction, system engagement, effectiveness, and overall diversification, relevance, and agility of recommendations.

VIII. Further Suggestions

Further research should investigate the design of advanced neural architectures capable of simultaneously multitasking on multiple objectives. Developing new strategies for adjusting the balance of trade-offs for accuracy, diversity, and fairness while maintaining real-time functionality would be vital. Additionally, incorporating edge re-ranking with lightweight models and federated learning enhances privacy and mitigates latency issues associated with mobile IoT devices. Integrating accuracy and traceability into undemocratic models will help rebuild trust among users, enabling broader adoption. Ultimately, refining paradigms of counterfactual learning would enhance the utilization of off-policy and simulated user interaction data on the receiver side, thereby increasing the dependability and versatility of recommendation systems across various fields.

References

[1] Abri, R., Abri, S., & Cetin, S. (2022, March). Providing a topic-based LSTM model to re-rank search results. In *Proceedings of the 2022 7th International Conference on Machine Learning Technologies* (pp. 249-254). https://doi.org/10.1145/3529399.3529438

[2] Ahmed, M., & Pandey, S. K. (2024). Digital Innovation Management: A Study of How Firms Generate and Implement Digital Ideas. *Global Perspectives in Management*, *2*(3), 13-23.

[3] Ai, Q., Bi, K., Guo, J., & Croft, W. B. (2018, June). Learning a deep listwise context model for ranking refinement. In *The 41st international ACM SIGIR conference on research & development in information retrieval* (pp. 135-144). https://doi.org/10.1145/3209978.3209985

[4] Bello, I., Kulkarni, S., Jain, S., Boutilier, C., Chi, E., Eban, E., ... & Meshi, O. (2018). Seq2Slate: Re-ranking and slate optimization with RNNs.

[5] Calzolari, N., Huang, C. R., Kim, H., Pustejovsky, J., Wanner, L., Choi, K. S., ... & Na, S. H. (2022, October). Proceedings of the 29th international conference on computational linguistics. In *Proceedings of the 29th International Conference on Computational Linguistics*.

[6] Carbonell, J., & Goldstein, J. (1998, August). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 335-336). https://doi.org/10.1145/290941.291025

[7] Carraro, D., & Bridge, D. (2024). Enhancing recommendation diversity by re-ranking with large language models. *ACM Transactions on Recommender Systems*. https://doi.org/10.1145/3700604

[8] Conti, V., Rundo, L., Militello, C., Mauri, G., & Vitabile, S. (2017). Resource-Efficient Hardware Implementation of a Neural-based Node for Automatic Fingerprint Classification. *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, *8*(4), 19-36.

[9] David Winster Praveenraj, D., Prabha, T., Kalyan Ram, M., Muthusundari, S., & Madeswaran, A. (2024). Management and Sales Forecasting of an E-commerce Information System Using Data Mining and Convolutional Neural Networks. *Indian Journal of Information Sources and Services*, *14*(2), 139–145. https://doi.org/10.51983/ijiss-2024.14.2.20

[10] Deng, Z., Dou, Z., Zhu, Y., & Wen, J. R. (2025). A Model-agnostic Pre-training Framework for Search Result Diversification. *ACM Transactions on Information Systems*. https://doi.org/10.1145/3764662

[11] Ganesh, R., & Sivakumar, D. R. (2021). Diagnosis of Brain Tumor Using Artificial Neural Network. *Int. Acad. J. Innov. Res.*, *8*(1), 06-10. https://doi.org/10.9756/IAJIR/V8I1/IAJIR0802

[12] Ge, M., Delgado-Battenfeld, C., & Jannach, D. (2010, September). Beyond accuracy: evaluating recommender systems by coverage and serendipity. In *Proceedings of the fourth ACM conference on Recommender systems* (pp. 257-260). https://doi.org/10.1145/1864708.1864761

[13] Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., ... & Ramage, D. (2018). Federated learning for mobile keyboard prediction. https://doi.org/10.48550/arXiv.1811.03604

[14] Huang, J., Li, Y., Sun, S., Zhang, B., & Huang, J. (2020, October). Personalized flight itinerary ranking at fliggy. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (pp. 2541-2548). https://doi.org/10.1145/3340531.3412735

[15] Jelena, T., & Srđan, K. (2023). Smart mining: joint model for parametrization of coal excavation process based on artificial neural networks. *Archives for Technical Sciences*, *2*(29), 11-22. https://doi.org/10.59456/afts.2023.1529.011T

[16] Kim, J., Kim, K., Jeon, G. Y., & Sohn, M. M. (2022). Temporal Patterns Discovery of Evolving Graphs for Graph Neural Network (GNN)-based Anomaly Detection in Heterogeneous Networks. *J. Internet Serv. Inf. Secur.*, *12*(1), 72-82. https://doi.org/10.22667/JISIS.2022.02.28.072

[17] Li, Q., Li, L., Lin, J., & Zhong, W. (2025, July). Towards Principled Learning for Re-ranking in Recommender Systems. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 3030-3034). https://doi.org/10.1145/3726302.3730257

[18] Li, Y., Zhu, J., Liu, W., Su, L., Cai, G., Zhang, Q., ... & He, X. (2022, April). Pear: Personalized re-ranking with contextualized transformer for recommendation. In *Companion Proceedings of the Web Conference 2022* (pp. 62-66). https://doi.org/10.1145/3487553.3524208

[19] Liu, J., Dou, Z., Wang, X., Lu, S., & Wen, J. R. (2020, July). DVGAN: A minimax game for search result diversification combining explicit and implicit features. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 479-488). https://doi.org/10.1145/3397271.3401084

[20] Liu, W., Guo, J., Sonboli, N., Burke, R., & Zhang, S. (2019, September). Personalized fairness-aware re-ranking for microlending. In *Proceedings of the 13th ACM conference on recommender systems* (pp. 467-471). https://doi.org/10.1145/3298689.3347016

[21] Liu, W., Liu, Q., Tang, R., Chen, J., He, X., & Heng, P. A. (2020, October). Personalized Re-ranking with Item Relationships for E-commerce. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (pp. 925-934). https://doi.org/10.1145/3340531.3412332

[22] Nejad, N. D. (2015). Diagnosis of heart disease and stomach hyperacidity through iridology using a neural network. *International Academic Journal of Science and Engineering*, *2*(1), 120–128.

[23] Pasumarthi, R. K., Bruch, S., Wang, X., Li, C., Bendersky, M., Najork, M., ... & Wolf, S. (2019, July). Tf-ranking: Scalable tensorflow library for learning-to-rank. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2970-2978). https://doi.org/10.1145/3292500.3330677

[24] Pei, C., Zhang, Y., Zhang, Y., Sun, F., Lin, X., Sun, H., ... & Pei, D. (2019, September). Personalized re-ranking for recommendation. In *Proceedings of the 13th ACM conference on recommender systems* (pp. 3-11). https://doi.org/10.1145/3298689.3347000

[25] Shi, S., Caluyo, F., Hernandez, R., Sarmiento, J., & Rosales, C. A. (2024). Automatic Classification and Identification of Plant Disease Identification by Using a Convolutional Neural Network. *Natural and Engineering Sciences*, *9*(2), 184-197. https://doi.org/10.28978/nesciences.1569560

[26] Su, Z., Dou, Z., Zhu, Y., & Wen, J. R. (2024). Passage-aware search result diversification. *ACM Transactions on Information Systems*, *42*(5), 1-29. https://doi.org/10.1145/3653672

[27] Suji, M. A. M., & Kumar, R. A. (2022). Leukaemia, Convolutional Neural Networks, White Blood Cell, Classification, Image Extraction, Machine Learning, Comparison. *International Journal of Advances in Engineering and Emerging Technology*, *13*(1), 19-30.

[28] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998–6008.

[29] Xia, C., Shi, X., Xie, H., Liu, Q., & Shang, M. (2024, July). Hierarchical Reinforcement Learning for Long-term Fairness in Interactive Recommendation. In *2024 IEEE International Conference on Web Services (ICWS)* (pp. 300-309). IEEE. https://doi.org/10.1109/ICWS62655.2024.00052.

[30] Xu, Y., Shen, Q., Yin, J., Deng, Z., Wang, D., Chen, H., ... & Ge, J. (2023, August). Multi-channel Integrated Recommendation with Exposure Constraints. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 5338-5349). https://doi.org/10.1145/3580305.3599868

[31] Yu, P., Rahimi, R., Huang, Z., & Allan, J. (2023, October). Search result diversification using query aspects as bottlenecks. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (pp. 3040-3051). https://doi.org/10.1145/3583780.3615050

[32] Zou, L., Xia, L., Ding, Z., Yin, D., Song, J., & Liu, W. (2019, April). Reinforcement learning to diversify top-n recommendation. In *International Conference on Database Systems for Advanced Applications* (pp. 104-120). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-18579-4_7