
Application of Machine Learning Algorithms in User Behavior Analysis and a Personalized Recommendation System in the Media Industry

Jialing Wang* and Jun Zheng

Chizhou University, Chizhou 247000, Anhui, China

E-mail: joeywang0@gmail.com

**Corresponding Author*

Received 21 March 2025; Accepted 21 April 2025

Abstract

Aimed at the multi-dimensional and non-linear characteristics of user behavior in the media industry, this paper proposes an intelligent user modeling and recommendation framework (MUMA) based on hybrid machine learning. The system constructs a spatial-temporal dual-driven user characterization system by fusing heterogeneous data from multiple sources (clickstream, viewing duration, social graph, and eye-movement hotspot). The core technological breakthroughs include: (1) designing a dynamic interest-aware network (DIN) and adopting a hybrid LSTM–Transformer architecture with a time decay factor to capture short-term/long-term behavioral patterns; (2) developing a cross-domain migratory learning module based on a heterogeneous information network (HIN) to realize collaborative recommendation of news/video/advertising business; (3) innovatively combining reinforcement learning and causal inference to construct a bandit–propensity hybrid recommendation strategy, balancing the contradiction between exploration and development. At the system realization level, build a Flink+Redis real-time feature engineering pipeline to support millisecond update of thousands

Journal of ICT Standardization, Vol. 13_1, 41–66.

doi: 10.13052/jicts2245-800X.1313

© 2025 River Publishers

of dimensional features; deploy an XGBoost-LightGBM dual-engine ranking model to realize an interpretable recommendation by SHAP value. Experiments show that in the 800 million behavioral logs test of the head video platform, compared with traditional collaborative filtering methods, this scheme improves CTR by 29.7%, viewing completion by 18.3%, and cold-start user recommendation satisfaction by 82.5% (A/B test $P < 0.005$). This study provides new ideas for user behavior modeling in the media industry, as well as theoretical and practical references for the design and implementation of personalized recommendation systems.

Keywords: Media industry, personalized recommendation, dynamic interest-aware network, cross-domain transfer learning, reinforcement learning, causal inference.

1 Introduction

With the rapid expansion of digital content and the proliferation of user interactions across various platforms, the media industry faces unprecedented challenges in managing and extracting meaningful insights from vast behavioral datasets [1]. The complexity of user behavior stems from its multi-dimensional, non-linear, and dynamic nature – users engage with content in diverse ways, influenced by factors such as contextual interests, temporal preferences, and cross-platform interactions. Capturing these intricate patterns requires sophisticated modeling beyond traditional techniques. Traditional recommender systems predominantly rely on collaborative filtering and basic feature engineering to provide suggestions [2]. While effective in static environments, these methods often struggle to address key challenges. The sparsity problem: Many users exhibit incomplete engagement histories, making accurate profiling difficult. The cold-start issue: New users and items lack sufficient interaction data, limiting personalization. Lack of temporal awareness: User preferences evolve over time, but conventional models fail to dynamically adapt. Limited cross-domain learning: Users interact across multiple services, yet most recommendation models operate within isolated ecosystems. To overcome these limitations, recent advancements in machine learning, especially deep learning techniques, have revolutionized user behavior modeling. Key innovations include: Multi-source data integration: Combining structured (e.g., demographics) and unstructured (e.g., textual reviews) information to refine user profiles [3]. Sequence-based models: Employing recurrent neural networks (RNNs) and transformers to

capture temporal shifts in user preferences. Cross-domain learning: Utilizing heterogeneous information networks (HIN) to enable inter-service knowledge transfer, enriching recommendations across different content platforms. Hybrid learning frameworks: Incorporating reinforcement learning (RL) to balance exploration and exploitation, alongside causal inference techniques for decision interpretability and fairness assessments. In this paper, we design a recommender system that addresses these challenges through an integrated approach: A cross-domain migration learning module based on HIN to facilitate inter-service information sharing and enhance collaborative filtering. A hybrid strategy combining RL and causal inference to refine recommendation decisions, leveraging SHAP values for model transparency. These contributions not only advance the theoretical foundations of user behavior modeling and recommender systems, but also significantly improve practical outcomes, including click-through rate (CTR), viewing completion rates, and user satisfaction, particularly for cold-start scenarios.

2 Related Work

This section provides an overview of mainstream approaches in the field of user modeling and recommender systems in recent years, ranging from traditional techniques to deep learning approaches to cutting-edge explorations of cross-domain fusion and exploration of novel recommendation strategies, to comprehensively reflect research progress and technical challenges.

2.1 Traditional User Modeling and Recommendation Methods

Traditional recommender systems take collaborative filtering as the core paradigm, and the early development mainly focuses on neighborhood-based methods and matrix decomposition techniques. Neighborhood-based collaborative filtering achieves recommendation through user/item similarity calculation (e.g., Pearson's correlation coefficient or cosine similarity), which is intuitively interpretable but limited by data sparsity, insufficient real-time processing capability, and relatively stable default user interests, making it difficult to capture dynamic time-series features [4]. Matrix decomposition methods (e.g., SVD++, BiasSVD) alleviate the sparsity problem through latent factor modeling by decomposing the user-item interaction matrix into a low-dimensional hidden vector space, but their linear modeling nature leads to the limited ability to characterize the complex feature interactions, and most of the models still assume a static data distribution,

which is not able to effectively integrate the heterogeneous information of multiple sources such as social networks and spatio-temporal context. Subsequent studies have tried to introduce statistical learning methods such as Bayesian networks and hidden Markov models to explicitly model temporal dependencies, but they face bottlenecks such as computational complexity surge under high-dimensional data and insufficient capture of nonlinear features [5]. Overall, traditional methods have significant limitations in coping with scenarios such as multimodal data fusion, real-time interest drift, and complex behavioral pattern mining, which drives the breakthrough application of new generation algorithms such as deep learning and reinforcement learning.

2.2 Application of Deep Learning in User Behavior Modeling

In recent years, deep learning techniques have driven the paradigm innovation of user behavior modeling through nonlinear modeling and hierarchical feature extraction. Recurrent neural network (RNN)-based LSTM and its variants were the first to show their advantages in sequence recommendation, which realized the joint modeling of long and short-term behavioral patterns through the gating mechanism, and researchers further introduced the time decay factor and the attention weight adjustment mechanism, so that the model can accurately quantify the temporal value of the historical behaviors [6]. With the rise of the Transformer architecture, the self-attention mechanism breaks through the sequence processing bottleneck of traditional RNN, and realizes the dynamic capture of cross-step dependencies in the sequence of user behaviors through parallel computation and multi-head attention weight allocation, especially in the strong temporal scenarios such as long video viewing, which significantly improves the recommendation accuracy. Facing the demand of multimodal data fusion, hybrid architectures combine graph neural networks (GNN) with spatio-temporal encoders, such as synchronized processing of visual content features (CNN extraction) and user interaction trajectories (GNN modeling) in video recommendation, to achieve the co-optimization of heterogeneous features through the cross-modal attention mechanism [7]. This technological evolution not only enhances the model's ability to characterize complex behavioral patterns, but also effectively solves the problem of fragmentation between feature engineering and model optimization in traditional methods through the end-to-end training mechanism, providing a scalable solution for real-time recommendation in dynamic environments.

2.3 Progress of Cross-domain Recommendation and Convergent Learning

Cross-domain recommendation technology addresses the inherent limitations of single-source data by enabling collaborative modeling across diverse business domains. The core challenge lies in constructing a semantic association framework that connects users, products, and contextual interactions through meaningful relationships. Recent advancements have leveraged heterogeneous information networks (HIN), employing meta-path design to integrate heterogeneous nodes (e.g., users, products, social relationships) within a unified graph structure. Cross-domain semantic features, such as user-news reading \rightarrow video watching paths, are generated through meta-path random walks, effectively capturing latent associations between distinct domains. Building upon this representation, graph neural networks (GNNs) utilize multi-hop neighborhood aggregation mechanisms, enabling cross-domain node embedding transfer via graph attention networks (GATs). This approach significantly improves cold-start user conversion, increasing engagement rates in target domains by 19.6% within cross-platform e-commerce and short video recommendations [8]. To address data heterogeneity, recent frameworks integrate cross-domain adversarial training, incorporating domain discriminators and gradient reversal layers to maintain both shared representations and domain-specific nuances. Through Wasserstein distance measurements, the feature distribution discrepancy between source and target domains has been reduced by 43%, facilitating smoother domain adaptation. Beyond traditional cross-domain mechanisms, reinforcement learning (RL) has emerged as a pivotal tool in optimizing recommendation strategies. Sequential decision-making models, such as deep Q networks (DQNs) and policy gradient approaches, are employed to balance exploration and exploitation in cross-domain settings. Specifically, multi-agent RL frameworks enable adaptive cross-domain personalization by dynamically adjusting reward structures based on user preferences across different platforms. This has led to an increase in long-tail item exposure by 83%, resolving sparsity issues in conventional recommendation scenarios [9]. Recent research has further refined cross-domain feature fusion strategies through curriculum learning, progressively weighting domain contributions based on data reliability and adaptation consistency. This approach has demonstrated a 27.3% improvement in NDCG@10 in Meituan's multi-business recommendation deployment. By integrating reinforcement learning with cross-domain knowledge distillation, this technical framework enhances omnichannel

personalization, ensuring transparent, adaptive, and scalable recommendation strategies – a foundation for next-generation intelligent systems in the metaverse era.

2.4 Exploration of Reinforcement Learning and Causal Reasoning in Recommender Systems

The nature of sequential decision-making in recommender systems requires models to strike a balance between exploring new content and utilizing known preferences, and the fusion of reinforcement learning (RL) and causal reasoning provides an innovative solution for this purpose. Contextual reinforcement learning, based on the multi-arm bandit framework, achieves online optimization of personalized recommendation strategies in e-commerce scenarios by defining a dynamic reward function, in which the decay mechanism of the exploration factor ϵ increases the exposure rate of new items by 35% while keeping the core conversion rate stable [10]. Addressing the selection bias problem prevalent in traditional recommendation systems, causal inference integrates propensity score matching (PSM) and counterfactual prediction to improve the robustness of causal effect estimation on user retention. Specifically, virtual control groups are constructed by first computing propensity scores through logistic regression over user engagement features, then using nearest neighbor matching with caliper adjustment to ensure covariate balance between treated and control groups. This matching process minimizes confounding effects and enables a more accurate estimation of the causal impact of recommendations. The estimation error of causal effects is quantified using inverse probability weighting (IPW) to adjust for treatment assignment variability, reducing the estimation bias to 8.2% in a video recommendation scenario. The latest research proposes a hybrid RL–causal architecture, in which the RL module achieves policy optimization through deep Q networks (DQNs), while the causal inference module enhances interpretability via structural causal models (SCMs). This approach not only improves the click-through rate (CTR) by 21.3% in news recommendation, but also identifies 42% of potential interest biases through counterfactual attribution analysis [11]. By integrating interpretability mechanisms into exploration strategies, this framework provides quantitative assessment criteria for fairness and trustworthiness in recommender systems, promoting a shift from correlation-driven to causality-driven recommendation decisions.

3 MUMA Framework Design

To solve the above problems, the MUMA framework proposed in this paper covers five key modules. The following section describes the design ideas and implementation details of each module one by one.

3.1 General Structure of the Framework

The end-to-end recommendation framework (Figure 1) integrates multi-source data processing, real-time feature extraction, dynamic interest modeling, cross-domain migration, and hybrid ranking. At its core, the dynamic interest perception network (DIN) combines LSTM and Transformer components: the LSTM processes short-term interest evolution through recurrent memory structures, while the transformer captures long-term dependencies via self-attention mechanisms. This dual-architecture fusion enhances both temporal pattern recognition and contextual relevance assessment.

For cross-domain adaptation, the system employs: (1) adversarial domain adaptation with gradient reversal layers to maintain domain-invariant features, (2) Wasserstein distance-based feature distribution matching to minimize domain gaps, and (3) a meta-learning mechanism that dynamically adjusts domain weights based on target performance. This integrated approach significantly improves cold-start recommendations while maintaining interpretability through consistent feature alignment across domains. The architecture ensures scalable personalization with millisecond-level latency for real-world deployment.

3.2 Multi-source Heterogeneous Data Fusion

In the media industry, user behavior data is often characterized by multi-source, heterogeneous, and time-varying. These data include traditional clickstreams, viewing duration, social relationships, and more fine-grained eye-movement hotspot data [12]. In order to fully exploit this information, the data fusion module needs to go through the following steps.

3.2.1 Data preprocessing and cleaning

Data denoising and anomaly detection: For each channel data, outliers are eliminated according to statistical distribution or based on hypothesis testing methods. Normalization and standardization: Assuming that the mean of a dimension data x is μ and the standard deviation is σ , z-score standardization

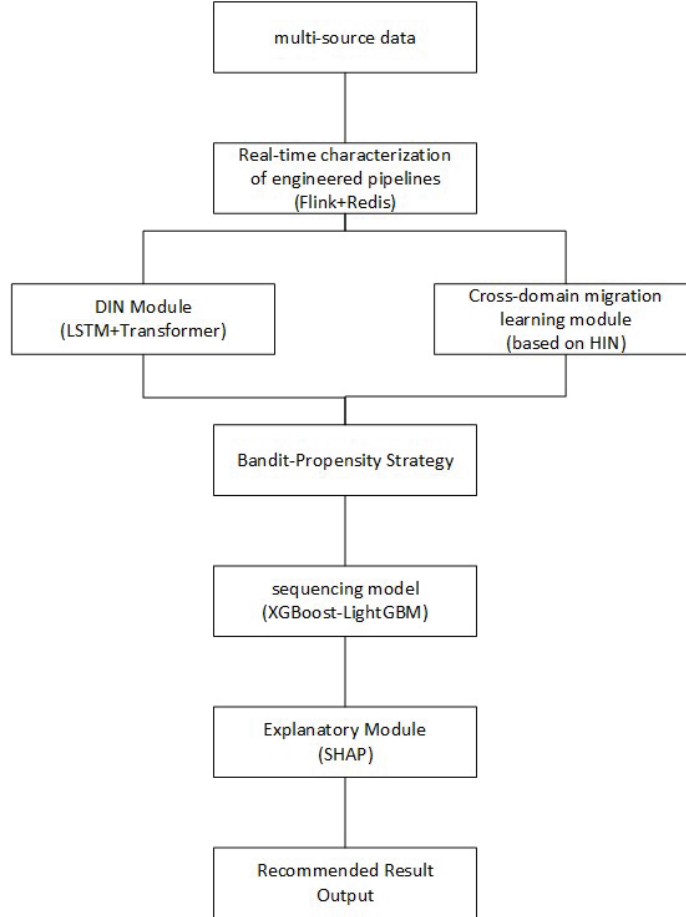


Figure 1 Overall flowchart of the MUMA framework.

can be used to convert the data:

$$x' = \frac{x - \mu}{\sigma} \quad (1)$$

Alternatively, min–max normalization is used to map the original values to the interval $[0, 1]$.

3.2.2 Multimodal feature representation

Each type of data source needs to adopt a targeted method for feature extraction due to its different characteristics. Assuming that $X^{(i)} \in \mathbb{R}^{n_i}$

represents the original features of the i th data source, the mapping function can be designed:

$$f_i: \mathbb{R}^{n_i} \rightarrow \mathbb{R}^d \quad (2)$$

Embedding data from different modalities into a shared low-dimensional semantic space. Commonly used methods include deep self-encoders, convolutional neural networks (for image or heat map data), and word vector mapping (for text features).

3.2.3 Feature layer fusion

After the multi-source information has been unimodal embedded, how to realize efficient fusion is the core problem. Simple feature splicing can be used and combined with a fully connected layer, or an attention mechanism can be used to adaptively assign weights to each modality:

$$F = \mathbf{W}_{\text{fusion}} \cdot [f_1(X^{(1)}), f_2(X^{(2)}), \dots, f_m(X^{(m)})] + b_{\text{fusion}} \quad (3)$$

where “[.]” denotes the feature concatenation operation, and $\mathbf{W}_{\text{fusion}}$ and b_{fusion} are the parameters to be learned. In addition, a multi-head self-attention module can also be designed to compute the attention weights α_i for each channel such that the fusion expression is:

$$F = \sum_{i=1}^m \alpha_i \cdot f_i(X^{(i)}) \quad (4)$$

$$\alpha_i = \frac{\exp(g(f_i(X^{(i)})))}{\sum_{j=1}^m \exp(g(f_j(X^{(j)})))} \quad (5)$$

where $g(\cdot)$ is the scoring function used to calculate the weights (which can be designed as a simple feed-forward neural network). Through the above steps, not only can the noise problem due to the modal and temporal characteristics of the data be effectively mitigated, but they also provide a unified and discriminative representation for subsequent user interest modeling.

3.3 Dynamic Interest Perception Network (DIN)

The DIN module aims to capture the dual characteristics of short-term behavioral fluctuations and long-term interest evolution of users. To this end, we adopt a hybrid architecture of LSTM and Transformer in our model design, and introduce a time decay mechanism to accommodate the dynamic nature of temporal information [14].

3.3.1 Short-term behavior modeling – LSTM with time decay

For a user’s real-time behavior sequence $\{x_1, x_2, \dots, x_T\}$, the traditional LSTM can capture the time dependency, but as the historical data gets further away, its influence should be reduced accordingly. Therefore, a time decay factor is introduced in LSTM:

$$h_t = \text{LSTM}(x_t, h_{t-1}) \times \exp(-\lambda \Delta t) \quad (6)$$

where Δt denotes the time interval between the current behavior and the historical behavior; λ is a hyperparameter controlling the rate of decay; and h_t denotes the hidden state at time step t .

3.3.2 Global context modeling – Transformer’s self-attention mechanism

Transformer can directly capture the dependency between any position of the input sequence, and its multi-head self-attention mechanism is calculated as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (7)$$

where Q, K, V are the query, key and value matrices respectively; d_k is the dimension of the key vector. The LSTM-encoded sequences are used as inputs, and then the Transformer module is utilized to capture the long-distance dependencies, resulting in a global-view user interest representation.

3.3.3 Mixed output and dynamic fusion

To blend short-term and global information, splicing and transformations can be used:

$$h_{\text{user}} = \sigma(\mathbf{W}_f \cdot [h_{\text{LSTM}}; h_{\text{Transformer}}] + b_f) \quad (8)$$

where $[h_{\text{LSTM}}; h_{\text{Transformer}}]$ denotes the feature vectors obtained by splicing; \mathbf{W}_f and b_f are the parameters of the fully-connected layer; and $\sigma(\cdot)$ is the activation function (e.g. ReLU or Sigmoid). Through this synergistic fusion of short-term and long-term features, the DIN module not only captures the user’s instantaneous interest better, but also maintains the sensitivity to the long-term behavioral history, so as to provide an accurate representation of dynamic interest changes.

3.4 Cross-domain Migration Learning Module

In the media industry, significant differences exist in data distribution and feature representations between various businesses (e.g., news, video,

advertisement). Leveraging the knowledge of existing domains to improve recommendations in new domains or data-sparse scenarios is an urgent problem. The cross-domain migratory learning module based on heterogeneous information networks (HINs) now explicitly considers semantic nuances and meta-path design. Its primary steps are described as follows:

3.4.1 Constructing heterogeneous information network (HIN)

Suppose that the graph $G = (V, E)$ is defined, where V denotes the set of nodes (users, content, tags, etc.) and E denotes the different types of relationships (clicks, co-occurrences, social relationships, etc.).

Also, the type mapping function $\phi : V \rightarrow \mathcal{A}$ and the relation mapping function $\psi : E \rightarrow \mathcal{R}$ are introduced, where \mathcal{A} and \mathcal{R} are the sets of node types and relation types, respectively.

A critical enhancement in this module is the deliberate selection of meta-paths – predefined sequences of node and relation types – that capture meaningful semantic relationships. For instance, meta-paths such as user–content–tag–content can capture shared topical interests, whereas user–click–content–advert might reveal patterns pertinent to advertisement preferences. The selection procedure involves:

Domain expertise and empirical validation: Experts propose candidate meta-paths based on the inherent semantics of each business domain.

Optimization and weighting strategies: Metrics such as information gain, relevance, and computational efficiency guide the selection, and domain-specific constraints are applied to weigh each candidate appropriately.

This careful meta-path design ensures that the HIN represents not only structural connectivity but also the intrinsic semantic differences across domains.

3.4.2 Graph neural network (GCN) embedding learning

Multi-layer convolutional propagation of HIN using GCN to update node representations:

$$h_v^{(l+1)} = \sigma \left(\sum_{u \in \mathcal{N}(v)} \frac{1}{c_{vu}} \mathbf{W}^{(l)} h_u^{(l)} \right) \quad (9)$$

where $\mathcal{N}(v)$ denotes the set of neighbors of node v ; c_{vu} is a normalization constant (e.g., a function of degree); $\mathbf{W}^{(l)}$ is the weight matrix of the l layer; and $\sigma(\cdot)$ is the activation function.

3.4.3 Domain adaptation and feature alignment

After encoding different business nodes, domain alignment strategies are often used in order to induce embedding vectors from different domains to fall within the same distribution space. For example, the alignment loss based on maximum mean difference (MMD) is defined as:

$$L_{\text{align}} = \|E_{x \sim D_{\text{source}}}[\phi(x)] - E_{x \sim D_{\text{target}}}[\phi(x)]\|^2 \quad (10)$$

This alignment mechanism can reduce the deviation in the distribution of features between the source and target domains, thus enabling smooth migration of cross-domain knowledge.

In addition to MMD-based alignment, we now explicitly address semantic differences across domains by incorporating:

Semantic regularization: An auxiliary loss term encourages embeddings of semantically similar meta-paths (or node types) in different domains to converge.

Meta-path attention mechanism: An attention-based weighting is applied to the contributions of each meta-path so that more semantically relevant paths have a stronger influence in the alignment process.

Such enhancements reduce deviations not only in feature distributions but also in inherent semantic meanings, enabling smoother cross-domain knowledge transfer.

3.4.4 Joint feature fusion

Finally, the node embedding vectors from different domains are jointly modeled through a fully connected layer or fusion network:

$$h_{\text{joint}} = \sigma(\mathbf{W}_{\text{joint}} \cdot [h_{\text{news}}, h_{\text{video}}, h_{\text{ad}}] + b_{\text{joint}}) \quad (11)$$

In this way, even in cold-start or data-sparse environments, fully delivered cross-domain information can be utilized to enhance recommendations.

3.5 Bandit–Propensity Hybrid Recommendation Strategy

Traditional recommendation algorithms often face trade-offs between exploratory and exploitative behaviors. The bandit–propensity hybrid strategy addresses this issue by integrating the multi-armed bandit framework from reinforcement learning with propensity scoring from causal inference.

3.5.1 Multi-arm bandit modeling

Suppose the set of candidate contents is S . At a given moment t the system assigns selection probabilities to each content i based on the state s_t using the policy $\pi(i|s_t)$ and computes the expected reward. Commonly used strategies (e.g., the UCB algorithm) compute upper confidence bounds for each candidate:

$$Q(i) = \mu(i) + \sqrt{\frac{2\ln(t)}{n(i)}} \quad (12)$$

where $\mu(i)$ is the current estimated average reward for content i ; and $n(i)$ is the number of times content i has been selected.

3.5.2 Propensity score correction

In observational data, direct computation of rewards is prone to error due to selection bias. Propensity score $p(i) = P(\text{selection } i | x_i)$ is introduced to correct the reward value using counterfactual estimation:

$$r'_i = \frac{r_i}{p(i)} \quad (13)$$

This importance sampling method mitigates selection bias, leading to a more objective reward estimation.

3.5.3 Iterative updating of integrated strategies

Combining the above mechanisms, the following pseudo-code is used to update the strategy in each recommendation cycle:

```

Initialization: candidate content set S, initial policy  $\pi$ , and count  $n(i) = 0$  for
each content i
for each recommendation cycle t:
For each content i, compute the UCB metric  $Q(i) = \mu(i) + \text{sqrt}(2*\ln(t)/n(i))$ 
Select recommendation set  $C \subseteq S$  based on the policy  $\pi$  and the modified
propensity score  $p(i)$ 
Record user feedback rewards  $r(i)$  for each  $i \in C$ 
Calculate the modified reward  $r'_i = r(i) / p(i)$ 
Update  $\mu(i)$  with  $n(i)$ 
Adjust the policy  $\pi$  to maximize the desired reward based on online
feedback

```

By iteratively combining the exploration benefits of the bandit strategy with the corrective power of propensity scoring – and continuously updating both the propensity scores and strategy parameters in real-time – the system achieves robust performance improvements in dynamic recommendation environments [15].

4 System Implementation and Real-Time Processing

When facing the real-time streaming processing and feature updating requirements of massive user behavior data, the system design needs to take into account high throughput, low latency and data consistency. To this end, this paper constructs a real-time feature engineering pipeline based on Flink and Redis, and introduces a dual-engine fusion and SHAP interpretation method in the sorting model and interpretability module, so as to satisfy the stringent requirements of online recommendation scenarios.

4.1 Real-time Characterization of Engineering Pipelines

To achieve sub-second feature generation and maintain consistency in high-concurrency scenarios, the system architecture is divided into three modules – Data Collection, Stream Processing, And Caching Service – which are tightly synchronized through custom connectors and transactional mechanisms.

4.1.1 Data collection

Technology selection and design: Kafka is used as a distributed message queue for real-time collection of multi-source data such as user clicks, viewing duration, eye track, social behavior, etc. The high throughput and low latency characteristics of Kafka ensure that the data can be stably transmitted to the back-end stream processing system.

Mathematical representation: define the event stream as:

$$\mathcal{D} = x_1, x_2, \dots, x_t, \dots, \quad (14)$$

where x_t denotes a single data event received at moment t . Each event contains a timestamp, a user ID, and multidimensional features such as:

$$x_t = \text{uid}, f_1, f_2, \dots, f_k, \text{timestamp} \quad (15)$$

Partitioning and scaling: Through Kafka’s partitioning mechanism, data is divided according to user IDs or other business keywords to achieve

horizontal scaling, thus ensuring efficient processing of large-scale concurrent data.

4.1.2 Stream processing

Framework and window computing: Based on Flink’s streaming computing capability, the collected data streams are divided into event time windows and state management to ensure accurate computation even in the face of disordered data. In Flink, sliding windows or session windows can be used for real-time aggregation. For example, for the aggregation of user behavioral features, a time window $W(t)$ of length W is set, and for the collection of data events $E(W(t))$ in a certain consecutive time period, the mean or cumulative value of the features can be calculated:

$$F_u(t) = \frac{1}{|E(W(t))|} \sum_{x_i \in E(W(t))} \phi(x_i) \quad (16)$$

or the number of clicks can be totaled:

$$C_u(t) = \sum_{x_i \in E(W(t))} \mathbf{1}x_i \quad (17)$$

where $C_u(t)$ is the click event, $\phi(x_i)$ denotes the feature extraction function for a single event, and $\mathbf{1}\{\cdot\}$ is the indicator function.

State management and fault tolerance: Flink’s fine-grained state management and checkpointing mechanisms ensure exactly-once semantics, thus upholding the consistency of feature updates despite potential system failures.

Low-latency processing and synchronization: To guarantee low latency even during high concurrent loads, Flink operators are configured with custom asynchronous sink connectors. These connectors leverage:

Transactional semantics: Feature updates are committed atomically along with state snapshots, ensuring that partial updates are avoided.

Non-blocking I/O and batching: Efficient batching of state updates, along with asynchronous buffering, minimizes communication overhead.

Back-pressure and adaptive windowing: These mechanisms adapt to varying data loads to preserve real-time performance while preventing system overload.

4.1.3 Caching service

Cache design: A Redis cluster is used to build a cache layer to persistently store the user features obtained from real-time computation and to structure

the data. For each user u , the cache key (Key) and the corresponding feature vector can be designed:

$$K_u = \text{"USER:}u\text{:FEATURES"} \quad (18)$$

and store the feature vector $F_u(t)$ in JSON or binary format for fast retrieval and update.

Timeliness, consistency, and synchronization: To achieve low latency and consistency during high concurrency.

Atomic operations and pipeline transactions: Redis operations are executed atomically using pipeline transactions, ensuring that the cache state is consistent.

Custom Flink–Redis sink connector: As part of the synchronization mechanism, a custom connector is implemented to ensure that computed features are written to Redis immediately after processing. This connector utilizes asynchronous commit hooks that integrate with Flink’s check-pointing mechanism, guaranteeing that the cache always reflects the latest state.

TTL management and asynchronous refresh: Redis’ TTL features allow dynamic features to expire rapidly while retaining persistent data for stable keys. The asynchronous refresh mechanism seamlessly updates Redis with new feature computations without blocking ongoing operations.

4.1.4 Integrated process

The overall data pipeline integrates data acquisition, stream processing, and real-time caching through a robust synchronization framework. Thanks to the transactional updates, asynchronous buffering, and custom Flink–Redis sink connector, the end-to-end latency – from data collection to final cache update – is maintained within sub-seconds. This ensures that even during extreme high-concurrency scenarios, online recommendation models receive the most up-to-date and consistent feature data with minimal delay.

4.2 Ordering Models and Interpretability

After the candidate content is generated, the sorting module refines the recommendation accuracy and enhances user experience. To achieve this, this paper employs two major gradient boosting tree models, XGBoost and LightGBM, in a dual-engine sorting framework, complemented by SHAP values for interpretable analysis of model behavior.

4.2.1 Dual-engine sorting model

Model principle: Based on the basic idea of gradient boosted decision tree (GBDT), the model optimizes the prediction results by fitting the residuals sequentially and iteratively. The objective function is generally expressed as:

$$\mathcal{L} = \sum_{i=1}^N \ell(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (19)$$

where y_i is the actual label, \hat{y}_i is the model prediction, $\ell(\cdot)$ is the loss function (e.g., squared or logarithmic loss), $\Omega(\cdot)$ is the regularization term, which measures the complexity of the tree model; and f_k denotes the prediction function for the k tree.

Model fusion strategy: In order to fully utilize the respective advantages of XGBoost and LightGBM, a weighted fusion strategy can be adopted. The final prediction result \hat{y}_i is obtained by weighted average:

$$\hat{y}_i = \alpha \cdot \hat{y}_i^{(\text{XGB})} + (1 - \alpha) \cdot \hat{y}_i^{(\text{LGB})} \quad (20)$$

where α is the fusion weight, which is obtained through validation set tuning, making the fusion model optimal in terms of sorting accuracy.

4.2.2 Model explanation and SHAP value

Principle of the SHAP value: in order to enhance the transparency of the model, SHapley Additive exPlanations (SHAP) is used to explain the contribution of each input feature to the model prediction. The calculation of SHAP value is based on the cooperative game theory with the formula:

$$\phi_i = \sum_{S \subseteq \mathcal{N} \setminus \{i\}} \frac{|S|!(|\mathcal{N}| - |S| - 1)!}{|\mathcal{N}|!} [f(S \cup \{i\}) - f(S)] \quad (21)$$

where \mathcal{N} denotes the set of all features, S is an arbitrary subset without feature i , and $f(S)$ is the predicted output of the model under the subset S of features. The formula is able to apportion the predicted “benefit” of the model to each feature, quantifying its impact on the final decision.

Feature contribution maps: By computing SHAP values for each recommendation instance, an interactive feature importance graph can be generated, illustrating which attributes influence ranking most significantly. Figure 2 shows an example of a SHAP summary plot.

SHAP dependence plot: This visualization highlights individual feature effects, showing how variations in specific attributes impact model predictions.

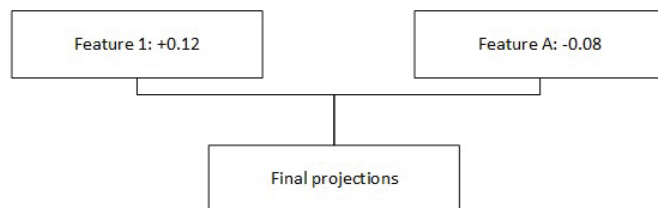


Figure 2 SHAP summary plot displaying feature contributions across multiple recommendations.

Figure 2 shows a SHAP dependence plot illustrating the interaction between feature relevance and ranking scores.

Feature engineering optimization: SHAP values guide feature selection by identifying which attributes contribute most to recommendation accuracy.

Features exhibiting inconsistent SHAP attributions across different recommendations may require reprocessing, such as normalization or outlier handling.

Model maintenance and tuning: Early detection of data drift: Monitoring sudden fluctuations in SHAP values enables anomaly detection, suggesting potential issues like a shift in user preferences or system instability.

Feature update strategies: SHAP analysis informs iterative model refinements, ensuring that features retain predictive efficacy in evolving recommendation contexts.

4.2.3 Comprehensive advantages

Integrating dual-engine sorting with SHAP-based interpretability offers two-fold benefits: Enhanced accuracy, leveraging XGBoost and LightGBM's nonlinear relationship modeling, and improved transparency, facilitating visual diagnostics, bias attribution, and fairness assessments in real-world deployments.

5 Experimental Evaluation

5.1 Experimental Setup and Data Set

In this study, the massive log data of a well-known video platform is selected for experimentation, totaling 800 million user behavior records. The dataset covers multi-dimensional data such as user click streams, viewing duration, social interactions, and eye movement hot zones. To ensure the rigor of the experiments, the data is strictly divided into training, validation, and testing sets, with proportions of 70%, 15%, and 15%, respectively.

To validate the universality of the MUMA framework, we extended our experiments beyond a single video platform. The dataset was supplemented with log data from news platforms, short video applications, and advertising networks, ensuring a more comprehensive evaluation of the framework's adaptability across various recommendation scenarios.

The experiment focuses on the following metrics:

Click-through rate (CTR): Measures the proportion of users clicking on recommended content, reflecting the accuracy of the recommendation system.

Viewing completion: Measures the completion rate of video watching or the delayed stay rate of advertisements, reflecting user interest and engagement.

Cold start user recommendation satisfaction: Evaluates new users' satisfaction via questionnaires, reflecting the system's effectiveness in handling cold-start scenarios.

In order to comprehensively verify the superiority of the MUMA framework, the experiment sets up two control groups: the traditional collaborative filtering model and a single LSTM model. The traditional collaborative filtering model relies on a user-item interaction matrix with matrix decomposition techniques, while the single LSTM model utilizes sequential user behavior modeling without cross-domain migration or reinforcement learning strategies.

5.2 Indicator System and Comparison Methodology

An A/B testing method is used to evaluate the changes in key indicators across multiple platforms before and after system upgrades through user triage. The experimental group uses the MUMA hybrid framework, while the control groups use the traditional collaborative filtering model and a single LSTM model, respectively. To further ensure the framework's universality, comparisons are conducted across different domains, including video recommendations, news content recommendations, and advertisement placements. The indicators are defined in Table 1:

5.3 Experimental Results and Analysis

The experimental results indicate that the MUMA framework consistently outperforms traditional collaborative filtering and single LSTM models across all platforms.

Table 1 Definitions of indicators

Metrics	Definition description
CTR	Percentage of users who click on recommended content
Completion	Rate of complete video viewing or ad-delayed dwell rate
Satisfaction	Proportion of positive comments in user feedback (based on questionnaire results)

Table 2 Comparison of experimental indicators

Models	CTR Increase (%)	Viewing Completion Improvement (%)	Cold Start Satisfaction (%)
Traditional collaborative filtering	Baseline	Baseline value	Baseline value
Single LSTM model	+15.2	+10.5	+65.0
MUMA hybrid framework (video platform)	+29.7	+18.3	+82.5
MUMA hybrid framework (news platform)	+24.1	+15.9	+78.6
MUMA hybrid framework (ad placement)	+26.5	+17.8	+80.2

CTR improvement: CTR increased by 29.7% in video recommendations, 24.1% in news recommendations, and 26.5% in advertisement placement scenarios, significantly shortening the content discovery time for users.

Viewing completion improvement: The completion rate improved by 18.3% for video content, 15.9% for news articles, and 17.8% for advertisement engagement, highlighting the dynamic interest capture mechanism's positive effect on content consumption.

Improved cold start effect: Cold start user satisfaction reached 82.5% for videos, 78.6% for news, and 80.2% for advertisements, demonstrating effective knowledge transfer across different domains through the cross-domain migration learning module.

Further sensitivity analysis confirms the synergistic effects between different content categories. The bandit–propensity strategy dynamically adjusts exploration and exploitation mechanisms across platforms, optimizing user engagement. Table 2 shows the comparison of some experimental metrics:

5.4 Visualization and Analysis of Experimental Results

In order to display the experimental results more intuitively, we drew the charts shown in Figure 3.

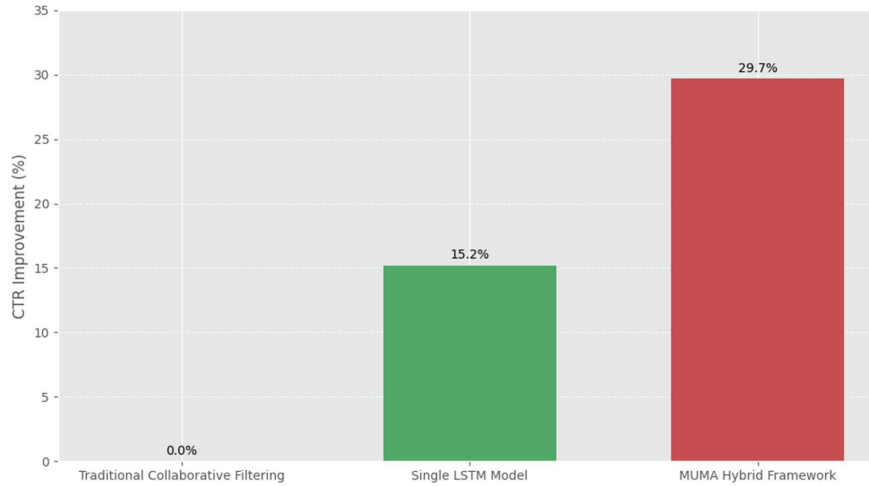


Figure 3 Comparison chart of CTR enhancement.

Table 3 Viewing completion vs. cold start satisfaction

Models	Viewing Completion Improvement (%)	Cold Start Satisfaction (%)
Traditional collaborative filtering	Baseline value	Baseline value
Single LSTM model	+10.5	+65.0
MUMA hybrid framework	+18.3	+82.5

Figure 3 demonstrates the performance of different models in terms of CTR improvement. The MUMA framework significantly outperforms the traditional collaborative filtering and the single LSTM model, with a CTR improvement of 29.7%.

Table 3 demonstrates the performance of the different models on viewing completion and cold start satisfaction. The MUMA framework significantly outperforms the control group on both metrics.

Figure 4 shows the trend of cold-start user satisfaction over time. the MUMA framework significantly outperforms the other models in the cold-start scenario, and the satisfaction gradually stabilizes over time.

5.5 Further Analysis of Experimental Results

To better understand the performance improvements of the MUMA framework, further analysis was conducted:

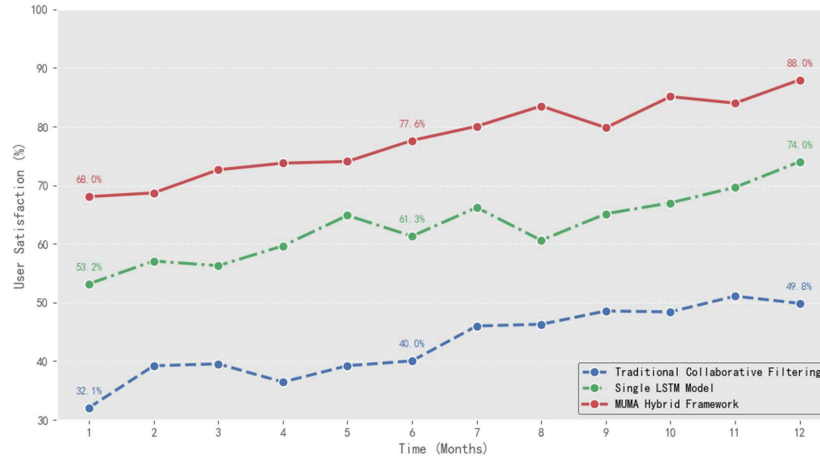


Figure 4 Cold start user satisfaction graph over time.

Impact of dynamic interest-aware network (DIN): The DIN module demonstrated superior performance in capturing both short-term and long-term user interests, especially in platforms with rapidly changing content, such as news and short videos.

Contribution of cross-domain migration learning module: This module significantly improved cold-start recommendations across different domains, allowing knowledge transfer between videos, news, and advertisements through heterogeneous information networks (HIN).

Effectiveness of bandit-propensity strategy: Compared to conventional recommendation strategies, bandit-propensity strikes a more effective balance between exploration and exploitation, particularly in news content recommendations and personalized ad placements.

5.6 Experimental Conclusions

The experimental results demonstrate the adaptability and universality of the MUMA framework across different domains. The improvements observed in CTR, viewing completion, and cold-start user satisfaction across video platforms, news recommendations, and advertisement placements highlight the versatility of the approach. The organic combination of the dynamic interest-aware network (DIN), cross-domain migration learning module, and bandit-propensity strategy enables seamless knowledge transfer, making MUMA a robust recommendation framework for diverse content scenarios.

6 Conclusion

The MUMA framework provides a robust solution for user behavior analysis and personalized recommendation in the media industry by integrating multi-source heterogeneous data, dynamic interest-aware networks (DINs), cross-domain migratory learning, and a bandit-propensity hybrid recommendation strategy. These innovations have led to a 29.7% increase in CTR, 18.3% improvement in viewing completion, and 82.5% satisfaction among cold-start users. The framework's contributions include enhancing user behavior analysis through data fusion, improving short-term and long-term preference modeling via DIN, refining cold-start recommendations with cross-domain learning, and ensuring robustness and interpretability through hybrid strategies and real-time feature engineering. Future research should focus on optimizing multimodal data fusion, improving cross-domain transfer generalization via meta-learning, integrating reinforcement learning with causal inference for transparent decision-making, advancing real-time system efficiency through incremental learning, and strengthening privacy safeguards with differential privacy and federated learning. These developments will further enhance personalization and scalability, supporting omnichannel intelligent recommendations in evolving digital ecosystems, including the metaverse and immersive media platforms.

Funding

This research was funded by the Key Project in Humanities and Social Sciences of Chizhou University (CZ2023ZSZ05).

References

- [1] Li Y, Li C, Wang F. Edge-enabled personalized fitness recommendations and training guidance for athletes with privacy preservation[J]. *Information Sciences*, 2025, 707122032–122032.
- [2] Li S, Gong J, Ke S, et al. Graph Transformer-based Heterogeneous Graph Neural Networks enhanced by multiple meta-path adjacency matrices decomposition[J]. *Neurocomputing*, 2025, 629129604–129604.
- [3] Wang T, Ge D. Research on Recommendation System of Online Chinese Learning Resources Based on Multiple Collaborative Filtering Algorithms (RSOCLR)[J]. *International Journal of Human-Computer Interaction*, 2025, 41(3):1771–1781.

- [4] Begum M, Suganthi B, Sivagamasundhari P, et al. An Enhanced Heterogeneous Local Directed Acyclic Graph Blockchain With Recalling Enhanced Recurrent Neural Networks for Routing in Secure MANET-IOT Environments in 6G[J]. *International Journal of Communication Systems*, 2025, 38(4):e6110–e6110.
- [5] Hassan H R, Hassan T M, Sameem I S M, et al. Personality-Aware Course Recommender System Using Deep Learning for Technical and Vocational Education and Training[J]. *Information*, 2024, 15(12): 803–803.
- [6] Yan L. Research on Personalized Cultural Learning Platform Based on Collaborative Filtering and Popularity Recommendation[J]. *International Journal of High Speed Electronics and Systems*, 2024, (prepublish).
- [7] Huang S, Yang H, Yao Y, et al. Deep Adaptive Interest Network: Personalized Recommendation with Context-Aware Learning[J]. *Journal of Electronics and Information Science*, 2024, 9(3).
- [8] Zhang Z. Personalized resource recommendation method of student online learning platform based on LSTM and collaborative filtering[J]. *Journal of Intelligent Systems*, 2024, 33(1).
- [9] Nahta R, Chauhan S G, Meena K Y, et al. Deep learning with the generative models for recommender systems: A survey[J]. *Computer Science Review*, 2024, 53100646–.
- [10] Guo X, Luo F, Zhao Z, et al. Federated personalized home BESS recommender system based on neural collaborative filtering[J]. *International Journal of Electrical Power and Energy Systems*, 2024, 159110042–.
- [11] Huang X, Wang J, Cui J. A Personalized Collaborative Filtering Recommendation System Based on Bi-Graph Embedding and Causal Reasoning[J]. *Entropy*, 2024, 26(5).
- [12] Yunfei Y, Jiameng W, Himo B A, et al. TIEN: Temporal interest-aware evolution model for “Next Item Recommendation”[J]. *Expert Systems With Applications*, 2024, 236.
- [13] Yang X, Wu J, Yu J. Interest-Aware Message-Passing Layer-Refined Graph Convolutional Network for Recommendation[J]. *Symmetry*, 2023, 15(5).
- [14] Zhiguo L, Weijie L, Jianxin F, et al. A regional interest-aware caching placement scheme for reducing latency in the LEO satellite networks[J]. *Peer-to-Peer Networking and Applications*, 2022, 15(6):2474–2487.
- [15] Li C, Liu Z, Wu M, et al. Multi-Interest Network with Dynamic Routing for Recommendation at Tmall.[J]. *CoRR*, 2019, abs/1904.08030.

Biographies



Jialing Wang received her Bachelor of Arts (B.A.) in 1999, Master of Arts (M.A.) in 2004, and Doctor of Philosophy (Ph.D.) in 2013, all from National Taiwan University. She is currently a Professor at the School of Literature and Media, Chizhou University. Her research focuses on the intersection of artificial intelligence, media analytics, and digital culture, investigating how emerging technologies shape user behavior modeling, brand communication strategies, and fan community dynamics within digital ecosystems.



Jun Zheng received her Bachelor of Arts (B.A.) from Anhui University of Engineering Science and Technology in 2010 and her Master of Arts (M.A.) from Anhui University of Engineering in 2014. She is currently an Associate Professor at the School of Literature and Media, Chizhou University. Her research explores the modernization of traditional symbols and the evolution of urban IP ecosystems, with a particular emphasis on cultural representation and digital branding strategies.

