Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

QOS-BASED WEB SERVICE RECOMMENDATION: A SYSTEMATIC LITERATURE REVIEW

Gajendra Singh*[1], Shyamol Banerjee[2]

[1] M.Tech Student, Dept. of CSE, SRCEM
[2] Assistant Professor, Dept. of CSE, SRCEM
*Corresponding Author: gsshekhar15@Gmail.com

Abstract

Web service recommendation has become an essential research area due to the rapid growth of cloud computing and service-oriented architectures. Functional similarity alone is insufficient for selecting appropriate web services because multiple services often provide identical functionality with varying Quality of Service (QoS) characteristics. QoS-aware recommendation systems assist users in selecting services based on attributes such as response time, availability, reliability, throughput, and reputation. This review presents a comprehensive analysis of existing QoS-based web service recommendation techniques, including collaborative filtering, matrix factorization, clustering, deep learning, graph-based models, and hybrid recommendation approaches. The paper compares existing methodologies, identifies their strengths and limitations, discusses publicly available QoS datasets and evaluation metrics, and highlights current research challenges and future directions. The review serves as a valuable reference for researchers developing next-generation intelligent web service recommendation systems.

Keywords: Web Service Recommendation, Quality of Service, QoS Prediction, Collaborative Filtering, Matrix Factorization, Deep Learning, Service Computing.

1. Introduction

The rapid evolution of the Internet, cloud computing, distributed systems, and Service-Oriented Architecture (SOA) has significantly increased the availability of web services across various application domains. Web services enable interoperable communication between heterogeneous software applications through standardized protocols such as HTTP, SOAP, REST, XML, and JSON. They have become fundamental components of modern information systems, supporting diverse applications including e-commerce, healthcare, finance, education, smart cities, the Internet of Things (IoT), and enterprise computing. As the number of publicly available web services continues to grow, users are often presented with multiple services offering identical or highly similar functionality. Consequently, identifying the most suitable service has become a challenging task.

Traditional web service selection methods primarily rely on functional requirements, where services are selected based on the operations they provide. However, functional properties alone are no longer sufficient because several services may implement the same functionality while exhibiting different levels of performance and reliability. In practical scenarios, users expect services not only to satisfy their functional requirements but also to deliver superior performance, reliability, availability, and user experience.

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

Therefore, non-functional characteristics, commonly referred to as Quality of Service (QoS), have become critical factors in web service discovery, selection, and recommendation.

Quality of Service represents a collection of measurable non-functional attributes that describe the performance and reliability of a web service. Common QoS parameters include response time, availability, throughput, reliability, latency, success rate, cost, scalability, security, and reputation. These attributes vary depending on network conditions, server workload, geographical location, and user environment. Since QoS values are dynamic and differ across users and locations, selecting an optimal web service based solely on advertised QoS values is often ineffective. Consequently, QoS-aware recommendation systems have emerged as an important research area to assist users in selecting services that best satisfy their quality expectations.

QoS-based web service recommendation aims to recommend the most appropriate services by analyzing historical QoS observations, user preferences, service characteristics, and contextual information. Recommendation systems predict the expected QoS values of candidate services and rank them according to users' quality requirements. These systems improve user satisfaction while reducing the effort required to evaluate numerous available services. Over the past decade, researchers have proposed various recommendation techniques, including collaborative filtering, content-based filtering, matrix factorization, clustering algorithms, probabilistic models, graph-based learning, deep learning, reinforcement learning, and hybrid recommendation frameworks. Each technique addresses specific challenges while exhibiting unique strengths and limitations.

Collaborative Filtering (CF) has been one of the most widely adopted approaches because it exploits historical QoS experiences of similar users to predict unknown QoS values. Nevertheless, CF suffers from challenges such as data sparsity and the cold-start problem, which reduce recommendation accuracy when historical data are limited. To overcome these limitations, researchers introduced matrix factorization methods, including Singular Value Decomposition (SVD), Probabilistic Matrix Factorization (PMF), and Non-negative Matrix Factorization (NMF), which effectively capture latent relationships between users and services. More recently, deep learning techniques, Graph Neural Networks (GNNs), attention mechanisms, and transformer-based architectures have demonstrated remarkable improvements in QoS prediction by learning complex nonlinear relationships from large-scale datasets.

The emergence of cloud computing, edge computing, mobile computing, and the Internet of Things has further increased the complexity of web service ecosystems. Modern applications often involve dynamic service composition, context-aware service selection, and real-time QoS monitoring. Furthermore, issues such as service reliability, trust management, privacy preservation, scalability, energy efficiency, and explainability have become increasingly important in designing intelligent recommendation systems. Emerging technologies, including blockchain, federated learning, explainable artificial intelligence (XAI), and large language models (LLMs), provide promising opportunities to address these challenges and enhance future QoS-aware recommendation frameworks.

Despite extensive research in this field, several limitations remain unresolved. Existing recommendation methods often struggle with highly sparse QoS datasets, rapidly changing service environments, heterogeneous user preferences, cross-cloud interoperability, and real-time service adaptation. Moreover, many existing surveys focus on specific algorithms or limited aspects of QoS prediction and do not provide a comprehensive comparison of traditional machine learning methods, deep learning approaches, graph-based techniques, and emerging intelligent recommendation models within a unified framework. As the field continues to evolve, there is a growing need for an updated review that systematically analyzes recent developments, identifies research gaps, and highlights future research opportunities.

This review paper addresses this need by presenting a comprehensive analysis of QoS-based web service recommendation techniques. It examines classical recommendation methods alongside recent advances in

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

machine learning, deep learning, graph-based learning, and hybrid recommendation models. The paper also reviews widely used QoS datasets, evaluation metrics, application domains, and benchmarking methodologies. Furthermore, it identifies the major research challenges affecting current recommendation systems and discusses emerging research directions that are likely to shape the next generation of intelligent web service recommendation frameworks.

The major contributions of this review are as follows:

- A comprehensive overview of QoS concepts and their significance in web service recommendation.
- A systematic classification of existing recommendation techniques, including traditional, machine learning-based, deep learning-based, graph-based, and hybrid approaches.
- A comparative analysis of existing methods based on prediction accuracy, scalability, computational complexity, sparsity handling, and recommendation quality.
- A review of publicly available QoS datasets and commonly used evaluation metrics.
- Identification of current research challenges, limitations, and open issues in QoS-aware recommendation.

Discussion of emerging research directions, including federated learning, Graph Neural Networks, explainable AI, blockchain-enabled trust management, reinforcement learning, and large language models for intelligent web service recommendation.

2. Quality of Service (QoS) Parameters

2.1 Overview

The increasing adoption of Service-Oriented Architecture (SOA), cloud computing, and distributed applications has led to the rapid growth of web services that offer similar functionalities. Since multiple services often provide identical functional capabilities, selecting the most suitable service based solely on functionality is insufficient. Consequently, Quality of Service (QoS) has become a key criterion for evaluating, comparing, and recommending web services.

Quality of Service refers to a set of non-functional attributes that describe the performance, reliability, efficiency, and overall quality of a web service. Unlike functional properties, which define *what* a service does, QoS attributes describe *how well* the service performs. These parameters enable users and service consumers to differentiate among functionally equivalent services and select those that best satisfy their performance requirements.

QoS values are dynamic and influenced by several factors, including network latency, server workload, geographical location, hardware configuration, virtualization, cloud infrastructure, and user access patterns. Therefore, accurate measurement, prediction, and recommendation of QoS have become fundamental research challenges in service computing.

Researchers generally classify QoS attributes into two broad categories:

- **Performance-related QoS parameters**, which measure execution efficiency and responsiveness.
- **Business and reliability-related QoS parameters**, which evaluate trustworthiness, cost, availability, and user satisfaction.

The most commonly used QoS parameters in web service recommendation are discussed below.

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

2.2 Response Time

Response time is one of the most important QoS attributes used in service recommendation. It represents the total time elapsed between sending a service request and receiving the corresponding response.

Mathematically,

Response Time = Response Arrival Time − Request Sending Time

A lower response time indicates better service performance and improved user experience.

Factors affecting response time include:

- Network latency
- Server processing speed
- System workload
- Data transmission delay
- Geographic distance
- Virtual machine allocation

Response time is widely used in collaborative filtering, QoS prediction, and ranking algorithms because it directly influences customer satisfaction.

2.3 Availability

Availability measures the probability that a web service remains operational and accessible whenever users request it.

It is commonly calculated as:

Availability = (Total Uptime / Total Observation Time) × 100%

A service with high availability experiences fewer failures and provides uninterrupted access.

High availability is particularly important for:

- Online banking
- Healthcare systems
- E-commerce platforms
- Government services
- Cloud applications

Service providers typically aim for availability levels above 99%.

2.4 Reliability

Reliability indicates the ability of a web service to perform its intended functions consistently without failure over a specified period.

Reliability is influenced by:

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

- Software stability
- Hardware reliability
- Fault tolerance
- Recovery mechanisms
- Error handling

2.5 Throughput

Throughput represents the number of successful service requests processed within a given time interval its depends on some key points as.

- CPU performance
- Memory availability
- Network bandwidth
- Database performance
- Load balancing strategies

High-throughput services are essential for applications with large numbers of concurrent users.

2.6 Latency

Latency is the communication delay experienced during data transmission between the client and the service provider.

Latency includes:

- Network propagation delay
- Transmission delay
- Processing delay
- Queueing delay

2.7 Success Rate

Success rate measures the percentage of service requests completed successfully without errors.

$$\textbf{Success Rate} = \textbf{(Successful Requests / Total Requests)} \times \textbf{100\%}$$

A higher success rate indicates greater service stability and reliability.

2.8 Reputation

Reputation reflects the overall trustworthiness of a web service based on historical user experiences and feedback.

Reputation may be derived from:

- User ratings
- Customer reviews
- Historical QoS records
- Recommendation scores
- Service rankings

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

Highly reputed services are generally preferred because they have demonstrated consistent performance over time.

2.9 Cost

Cost represents the monetary expense associated with using a web service. Common pricing models include here below:

- Pay-per-use
- Subscription-based
- Freemium
- Transaction-based
- Usage-based cloud pricing

3. QoS-Based Recommendation Framework

3.1 Overview

The rapid growth of cloud computing and Service-Oriented Architecture (SOA) has resulted in the availability of thousands of web services offering similar functionalities. While functional requirements determine whether a service can perform a specific task, they do not guarantee optimal performance or user satisfaction. Consequently, Quality of Service (QoS)-based recommendation systems have emerged as an effective solution for identifying the most suitable web services based on users' non-functional requirements.

A QoS-based recommendation framework predicts the expected quality of candidate services and recommends those that best satisfy users' preferences. Unlike traditional service discovery methods, which rely primarily on functional descriptions, QoS-aware recommendation systems utilize historical QoS observations, user behavior, service characteristics, and contextual information to generate personalized recommendations.

A typical recommendation framework consists of multiple interconnected components, including data collection, QoS preprocessing, prediction, recommendation generation, and performance evaluation. Figure 1 illustrates the general architecture of a QoS-based web service recommendation system.

3.2 General Framework Architecture

The overall workflow of a QoS-based recommendation system can be represented as follows:

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

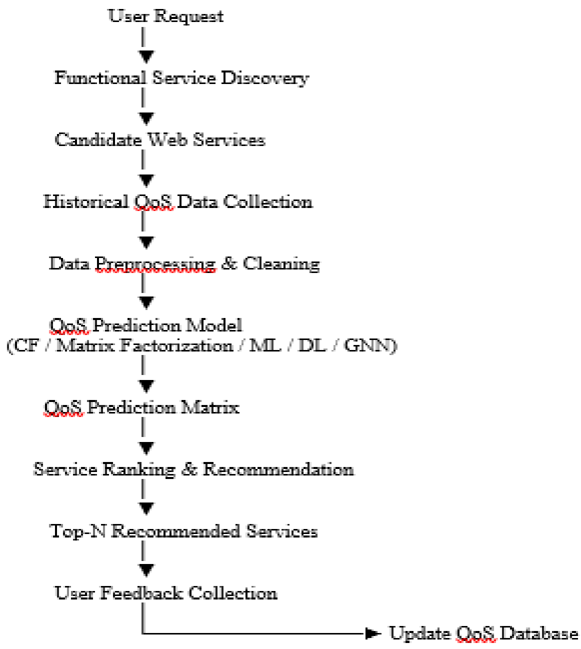

Figure 1 continuous learning cycle

The framework operates as a continuous learning cycle in which user feedback and newly observed QoS values are incorporated to improve future recommendations.

3.3 Framework Components

3.3.1 User Request

The recommendation process begins when a user submits a request for a particular web service. The request generally contains:

- Functional requirements
- QoS preferences
- Budget constraints
- Geographic location
- Application context

For example, a user may request a payment gateway service with a response time below 500 milliseconds and availability above 99.9%.

3.3.2 Functional Service Discovery

The system first identifies services that satisfy the required functionality using service descriptions, APIs, semantic information, or service registries.

Common discovery mechanisms include:

- WSDL matching
- REST API descriptions
- Semantic Web Services

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

- UDDI registries
- Ontology-based matching

3.3.3 QoS Data Collection

After identifying candidate services, the framework collects QoS observations from multiple sources.

Typical data sources include:

- Historical service invocation records
- User monitoring logs
- Cloud monitoring systems
- Benchmark datasets
- User feedback
- Service provider information

4. Comparison of Existing Methods

4.1 Overview

Over the past two decades, numerous QoS-based web service recommendation techniques have been proposed to improve service selection and QoS prediction accuracy. These approaches differ in terms of learning strategy, computational complexity, scalability, prediction accuracy, and their ability to handle sparse QoS data. Traditional techniques, such as Collaborative Filtering (CF), primarily exploit historical user-service interactions, whereas more recent approaches employ machine learning, deep learning, graph learning, and hybrid models to capture complex relationships among users, services, and QoS attributes.

Each recommendation technique possesses distinct strengths and weaknesses. Therefore, selecting an appropriate recommendation model depends on factors such as dataset size, sparsity level, computational resources, application domain, and real-time requirements. Table 5 summarizes the characteristics of major QoS-based recommendation techniques.

Table 1. Comparison of Existing QoS-Based Web Service Recommendation Methods

Technique	Principle	Advantages	Limitations	Suitable Applications
Collaborative Filtering (CF)	Predicts QoS using similar users or services	Simple, personalized, widely adopted	Cold-start problem, data sparsity, reduced accuracy with limited data	Small and medium-scale recommendation systems
Content-Based Filtering	Recommends services based on service features and user preferences	Effective for new users, independent of other users	Limited personalization, overspecialization	Personalized service recommendation
Matrix Factorization (MF)	Learns latent relationships between users and services	High prediction accuracy, handles sparse matrices	Computationally intensive, parameter tuning required	Large-scale cloud environments
Clustering-Based	Groups users or services with similar	Reduces computational	Sensitive to cluster initialization, may lose	Large datasets with

Methods	QoS characteristics	cost, improves scalability	local information	similar user groups
Machine Learning Models	Learns QoS patterns using supervised algorithms	Handles nonlinear relationships, adaptable	Requires feature engineering and training data	QoS prediction and classification
Deep Learning Models	Uses neural networks to learn complex QoS patterns	High accuracy, automatic feature extraction	High computational cost, large datasets required	Intelligent cloud service recommendation
Graph-Based Models	Models relationships using graph structures	Captures complex dependencies, effective for sparse data	Complex implementation, increased memory requirements	Large-scale service ecosystems
Hybrid Models	Combines multiple recommendation techniques	High prediction accuracy, alleviates cold-start and sparsity	Increased computational complexity	Modern cloud-based recommendation systems

4.2 Comparative Analysis

Collaborative Filtering remains one of the most widely adopted techniques due to its simplicity and effectiveness in exploiting historical QoS observations. However, its performance deteriorates significantly under sparse data conditions and for newly introduced users or services. Matrix Factorization improves upon traditional collaborative filtering by identifying latent features that explain user-service interactions, thereby achieving higher prediction accuracy.

Clustering-based methods reduce computational overhead by partitioning similar users or services into groups before prediction. Although clustering improves scalability, the quality of recommendations depends heavily on the clustering algorithm and parameter selection.

Machine learning approaches such as Random Forest, Support Vector Machine, and Gradient Boosting capture nonlinear QoS relationships more effectively than conventional statistical methods. Deep learning models, including Autoencoders, CNNs, LSTMs, and Transformer architectures, further enhance prediction accuracy by automatically learning hierarchical feature representations from large-scale QoS datasets.

Graph-based recommendation techniques have recently attracted considerable attention because they explicitly model relationships among users, services, and QoS attributes. Graph Neural Networks (GNNs) have demonstrated promising results in handling sparse datasets and learning complex service dependencies.

Hybrid recommendation systems integrate two or more techniques, such as Collaborative Filtering with Deep Learning or Matrix Factorization with Graph Neural Networks, to overcome the limitations of individual approaches. These systems generally achieve the highest recommendation accuracy but require greater computational resources and sophisticated optimization strategies.

Overall, recent studies indicate a clear transition from traditional collaborative filtering methods toward intelligent hybrid frameworks that combine machine learning, deep learning, graph learning, and context-aware computing. Such approaches offer improved prediction accuracy, better scalability, and greater robustness in dynamic cloud environments.

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

5. Research Challenges

Although significant progress has been made in QoS-based web service recommendation, several open research challenges continue to limit the effectiveness and practical deployment of existing systems.

5.1 Data Sparsity

QoS datasets are inherently sparse because users invoke only a small fraction of the available services. As a result, many entries in the user-service QoS matrix remain unknown, reducing the prediction accuracy of collaborative filtering and matrix factorization techniques.

5.2 Cold-Start Problem

New users and newly published web services often lack historical QoS observations. Without sufficient interaction data, recommendation algorithms cannot accurately estimate QoS values, leading to poor recommendation quality during the initial stages.

5.3 Dynamic QoS Characteristics

QoS values are not static. They continuously change due to network congestion, server workload, hardware failures, geographical location, virtualization, and cloud resource allocation. Static recommendation models cannot effectively adapt to these real-time variations.

5.4 Scalability

Modern cloud platforms host thousands of web services and serve millions of users. Processing such large-scale QoS data requires highly scalable recommendation algorithms with low computational complexity and efficient memory utilization.

5.5 Explainability

Many deep learning and graph-based recommendation models function as black boxes, making it difficult to explain why a particular service has been recommended. Explainable Artificial Intelligence (XAI) is therefore emerging as an important research direction for improving transparency and user trust.

5.6 Emerging Technologies

The rapid adoption of cloud-native applications, microservices, edge computing, the Internet of Things (IoT), blockchain, and large language models introduces new requirements for QoS prediction and recommendation. Future recommendation systems must efficiently support these evolving computing paradigms.

5.7 Future Research Directions

Based on the identified challenges, future research should focus on:

- Federated learning for privacy-preserving QoS prediction.
- Graph Neural Networks for modeling complex user-service relationships.
- Transformer-based deep learning architectures for temporal QoS prediction.
- Reinforcement learning for adaptive service recommendation.
- Explainable AI (XAI) to improve transparency and trust.

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

- Blockchain-based trust management for secure QoS sharing.
- Context-aware and personalized recommendation frameworks.
- Energy-efficient and sustainable service recommendation.
- Multi-objective optimization techniques for balancing multiple QoS parameters.
- Integration of Large Language Models (LLMs) to support intelligent service discovery, semantic understanding, and automated recommendation.

6. Conclusion

Quality of Service (QoS)-based web service recommendation has become a critical research area in service computing, cloud computing, and Service-Oriented Architecture due to the increasing availability of functionally similar web services. Since functional properties alone cannot effectively distinguish among competing services, QoS attributes such as response time, availability, reliability, throughput, latency, reputation, security, and cost have emerged as essential criteria for service selection and recommendation. Consequently, numerous recommendation techniques have been proposed to predict unknown QoS values and assist users in selecting services that best satisfy their quality requirements.

This review has presented a comprehensive overview of QoS-based web service recommendation, including fundamental QoS parameters, recommendation frameworks, widely used prediction techniques, publicly available datasets, and evaluation metrics. Traditional methods such as Collaborative Filtering, Matrix Factorization, and clustering have laid the foundation for QoS prediction, while recent advances in machine learning, deep learning, Graph Neural Networks, and hybrid recommendation models have significantly improved prediction accuracy, scalability, and robustness. The comparative analysis demonstrates that no single recommendation technique is universally optimal. Instead, hybrid and intelligent learning frameworks generally provide superior performance by effectively addressing issues such as data sparsity, cold-start scenarios, and dynamic QoS variations.

Despite substantial progress, several important challenges remain unresolved, including sparse QoS observations, rapidly changing service environments, context awareness, privacy preservation, explainability, and large-scale deployment. Addressing these issues requires the development of adaptive, scalable, and trustworthy recommendation frameworks capable of operating efficiently in modern cloud and edge computing environments.

Future research is expected to leverage emerging technologies such as federated learning, Graph Neural Networks, reinforcement learning, explainable artificial intelligence, blockchain-based trust management, and Large Language Models to build next-generation QoS-aware recommendation systems. These intelligent systems will be capable of learning continuously from dynamic service ecosystems, providing personalized recommendations, protecting user privacy, and supporting real-time decision-making. Such advancements are expected to play a vital role in improving the reliability, efficiency, and user satisfaction of web service recommendation systems across diverse application domains.

References

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

[1] M. Alrifai and T. Risse, "Combining global optimization with local selection for efficient QoS-aware service composition," *Proceedings of the 18th International Conference on World Wide Web (WWW)*, pp. 881–890, 2009.

[2] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "QoS-aware web service recommendation by collaborative filtering," *IEEE Transactions on Services Computing*, vol. 4, no. 2, pp. 140–152, Apr.–Jun. 2011.

[3] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "WS-DREAM: A distributed reliability assessment mechanism for web services," *Proceedings of the IEEE International Conference on Web Services (ICWS)*, 2010.

[4] Z. Zheng, Y. Zhang, and M. R. Lyu, "Distributed QoS evaluation for real-world web services," *Proceedings of the IEEE International Conference on Web Services (ICWS)*, 2010.

[5] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "Collaborative web service QoS prediction via neighborhood integrated matrix factorization," *IEEE Transactions on Services Computing*, vol. 6, no. 3, pp. 289–299, 2013.

[6] M. Papazoglou, "Service-oriented computing: Concepts, characteristics and directions," *Proceedings of the IEEE International Conference on Web Information Systems Engineering*, 2003.

[7] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall, 2005.

[8] K. Zhang, Y. Yang, S. Nepal, D. Liu, and J. Chen, "A survey on service recommendation techniques," *Journal of Network and Computer Applications*, vol. 86, pp. 27–45, 2017.

[9] X. Chen, Z. Zheng, X. Liu, Z. Huang, and H. Sun, "Personalized QoS-aware web service recommendation and visualization," *IEEE Transactions on Services Computing*, vol. 6, no. 1, pp. 35–47, 2013.

[10] S. Wang, Z. Zheng, Z. Wu, M. R. Lyu, and F. Yang, "Reputation measurement and malicious feedback rating prevention in web service recommendation systems," *IEEE Transactions on Services Computing*, vol. 8, no. 5, pp. 755–767, 2015.

[11] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-Based Systems*, vol. 46, pp. 109–132, 2013.

[12] P. Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56–58, 1997.

[13] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 2009, Article ID 421425, 2009.

[14] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[15] C. Aggarwal, *Recommender Systems: The Textbook*. Springer, 2016.

[16] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*, 3rd ed. Springer, 2022.

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

Received: 2026-02-02
Accepted: 2026-06-10
Published Online: 2026-06-25
DOI: 10.70454/JRIST.020204

Journal of Recent Innovation in Science and Technology
E-ISSN: 3117-3926, P- 3117-535X

[18] W. Shi et al., "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[19] A. Vaswani et al., "Attention Is All You Need," *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[20] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *International Conference on Learning Representations (ICLR)*, 2017.

[21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[22] J. Zhou et al., "Graph Neural Networks: A Review of Methods and Applications," *AI Open*, vol. 1, pp. 57–81, 2020.

[23] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, 2019.

[24] D. Silver et al., "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–489, 2016.

[25] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2021.

[26] M. Wooldridge, *An Introduction to MultiAgent Systems*, 2nd ed. Wiley, 2009.

[27] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-oriented cloud computing: Vision, hype, and reality," *Proceedings of the IEEE International Conference on High Performance Computing and Communications*, 2008.

[28] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[29] J. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[30] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.