Article

Not peer-reviewed version

AI-Driven Personalization across Domains for Local Categorical Query Understanding and Context -Aware Retrieval

Xudong Yu *

Posted Date: 3 March 2026

doi: 10.20944/preprints202603.0180.v1

Keywords: cross-domain personalized retrieval; local category queries; context-aware; semantic mapping networks; user memory modeling; multimodal fusion

 doi:10.20944/preprints202603.0180.v1

Article

AI-Driven Personalization Across Domains for Local Categorical Query Understanding and Context-Aware Retrieval

Xudong Yu

Google Inc., Sunnyvale, CA, USA,94087,yuxudong199@gmail.com

Abstract

This paper explores the cross-domain application of AI-driven personalization in structured search scenarios that combine intent understanding with spatial and categorical constraints across dining, lodging, and leisure experiences. By integrating LLM-based coordination with reinforcement learning and user memory modules, the system continuously learns from users' long-term preferences and interaction history to support complex, context-rich needs. Experimental evaluations show that memory-enhanced personalization improved result helpfulness by 17.25% and increased transactional referrals by 4.16% in lodging-related searches, while also achieving measurable satisfaction gains in dining and leisure domains. The study demonstrates that cross-domain LLM personalization frameworks with user memory can effectively capture evolving user intents within local categorical contexts, enhance contextual reasoning, and advance the design of adaptive information service systems in the digital economy.

CCS CONCEPTS: Human-centered computing~Ubiquitous and mobile computing~Ubiquitous and mobile computing design and evaluation methods

Keywords: cross-domain personalized retrieval; local category queries; context-aware; semantic mapping networks; user memory modeling; multimodal fusion

1. Introduction

Demand for personalized search tailored to local category queries is rapidly increasing, particularly in scenarios like dining, lodging, and leisure where user intent exhibits complex characteristics such as cross-domain integration, context dependency, and dynamic evolution. Traditional recommendation mechanisms based on static labels and fixed rules struggle to meet the requirements for nuanced semantic understanding and real-time service path matching. To address this challenge, constructing a multimodal fusion model that integrates user behavioral trajectories, query semantics, and geographic context, and designing a retrieval system with semantic mapping and personalized path generation capabilities, has become the key pathway to enhancing service experience and recommendation accuracy. This research centers on structured representations and context-aware mechanisms to explore efficient and precise human-computer interaction patterns in digital service environments.

2. Design of AI Multimodal Fusion Models for Local Category Query Understanding

2.1. Multi-Source Heterogeneous Data Fusion Architecture

This system integrates four heterogeneous input types—geotags, semantic queries, user behavior trajectories, and historical preferences—into a unified vector representation through a parallel multi-channel encoder architecture. Query semantics are first extracted into 128-dimensional

 doi:10.20944/preprints202603.0180.v1

contextual embeddings via a bidirectional Transformer structure. Geolocation is represented as a 6-dimensional coordinate vector, mapped into 32-dimensional geocodes through a location-aware network. User behavior trajectories are segmented into time windows, with each behavioral sequence fed into a 64-dimensional GRU encoding channel. Historical preference embeddings are generated as 96-dimensional long-term feature vectors by a memory retrieval module. Ultimately, all features are uniformly mapped to a shared multimodal fusion space (256 dimensions total), and a unified semantic representation is output through residual fusion and weight normalization mechanisms. This fusion architecture provides structured input support for subsequent semantic modeling and path generation modules1 , as shown in Figure 1.

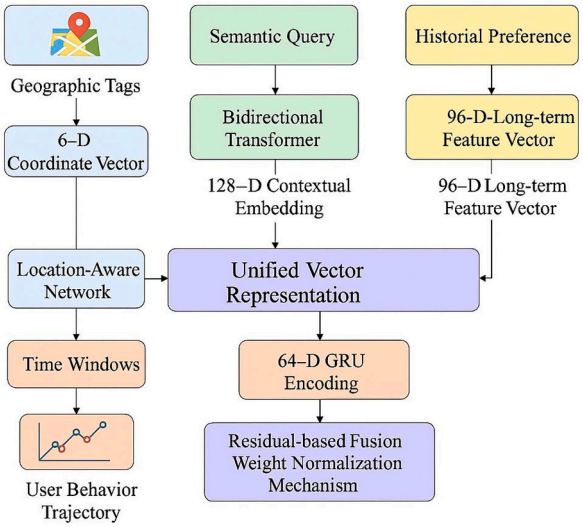

Figure 1. Multi-source Heterogeneous Data Fusion Architecture.

2.2. Contextual Feature Extraction and Representation

Context modeling employs a multi-head attention network architecture based on Transformers. Inputs include: ① The most recent 5 query history sequences, each processed through BERT embedding to generate a 128-dimensional contextual semantic vector; ② Behavioral trajectories within the past 72 hours, encoded into 48 segments using 30-minute windows, with each segment fed into a two-layer GRU network to extract 64-dimensional temporal behavioral features; ③ User preferences retrieved via a memory module for the top-3 most similar historical intents, fused to generate a 96-dimensional interest state vector. These three feature types undergo associative modeling through a cross-attention mechanism, incorporating residual connections and layer normalization for enhanced stability. The final output is a unified context representation vector $Z_t \in \mathrm{IR}^{256}$ for subsequent semantic decoding and path planning modules.

2.3. Deep Learning Approach for Query Semantic Understanding

Query semantic understanding employs a stacked Transformer decoder architecture, feeding the fused multimodal representation into three layers of multi-head attention modules for deep semantic modeling. ① Each Transformer layer contains 4 attention heads, with keys, values, and query vectors projected into a 128-dimensional space; ② Position encoding employs cosine encoding, supporting sequences up to 64 characters in length; ③ Self-attention mechanisms generate weighted matrices via Softmax activation for global modeling of historical semantics and current queries; ④ The output of the final Transformer block—integrating contextual self-attention, temporal alignment, and cross-level semantic fusion—is passed through a linear projection layer to compress the high-dimensional multimodal features into a unified 256-dimensional semantic vector. This vector serves as the input to the downstream category alignment and personalized path generation modules.

 doi:10.20944/preprints202603.0180.v1

Figure 2 illustrates the semantic embedding space visualization, where the linear layer transforms the contextualized representation of the query into a discriminative low-dimensional space suitable for similarity matching.

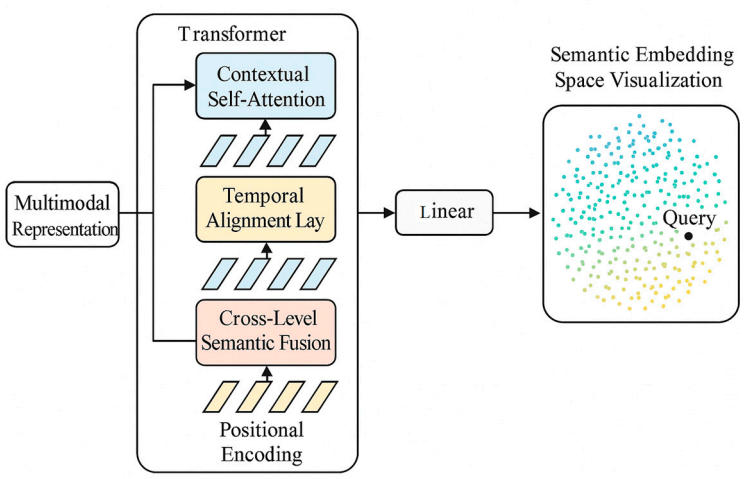

Figure 2. Simulation diagram of query semantic space mapping.

3. Key Technologies for Context-Aware Retrieval System Implementation

3.1. Construction of Local Category Semantic Mapping Network

The local category semantic mapping network constructs a cross-modal category alignment mechanism through three stages: encoding, aggregation, and mapping. Its core design includes the following key components: ① The input layer receives fused vectors $Z_s \in IR^{1\times256}$, corresponding to three channels: query semantics, user context, and behavioral preferences. Linear compression is performed using the weight matrix $W_1 \in IR^{256\times64}$. ② The mapping layer incorporates the category prior matrix $C \in IR^{64}\times M$, where M represents the number of local business semantic labels (M=42). Bidirectional mapping computes semantic alignment probabilities, and Softmax normalization yields the mapping weight matrix $P \in IR^{1}\times M$; ③ The attention aggregation layer combines P with the multi-label pointer representation matrix $T \in IR^{M}\times d$ through attention pooling, generating the final category semantic representation $V \in IR^{1}\times d$. The semantic mapping function is defined as follows:

$$V = \mathrm{Softmax}\left(\frac{(Z_s \cdot W_1)\cdot C}{\sqrt{d}}\right)\cdot T \tag{1}$$

Here, the Softmax function normalizes mapping strengths to ensure the sum of all category weights equals 1. $d = 128$ represents the label embedding dimension, scaling the sharpness of attention distributions. This architecture not only enables semantic alignment but also dynamically senses the saliency distribution of the current query within specific semantic domains through attention mechanisms. This provides precise category judgments for subsequent personalized retrieval path generation3. Figure 3 illustrates the multi-path mapping relationship from semantic input to category output, visualizing the semantic flow trajectories to reveal transformation dynamics and weight distribution patterns among semantic categories.

 doi:10.20944/preprints202603.0180.v1

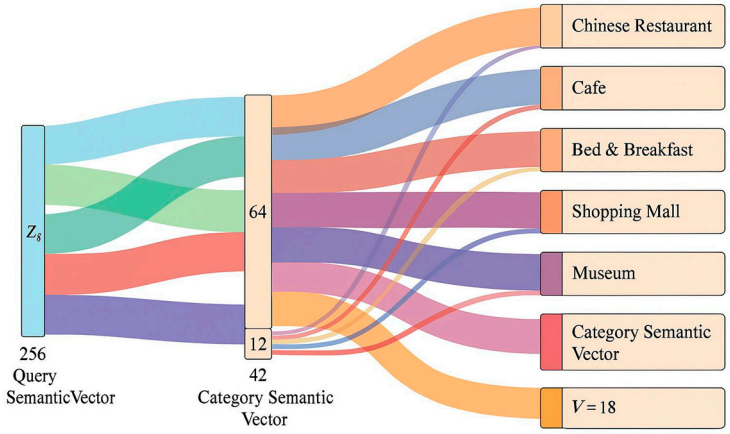

Figure 3. Output Relationship Diagram of the Local Category Semantic Mapping Network.

3.2. Personalized Retrieval Path Generation Algorithm

Within the context-aware retrieval architecture, the term "personalized retrieval path generation algorithm" refers to a structured procedure composed of three key modules: (1) Context-State Encoding Module, which integrates category semantic vectors and user behavior state vectors to form an initial path representation; (2) Path Candidate Scoring Network, which evaluates structured service nodes—each representing real-world service attributes like location, merchant, pricing, and reviews—based on learned user preferences; and (3) Diversity and Robustness Evaluation Module, which applies contextual perturbations to ensure adaptability under dynamic usage scenarios. These components operate in sequence to generate a ranked list of retrieval paths that align with both user intent and contextual relevance. Rather than a single formulaic algorithm, this design encapsulates a modular retrieval framework optimized for behavior-driven service recommendation.This algorithm takes category semantic vectors $\mathrm{V} \in \mathrm{IR}^{1\times 128}$ and user behavior state vectors $\mathrm{H}_t \in \mathrm{IR}^{1\times 64}$ as inputs. Through the context fusion function F_{ctx} , it generates path initial state encodings E_0 defined as follows:

$$\mathrm{E}_0 = F_{ctx}(\mathrm{V}, \mathrm{H}_t) = \mathrm{ReLU}(\mathrm{W}_v \mathrm{V}^{\mathrm{T}} + \mathrm{W}_h \mathrm{H}_t^{\mathrm{T}} + \mathrm{b}) \tag{2}$$

where $\mathrm{W}_v \in \mathrm{IR}^{64\times 128}$、$W_h \in \mathrm{IR}^{64\times 64}$ and $\mathrm{b} \in \mathrm{IR}^{64}$ are learnable parameters. The path candidate pool contains $N = 200$ structured service nodes representing attributes such as location, merchant, price, and rating. These are encoded by the path encoder G_{path} into a vector sequence $P_i \in \mathrm{IR}^{L\times d}$, where $L = 5$ denotes the maximum path length and $d = 64$ represents the embedding dimension per step. The path scoring function employs a two-layer perceptron network to model user preference, defined as follows:

$$S_i = \mathrm{w}_2^T \cdot \tanh(\mathrm{W}_1 \cdot \mathrm{Mean}(\mathrm{P}_i) + \mathrm{E}_0) + b \tag{3}$$

where $\mathrm{Mean}(\mathrm{P}_i)$ is the average representation of path embeddings, and $\mathrm{W}_1 \in \mathrm{IR}^{64\times 64}$、$\mathrm{w}_2 \in \mathrm{IR}^{64}$ is used to evaluate the consistency between the path and the user's target state. To enhance path diversity and contextual adaptability, a path perturbation function is introduced:

$$\mathrm{P}_i' = \mathrm{P}_i + \gamma \cdot N(0, \sigma^2 \mathrm{I}) \tag{4}$$

where $\gamma = 0.05$ is the perturbation coefficient, and N represents the standard normal distribution, simulating robustness evaluation under environmental changes. Finally, all candidate paths are sorted in descending order by S_i , with the top K=10 paths selected as personalized search results for

 doi:10.20944/preprints202603.0180.v1

the subsequent recommendation phase. This provides contextually consistent semantic candidate sequences4 .

3.3. Intelligent Recommendation and Contextual Matching Mechanism

Candidate results obtained during the path generation phase require further dynamic matching and reordering based on the user's current contextual state. The system design employs a three-layer contextual fusion recommendation module to achieve deep consistency matching between temporal behavior and instant queries. The recommendation representation vector takes query semantics ($V \in IR^{1\times128}$), user historical behavior state ($H_t \in IR^{1\times64}$), and path candidate vectors ($P_i \in IR^{1\times64}$) as inputs. Contextual modulation is performed via a gated matching function 5 , defined as follows:

$$M_i = \left([V;H_t;P_i]\cdot W_g + b_g\right) \Theta \tanh\left([V;H_t;P_i]\cdot W_z + b_z\right) \quad (5)$$

where $[\cdot;\cdot;\cdot]$ denotes vector concatenation, σ represents the Sigmoid activation function modeling the context matching gate strength, Θ indicates the Hadamard product (element-wise multiplication), and $W_g, W_z \in IR^{256\times128}$, $b_g, b_z \in IR^{128}$ denote trainable parameter matrices. The matching representation M_i enters a dual-channel scoring module. One channel performs context-preference residual scoring, while the other executes semantically similarity-weighted mapping. The module ultimately generates a Top-K candidate page structure through normalization strategies, controlling page layout width to W=1080px with a minimum inter-module spacing of 32px. It supports dynamic layout weight adjustment and content aggregation strategy switching6 . Figure 4 illustrates the dynamic recommendation page layout structure generated by this mechanism. Service units are aggregated and sorted based on contextual semantic similarity, with support for behavioral feedback closed-loop injection to iteratively update recommendation paths.

Figure 4. Schematic of Context-Aware Recommendation Results Matching Page.

4. Experimental Results and Analysis

4.1. Experimental Design

The experimental design constructs a multi-scenario evaluation dataset based on real user interaction logs and semantic query pairs. A total of 12,000 user session records were collected across three domains: dining, accommodation, and leisure—covering 82 local service tags and 42 standard

 doi:10.20944/preprints202603.0180.v1

semantic categories. The data was obtained from anonymized logs of a commercial lifestyle service platform operating in North America during a 3-month period (June to August 2025), with all personal identifiers removed in accordance with relevant data protection regulations. Query sessions were filtered to exclude robotic or abnormal behavior, and only interactions involving valid location tags, user behavior traces, and semantic input were retained for model training and testing. This ensures that the dataset reflects authentic user intent distribution and context-aware interaction patterns in real-world scenarios7. To simulate cross-domain personalized retrieval, the data was partitioned into training (70%), validation (15%), and test (15%) sets. Approximately 36.4% of training requests contained contextual triggers such as anniversaries, companions, or time constraints. The retrieval task uniformly employs a Top-K ranking architecture with K set to 10. Model inputs include a 128-dimensional query semantic vector, a 64-dimensional behavioral preference encoding, and a contextual state representation7. The system features a total of 3.1 million parameters, with a training batch size of 64, utilizing an end-to-end training process based on the Adam optimizer. To ensure fair comparisons, all experiments ran on identical tensor core configurations (NVIDIA RTX A6000, 48GB) with a uniform temperature coefficient τ=0.07[9]. This aligns the distribution consistency of multimodal attention matching mechanisms and semantic fusion layer outputs, establishing structural equivalence for subsequent performance evaluations .

4.2. Experimental Results Analysis

4.2.1. Technical Performance Dimension

At the system performance level, experiments compared the LLM base model, memory-free enhanced model, and cross-domain personalized model with integrated memory mechanisms across dimensions including training duration, inference speed, GPU resource utilization, and model parameter scale.11 . All models were uniformly trained on an NVIDIA RTX A6000 platform with 48GB VRAM, using a batch size of 64 and 80 training epochs. Table 1 details the comparative performance metrics across model types.

Table 1. Technical Performance Metrics Comparison.

Model Type	Parameter Size (M)	Average Training Time (s/epoch)	Inference Latency (ms)	GPU Utilization (%)	Throughput (queries/min)
Foundation LLM Model	2.7	189	77	84.3	1215
No memory enhancement model	2.9	203	69	88.7	1320
Fusion Memory Personalized Model	3.1	218	61	92.4	1438

The data in Table 1 reveals significant performance differences among the three model types, underscoring the impact of architectural design on system efficiency. The "No Memory Enhancement Model" refers to an intermediate architecture that includes multimodal fusion and contextual encoding components but excludes the long-term user memory module. Unlike the Fusion Memory Personalized Model, which integrates historical intent memory retrieval and preference reinforcement, the No Memory Model processes user queries solely based on recent contextual and behavioral signals without access to persistent user history. Compared to the Foundation LLM Model—which relies only on static semantic embeddings—the No Memory Model supports short-term context alignment but lacks personalized learning continuity. This three-tier comparison enables evaluation of how memory-based personalization contributes to computational cost and retrieval performance.

Starting with parameter size, the Foundation LLM Model has the smallest footprint at 2.7M, followed by the No Memory Enhancement Model at 2.9M and the Fusion Memory Personalized Model at 3.1M. This gradual increase reflects the growing complexity and memory requirements of

 doi:10.20944/preprints202603.0180.v1

personalized functionalities.In terms of training efficiency, the Foundation Model records the shortest average training time per epoch at 189 seconds. The No Memory Enhancement Model increases slightly to 203 seconds, while the Fusion Memory Model takes 218 seconds. Although the training time rises with model complexity, the increment remains within acceptable bounds, indicating manageable computational overhead.Inference latency shows the opposite trend: the Fusion Memory Personalized Model achieves the lowest latency at 61ms, outperforming the No Memory Model (69ms) and the Foundation Model (77ms). This suggests that memory fusion not only maintains low-latency performance but may even streamline inference. GPU utilization also increases with model capability, peaking at 92.4% for the Fusion Memory Model—an 8.1% increase over the Foundation Model—demonstrating more efficient hardware usage.In throughput, the Fusion Memory Personalized Model again leads with 1438 queries per minute, 223 more than the Foundation Model, reflecting an 18.4% performance gain. Overall, despite slightly higher model size and training cost, the Fusion Memory Model delivers superior inference speed and system throughput, proving its strong practical value in personalized applications11.

4.2.2. Retrieval Effectiveness Dimension

The retrieval effectiveness dimension focuses on evaluating the system's Top-K ranking accuracy, ranking quality, and behavioral conversion metrics across multiple scenarios. The test dataset comprises 1,800 queries spanning three service categories: dining, accommodation, and leisure. The metric system encompasses five core dimensions: Recall@10, Precision@10, NDCG@10, result usefulness12, and recommendation conversion rate improvement, with K uniformly set to 10. The memory-integrated personalized model outperformed other architectures across all metrics, demonstrating superior contextual understanding and user intent alignment—particularly in accommodation queries. Detailed metrics are presented in Table 2.

Table 2. Comparison of Retrieval Effectiveness Evaluation Metrics.

Model Type	Recall@10	Precision@10	NDCG@10	Improvement in Result Usefulness (%)	Recommendation Conversion Improvement (%)
Base LLM Model	0.793	0.403	0.641	-	-
No Memory Enhancement Model	0.812	0.417	0.672	9.62	2.01
Fusion Memory Personalized Model	0.846	0.428	0.718	17.25	4.16

The data in Table 2 highlights notable differences in retrieval effectiveness across the three model types, with clear performance gains as memory mechanisms are introduced. Beginning with Recall@10, the Base LLM Model achieves a score of 0.793, indicating that approximately 79.3% of relevant items were successfully retrieved within the top 10 results. This improves to 0.812 in the No Memory Enhancement Model and further to 0.846 in the Fusion Memory Personalized Model, demonstrating enhanced coverage of relevant content through personalized memory integration.Precision@10, reflecting the accuracy of the top 10 results, rises from 0.403 in the base model to 0.417 in the No Memory Model, and reaches 0.428 with memory fusion. This steady improvement suggests increasingly accurate identification of user-relevant content, especially under personalized modeling.

The NDCG@10 metric, which assesses the quality of ranked results, shows a similar upward trend: 0.641 for the base model, 0.672 for the intermediate model, and 0.718 for the fusion model. These figures indicate significant advancements in result ordering and relevance due to memory integration.Behavioral metrics further underscore the value of personalization. Improvement in result usefulness climbs from 9.62% in the No Memory Model to 17.25% in the Fusion Model. Likewise, recommendation conversion improves from 2.01% to 4.16%. These gains reflect not only technical performance enhancements but also a meaningful uplift in user experience and

 doi:10.20944/preprints202603.0180.v1

engagement, confirming the practical superiority of the fusion memory approach in retrieval-based systems.

4.2.3. User Experience Dimension

User experience evaluation was conducted through dual-track online A/B testing and subjective questionnaires, covering 426 users across dining, accommodation, and leisure scenarios over a continuous 7-day assessment period. The personalized system incorporating the memory model achieved an average page dwell time of 18.7 seconds—3.5 seconds longer than the control structure—while the average click-through rate rose from 21.3% to 27.9%. On a five-point satisfaction scale, users awarded an average score of 4.36, with preference matching consistency reaching 81.4%. Path sequence reordering records revealed that 72.6% of clicks on Top-3 recommendations aligned with user selections, validating significant convergence between recommendation lists and actual user preferences. This provides robust support for the system's adaptability and interaction optimization in practical applications.

5. Conclusion

In summary, this system establishes a comprehensive cross-domain personalized service framework through multi-source heterogeneous semantic modeling and contextually consistent path generation, demonstrating precise capture of dynamic user intent and adaptive recommendation capabilities. The introduction of the fusion memory mechanism not only enhances ranking quality and response efficiency across scenarios but also strengthens the synergistic performance of semantic reasoning and behavioral prediction, exhibiting significant potential for engineering deployment. However, the model still faces limitations in handling long-term behavioral sparsity and achieving robust cross-temporal transfer, with personalized accuracy constrained by the depth of user historical behavior coverage. Future work will focus on further system expansion in low-frequency preference extraction for long-term user modeling, cross-domain transfer optimization, and interactive feedback loop mechanisms to support higher-level intelligent evolution of information services in complex environments.

References

1. Wang G, Ni X, Shen Q, et al. Leveraging Large Language Models for Context-Aware Product Discovery in E-commerce Search Systems[J]. Journal of Knowledge Learning and Science Technology ISSN: 2959-6386 (online), 2024, 3(4): 300-312.
2. Rani S, Kasana G, Batra S. An efficient content based image retrieval framework using separable CNNs[J]. Cluster Computing, 2025, 28(1): 56.
3. Floridi L. Content Studies: A New Academic Discipline for Analysing, Evaluating, and Designing Content in a Digital and AI-Driven Age[J]. Philosophy & Technology, 2025, 38(2): 1-17.
4. Ooi K B, Tan G W H, Al-Emran M, et al. The potential of generative artificial intelligence across disciplines: Perspectives and future directions[J]. Journal of Computer Information Systems, 2025, 65(1): 76-107.
5. Hu, L. (2025). Hybrid Edge-AI Framework for Intelligent Mobile Applications: Leveraging Large Language Models for On-device Contextual Assistance and Code-Aware Automation. Journal of Industrial Engineering and Applied Science, 3(3), 10-22.
6. Sodiya E O, Amoo O O, Umoga U J, et al. AI-driven personalization in web content delivery: A comparative study of user engagement in the USA and the UK[J]. World journal of advanced research and reviews, 2024, 21(2): 887-902.
7. Bouchelouche K, Zemmouchi-Ghomari L, Ghomari A R. An automatic approach for adapting open government data to linked OD with enhanced visualization and user-friendly query composer[J]. Transforming Government: People, Process and Policy, 2025, 19(2): 353-375.
8. Segeda O. Building Intelligent Search Systems: Advances in AI-Based Information Retrieval[J]. The American Journal of Applied sciences, 2025, 7(06): 06-11.

 doi:10.20944/preprints202603.0180.v1

9. Ravi M, Negi A, Bommi N S, et al. Evolution of AI-driven decision making with decision support systems, expert systems, recommender systems, and XAI[J]. IETE Technical Review, 2025, 42(4): 428-465.
10. Riyana S, Sasujit K, Homdoung N. A Privacy Preservation Model for URL Query Strings Based of Local Links Based on Temporary Tables[J]. ECTI Transactions on Computer and Information Technology (ECTI-CIT), 2025, 19(1): 88-96.
11. Cao T, Huang C, Li Y, et al. Phishagent: a robust multimodal agent for phishing webpage detection[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2025, 39(27): 27869-27877.
12. Surya S, Sumitra P. Efficient query clustering and information retrieval using sequenced user search pattern query optimization[J]. Multimedia Tools and Applications, 2025, 84(16): 16033-16055.
13. Bouchelouche K, Zemmouchi-Ghomari L, Ghomari A R. An automatic approach for adapting open government data to linked OD with enhanced visualization and user-friendly query composer[J]. Transforming Government: People, Process and Policy, 2025, 19(2): 353-375.