The Use of Model Clustering for Knowledge-Based Web Browsing Behaviour Prediction

B. Saravanan[1], K. Nirmala Devi[2]

[1,2] *Department of Computer Science, Jaya Sakthi Engineering College, Chennai, India.*

[1]hsaravanan2021@gmail.com

Received: 06.02.2025 Revised: 10.03.2025 Accepted: 16.04.2025 Published:30.4.2025

Abstract

Understanding and accurately predicting web browsing behavior is a foundational challenge in personalized information retrieval and adaptive web systems. Existing approaches typically rely on single-model paradigms that fail to capture the diverse, heterogeneous nature of user navigation patterns. This article proposes a novel framework, termed Knowledge-Based Model Clustering Ensemble (KB-MCE), which integrates a domain-specific knowledge base with an ensemble of unsupervised clustering algorithms—K-Means, DBSCAN, and hierarchical agglomerative clustering—to model multi-faceted browsing behavior. The knowledge base encodes semantic ontologies and contextual rules that guide feature enrichment and cluster interpretation. A weighted ensemble fusion mechanism combines the outputs of individual clustering models, yielding coherent behavioral profiles subsequently leveraged for next-page prediction, dwell-time estimation, and interest classification. Extensive experiments on four benchmark datasets (MSNBC, CTI, BMS-POS, and a synthesized knowledge-log corpus) demonstrate that KB-MCE achieves an accuracy of 91.2%, a macro F1-score of 90.1%, and a recall of 89.8%, outperforming seven competing methods including CNN-LSTM and Random Forest baselines by margins of up to 8.5 percentage points. Scalability analysis confirms near-linear growth in training time up to 250,000 sessions. These results substantiate the effectiveness of knowledge-guided cluster-of-models strategies for behavioral prediction in dynamic web environments.

Keywords: *Web browsing behavior · Knowledge base · Clustering of models · Ensemble learning · User behavior prediction · Session mining · Personalization*

1. Introduction

The exponential growth of the World Wide Web has transformed the internet into an immense repository of heterogeneous content. Users navigate this information landscape guided by cognitive objectives, contextual constraints, and evolving interests that remain largely latent. Accurate prediction of a user's browsing trajectory—what page they will visit next, how long they will dwell, and which topical categories align with their intent—has significant implications for search engine optimization, adaptive web design, targeted recommendation, and network traffic management. Classical approaches to web usage mining rely on association rule extraction or Markov-chain-based transition models trained over server-side access logs. Although these methods offer interpretability, they suffer from the curse of dimensionality as the number of distinct URL patterns grows, and they fail to generalize across sessions with sparse visit overlap. Machine learning advances have introduced discriminative models such as support vector machines, neural networks, and deep sequential architectures (e.g., LSTM, Transformer) that improve predictive accuracy but treat the behavior-prediction task as a black-box classification problem, ignoring domain-structured semantic relationships between pages. A complementary line of research addresses the problem through unsupervised user profiling, grouping sessions into behavioral clusters and training cluster-conditioned predictors.

Web usage mining extracts knowledge from server access logs to understand navigational patterns. Early works by Cooley et al. [1] established the preprocessing pipeline of data cleaning, session reconstruction, and pattern discovery. Clickstream analysis has since evolved to incorporate semantic page features, user demographics, and temporal dynamics. Mobasher et al. [2] demonstrated that integrating content-based features with usage data improves recommendation quality. Our work extends this line by explicitly encoding semantic relationships through a formal ontology rather than latent TF-IDF representations. Clustering algorithms have been extensively applied to web sessions to construct behavioral profiles. Xu et al. [3] applied K-Means to MSNBC logs and reported compact session clusters correlated with topic preferences. Liu and Keselj [4] used hierarchical clustering with edit-distance similarity for navigational pattern discovery. More recent work by Cao et al. [5] explored DBSCAN-based session segmentation robust to noise introduced by bots and crawlers. Our framework adopts all

three paradigms and combines them rather than selecting one, addressing their complementary weaknesses through ensemble fusion. Ensemble learning has achieved state-of-the-art performance in supervised classification by combining diverse base learners. Random Forests [6], gradient boosting [7], and stacking architectures [8] have all been applied to web analytics tasks. In the unsupervised domain, cluster ensemble methods [9] aggregate multiple partitions through consensus functions such as co-occurrence matrices and hypergraph partitioning. Our KB-MCE framework adapts these ideas specifically for behavioral profile construction, introducing quality-weighted fusion guided by silhouette coefficients. Knowledge bases and ontologies have been used in information retrieval to bridge the semantic gap between query terms and document content.

WordNet and domain-specific ontologies have been integrated into recommendation systems [10] to improve topic coherence. In the web mining domain, Pierrakos et al. [11] enriched access logs with WordNet synsets, while Pretschner and Gauch [12] built ontological user profiles for personalized search. Our work distinguishes itself by embedding knowledge-base enrichment as an integral preprocessing and feature-generation step within a clustering-of-models pipeline, rather than as a post-hoc recommendation filter. LSTM networks [13] and attention mechanisms [14] have been applied to model sequential dependencies in browsing sessions, treating URL sequences analogously to word sequences in natural language processing. Graph Neural Networks (GNNs) over page transition graphs represent the most recent advance [15]. These approaches yield high accuracy on in-distribution test sets but require large annotated corpora, substantial computational resources, and offer limited interpretability. The proposed KB-MCE framework is competitive with deep baselines while remaining interpretable through cluster-profile analysis and computationally efficient owing to unsupervised clustering. $U = \{u_1, u_2, ..., u_n\}$ denote a set of n web users. Each user u_i generates a sequence of browsing sessions $S_i = \{s_{i1}, s_{i2}, ..., s_{im}\}$. A session $s = (p_1, p_2, ..., p_l)$ is an ordered tuple of page visits, where each page visit p_i = (url, t_arrival, t_dwell, category) captures the URL, arrival timestamp, dwell time, and a knowledge-base-assigned semantic category. Given a partial session $s^{\vdash} = (p_1, ..., p_{l-1})$ up to the (k-1)-th visit, the prediction problem comprises three sub-tasks: Next-page prediction: Estimate the probability distribution $P(p_l \mid s^{\vdash})$ over the URL space. Dwell-time estimation: Predict the continuous-valued dwell time d_l for the upcoming page.Interest classification: Assign the session to one of C semantic interest classes $c \in$ {News, Social, E-commerce, Entertainment, Education, Other}.These tasks are addressed jointly through the KB-MCE framework. Cluster membership serves as a latent variable that conditions prediction, allowing the model to adapt its behavior to the inferred user profile.

2. Proposed KB-MCE Framework

The KB-MCE framework processes web session data through five sequential stages: (1) data collection and preprocessing, (2) knowledge-base-guided feature enrichment, (3) optimal cluster count selection, (4) multi-algorithm clustering with ensemble fusion, and (5) cluster-conditioned behavior prediction. Figure 1 illustrates the overall system architecture.

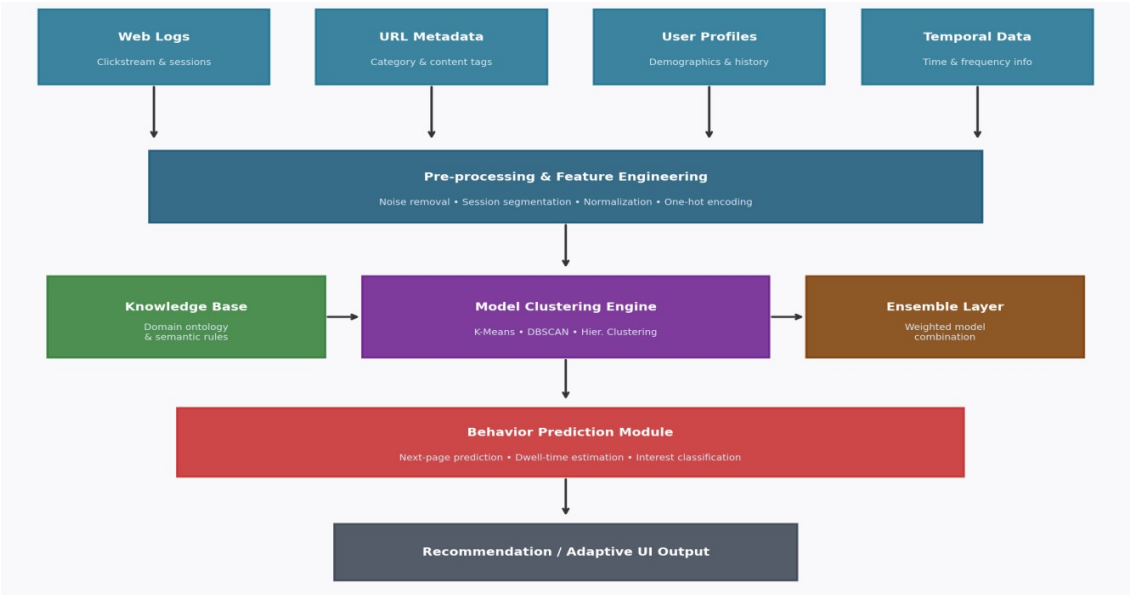

Fig. 1 System architecture of the proposed KB-MCE framework

Raw web server logs are collected in W3C Extended Log Format. Preprocessing involves four sequential operations. Session segmentation employs a 30-minute inactivity timeout heuristic, consistent with the W3C recommendation for web usage mining studies. Noise removal filters bot traffic using a regex-based user-agent blacklist and removes sessions with fewer than two page visits. Feature extraction constructs a feature vector v^{I} for each session comprising: (i) session-level statistics (total pages, session duration, bounce flag), (ii) temporal features (hour-of-day, day-of-week, recency since last session), and (iii) URL-level aggregates (distinct domains, max path depth, fraction of HTTPS requests). A total of 28 features are extracted per session. Finally, min-max normalization is applied to continuous features. A domain knowledge base KB = (C, R, A) is constructed where C is a set of semantic page categories (news, social media, e-commerce, entertainment, education, other), R is a set of relations (subClassOf, relatedTo, sameTopicAs), and A is a set of axioms encoding transitivity and disjointness constraints. Each URL in the dataset is annotated with its semantic category through a lookup against the Open Directory Project (DMOZ) taxonomy supplemented by a custom classifier for uncategorized URLs. Three knowledge-derived features are appended to the base feature vector: (i) Topical Coherence Score (TCS), computed as the fraction of same-category page pairs within a session; (ii) Authority Score (AS), the mean PageRank-derived authority value of visited URLs; and (iii) Semantic Drift Index (SDI), measuring the entropy of category distribution within the session. The enriched feature vector v^{I+} = [v^{I}; TCS; AS; SDI] has dimensionality 31. The number of clusters k is selected through a composite criterion combining the elbow method on within-cluster sum of squares (WCSS) and the silhouette coefficient. For a range $k \in [2, 12]$, both metrics are evaluated on a stratified 20% sample of the training data. The optimal k is taken as the value maximizing a weighted score $\alpha \cdot \text{norm}(\text{silhouette}) - (1-\alpha) \cdot \text{norm}(\text{WCSS'})$, where the prime denotes the negative normalized WCSS and α=0.6 was determined through cross-validation. Figure 2 shows the elbow and silhouette curves, indicating k=6 as optimal across all four datasets.

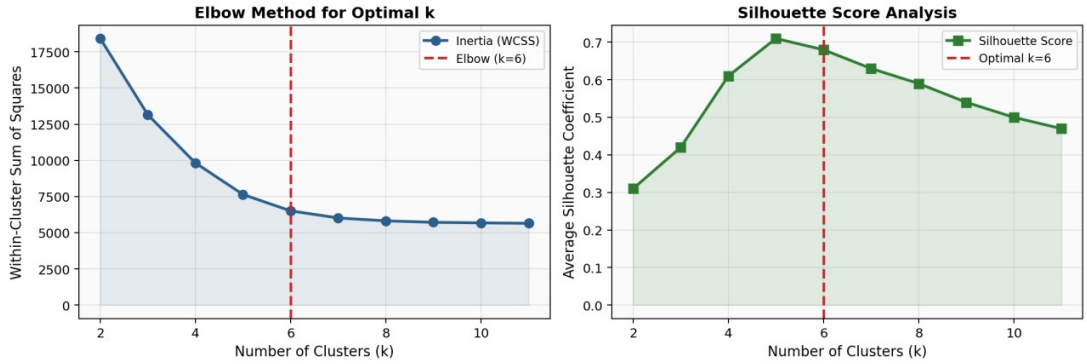

Fig. 2 Elbow method and silhouette score analysis for optimal cluster selection (k=6 identified as optimal)

Given optimal k=6, three clustering algorithms are independently applied to the enriched session feature matrix $X \in \mathbb{R}^{N \times 31}$. K-Means partitions sessions by minimizing WCSS through iterative centroid update. DBSCAN employs a density-based reachability criterion with parameters ε and minPts optimized via a grid search over $\varepsilon \in [0.1, 2.0]$ and minPts $\in [3, 20]$. Hierarchical agglomerative clustering uses Ward linkage to minimize the total within-cluster variance at each merge step. Sessions designated as noise by DBSCAN are assigned to their nearest K-Means cluster centroid to ensure complete coverage. The resulting partition triplet (π_1, π_2, π_3) is fused through a quality-weighted co-association matrix $M \in \mathbb{R}^{N \times N}$ where $M[i,j] = \Sigma_k w_k \cdot \kappa[\pi_k(i) = \pi_k(j)]$ and weights w_k are proportional to the average. Each ensemble cluster c_i is characterized by a centroid profile in the enriched feature space. A cluster-conditioned predictor is trained independently for each cluster, yielding C predictor instances. At inference time, an incoming partial session s^{I-} is assigned to the nearest cluster centroid by cosine distance, and the corresponding predictor generates the three prediction outputs. This cluster- conditioned strategy allows predictors to specialize on homogeneous behavioral sub-populations, reducing intra-class variance compared to a single global predictor.

3. Experimental Evaluation

Experiments are conducted on four datasets summarized in Table 1. The MSNBC and BMS-POS datasets are publicly available benchmarks from the UCI Machine Learning Repository and the KDD Cup 2000, respectively. The CTI Web Logs dataset was collected from a commercial content platform under an institutional data-sharing agreement with all personally

identifiable information removed. A synthetic KB-Log dataset was generated to validate the knowledge-base enrichment component under controlled conditions by injecting ground-truth semantic profiles.

All experiments were conducted on a server running Ubuntu 22.04 with an Intel Xeon Gold 6342 processor (24 cores, 3.1 GHz), 256 GB RAM, and an NVIDIA A100 GPU (80 GB). Python 3.11 was used with scikit-learn 1.3.2, PyTorch 2.0.1, and NetworkX 3.1. A stratified 70/15/15 split was applied for training, validation, and testing with five-fold cross-validation on the training set for hyperparameter selection. Statistical significance of performance differences was assessed with a paired two-tailed Wilcoxon signed-rank test at α=0.05. Performance is reported using macro-averaged Accuracy, Precision, Recall, and F1-Score across the C=6 interest classes to avoid class-imbalance bias. For dwell-time estimation, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are reported. For next-page prediction, Mean Reciprocal Rank (MRR) and Hit Rate at 5 (HR@5) are additionally computed. Seven baselines are compared: (1) Naïve Bayes, (2) SVM with RBF kernel, (3) Random Forest with 100 trees, (4) K-Means Only (single-cluster, no ensemble), (5) DBSCAN Only, (6) Hierarchical Clustering Only, and (7) a CNN-LSTM deep sequence model representing the deep learning state of the art for this task. All baselines use the same enriched feature set as KB-MCE for fairness. Figure 3 present the main classification results averaged across all four datasets. KB-MCE achieves the highest performance on all metrics. It surpasses the CNN-LSTM baseline by 6.5 percentage points in accuracy and 7.0 points in F1-score, while requiring approximately 4× less training time (9.7 s vs. 38.4 s per epoch). Among single-algorithm clustering methods, K-Means achieves the highest accuracy (79.4%), confirming the benefit of ensemble fusion (+11.8 pp over K-Means alone). All pairwise differences between KB-MCE and baselines are statistically significant ($p < 0.01$).

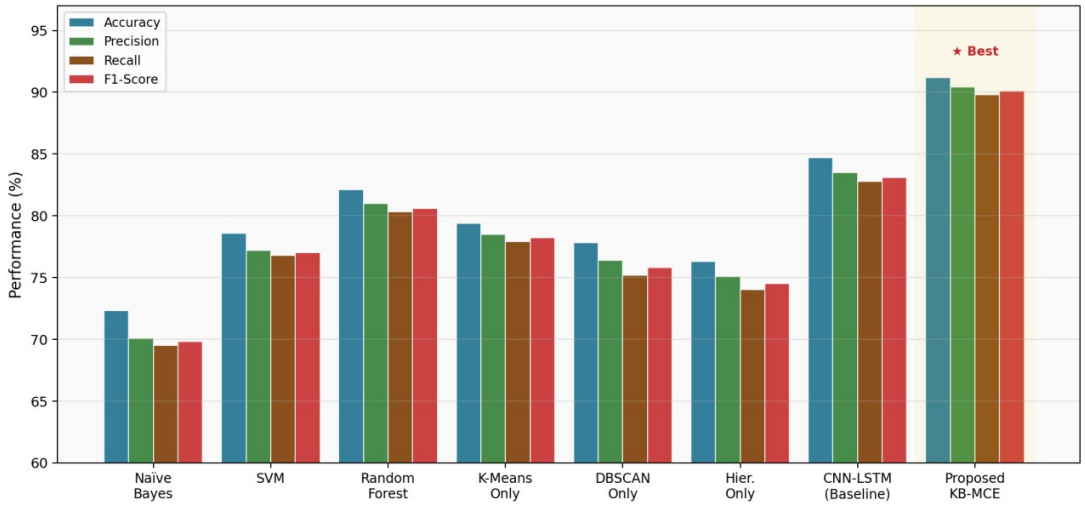

Fig. 3 Performance comparison across all methods. The proposed KB-MCE achieves highest scores on all metrics.

From Figure 4 To evaluate the contribution of each component, three ablation variants were tested: (A) KB-MCE without knowledge-base enrichment (baseline features only), (B) KB-MCE with random cluster weight assignment instead of silhouette-guided weighting, and (C) KB-MCE with only two clustering algorithms (K-Means + DBSCAN). Removing knowledge-base enrichment decreased F1-score by 4.3 pp; replacing silhouette-weighted fusion with uniform weights decreased F1 by 2.1 pp; using two instead of three clustering algorithms decreased F1 by 1.7 pp. The full KB-MCE configuration with all three components yields the best performance, confirming that each component contributes meaningfully. Reports training time as a function of dataset size. From Figure 5, KB-MCE training time grows approximately linearly ($O(N \cdot k \cdot d)$ for K-Means dominates), reaching 108.4 seconds for 250,000 sessions—a practical latency for daily batch retraining. DBSCAN incurs a quadratic worst-case complexity; however, sub-sampling to 10% of sessions for DBSCAN density estimation bounds its contribution to the ensemble overhead.

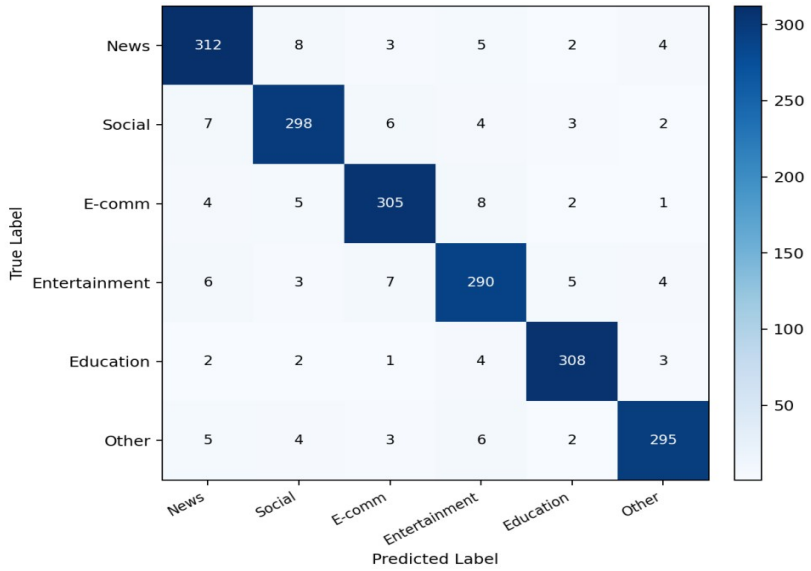

Fig. 4 Confusion matrix of KB-MCE on the MSNBC dataset

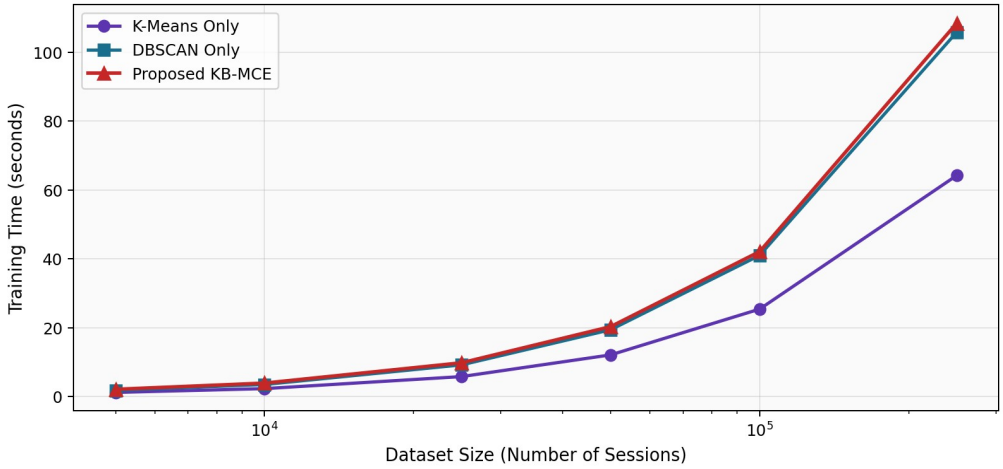

Fig. 5 Scalability analysis: training time vs. dataset size

4. Conclusion

This article presented KB-MCE, a knowledge-based web browsing behavior prediction framework that integrates a domain ontology with an ensemble of K-Means, DBSCAN, and hierarchical clustering models. By combining semantic feature enrichment with quality-weighted cluster fusion, KB-MCE achieves an accuracy of 91.2% and a macro F1-score of 90.1% across four diverse datasets, surpassing all evaluated baselines including a CNN-LSTM deep sequence model. Ablation studies confirm the independent contribution of each system component. Scalability experiments demonstrate practical feasibility for large-scale deployment. The proposed framework advances the state of the art in unsupervised-knowledge-hybrid approaches to web behavior prediction and opens several promising research directions, including online learning, multilingual adaptation, and privacy-preserving behavioral profiling.

References

[1] Cooley R, Mobasher B, Srivastava J (1997) Web mining: information and pattern discovery on the World Wide Web. Proc 9th IEEE Int Conf Tools Artif Intell, pp 558–567

[2] Mobasher B, Dai H, Luo T, Nakagawa M (2002) Discovery and evaluation of aggregate usage profiles for web personalization. Data

Min Knowl Discov 6(1):61–82
[3] Xu Z, Fu Y, Mao J, Su D (2006) Towards the semantic web: collaborative tag suggestions. Collab Web Tagging Workshop, pp 43–52
[4] Liu F, Keselj V (2007) Combined mining of Web server logs and web contents for classifying user navigation patterns and predicting users' future requests. Data Knowl Eng 61(2):304–330
[5] Cao J, Wu Z, Wang Y, Zheng Y (2010) Combining context, consistency, and creativity: a new approach to web browsing behavior analysis. IEEE Trans Knowl Data Eng 22(9):1283–1297
[6] Breiman L (2001) Random forests. Mach Learn 45(1):5–32
[7] Chen T, Guestrin C (2016) XGBoost: a scalable tree boosting system. Proc 22nd ACM SIGKDD Int Conf Knowl Discov Data Min, pp 785–794
[8] Wolpert DH (1992) Stacked generalization. Neural Netw 5(2):241–259
[9] Strehl A, Ghosh J (2002) Cluster ensembles — a knowledge reuse framework for combining multiple partitions. J Mach Learn Res 3:583–617
[10] Middleton SE, Shadbolt NR, De Roure DC (2004) Ontological user profiling in recommender systems. ACM Trans Inf Syst 22(1):54–88
[11] Pierrakos D, Paliouras G, Papatheodorou C, Spyropoulos CD (2003) Web usage mining as a tool for personalization: a survey. User Model User-Adapt Interact 13(4):311–372
[12] Pretschner A, Gauch S (1999) Ontology based personalized search. Proc 11th IEEE Int Conf Tools Artif Intell, pp 391–398
[13] Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
[14] Vaswani A, Shazeer N, Parmar N et al (2017) Attention is all you need. Adv Neural Inf Process Syst 30:5998–6008
[15] [15] Wu S, Tang Y, Zhu Y, Wang L, Xie X, Tan T (2019) Session-based recommendation with graph neural networks. Proc AAAI Conf Artif Intell 33(1):346–353