



# IJRASET

International Journal For Research in  
Applied Science and Engineering Technology



---

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

---

**Volume:** 14    **Issue:** V    **Month of publication:** May 2026

**DOI:** <https://doi.org/10.22214/ijraset.2026.81894>

[www.ijraset.com](http://www.ijraset.com)

Call:  08813907089

E-mail ID: [ijraset@gmail.com](mailto:ijraset@gmail.com)

# A Survey on AI-Powered Adaptive Learning Using Concept Mapping and Grounded Response Generation Technology

Prof. Nagarathna C, Arun Kumar A R, Bharath M, Girish DR, Maadhu Avati

Dept. of CSE Saphthagiri College of Engineering Bengaluru, India

**Abstract:** *The proliferation of digital educational resources has intensified the demand for intelligent systems capable of understanding, organizing, and presenting knowledge in ways that align with individual learner needs. Conventional e-learning environments continue to rely on passive content delivery mechanisms, including static document repositories, manual revision scheduling, and keyword-based retrieval, all of which fail to capture the semantic structure of a subject domain or the evolving retention state of a learner. This survey systematically examines the theoretical and technical foundations underlying intelligent adaptive learning, spanning Concept Graph construction, retrieval-supported generation, dense vector search, neural knowledge tracing, spaced-repetition scheduling, and AI-driven tutoring. Building on this review, the paper introduces Neuroweave, a comprehensive adaptive learning framework that ingests uploaded educational documents, extracts domain concepts, and organizes them into a living Concept Graph. Memory retention is quantified per concept through the Neuroweave Adaptive Memory Algorithm (NAMA), which weighs review frequency, quiz performance, graph-level reinforcement from connected concepts, and time-based exponential decay. Grounded question answering is delivered by coupling large language model generation with vector retrieval over the learner's own study material, ensuring factual accuracy and source transparency. The paper presents the architectural design, component interactions, algorithmic formulation, comparative analysis against related literature, identified limitations, and directions for future research.*

**Keywords:** *Adaptive learning, Concept Graph, retrieval-supported generation, semantic vector search, memory retention modeling, personalized education, spaced repetition, knowledge tracing.*

## I. INTRODUCTION

Advances in computing infrastructure, large language models, and graph databases have opened transformative possibilities for educational technology. Despite these capabilities, the majority of widely deployed learning platforms remain fundamentally static: learners access content sequenced by curriculum designers, progress through modules at a fixed pace, and receive generalized feedback that does not reflect individual knowledge states. This structural rigidity limits learning efficiency, particularly when students must prepare for domain-specific assessments under time pressure.

Graph-theoretic representations of knowledge provide a principled mechanism for encoding the interdependencies that exist among academic concepts. A Concept Graph (KG) can capture that understanding of a prerequisite concept is foundational to comprehending advanced material, and that weakness in one node has cascading implications for semantically adjacent nodes. This relational structure enables a learning system to identify not merely which topic a student struggles with, but also which upstream gaps are responsible for downstream confusion.

Retrieval-supported generation (RSG) addresses a fundamental limitation of parametric language models in educational contexts: the tendency to produce plausible but unsupported answers. By retrieving relevant passages from the learner's uploaded documents before generation, RSG architectures tightly couple AI responses to verified source material. This grounding mechanism is especially valuable in high-stakes academic settings where correctness is non-negotiable.

Memory modeling adds a third critical dimension. Cognitive science has long established that retention decays over time in a predictable pattern and can be reinforced through strategically timed review. Algorithms that incorporate these principles can determine not merely what a student has studied, but whether that knowledge is still reliably accessible and when review is most effective.

This survey reviews the relevant literature across these intersecting domains and presents Neuroweave, an integrated platform that synthesizes Concept Graph construction, RSG-based question answering, and algorithmic memory modeling into a unified adaptive learning environment.

The remainder of this paper is structured as follows: Section II discusses background and motivation; Section III surveys related literature; Section IV analyses existing systems; Section V describes the proposed system; Section VI details the methodology; Section VII discusses findings; Section VIII concludes; and Section IX lists references.

## II. BACKGROUND AND MOTIVATION

### A. Heterogeneity of Learner Profiles

Learners entering a course differ substantially in prior exposure, cognitive processing speed, preferred modality, and recall durability. A one-size-fits-all curriculum necessarily underserves students at both ends of the proficiency spectrum. Personalized adaptive systems have the potential to address this heterogeneity by dynamically adjusting content sequencing and review timing to the inferred state of each individual.

### B. Shortcomings of Contemporary Learning Platforms

Contemporary learning management systems (LMS) are primarily content warehouses. They excel at distributing materials and tracking completion metrics but offer little intelligence regarding conceptual relationships or memory trajectories. Standalone spaced-repetition tools such as Anki improve revision efficiency for memorisation-heavy tasks but impose substantial manual overhead for card creation and require the learner to independently judge card difficulty.

General-purpose AI assistants can respond to subject-matter queries but frequently produce factually unanchored responses, particularly for specialized academic content that may postdate model training.

### C. Convergence of Concept Graphs, RSG, and Memory Algorithms

The combination of Concept Graphs, retrieval-supported generation, and memory decay algorithms creates a mutually reinforcing foundation for intelligent learning. Concept Graphs supply structured context for retrieval; retrieval grounds generation in verified source text; and memory algorithms prioritize which graph nodes require urgent review. This convergence is the conceptual basis of Neuroweave and motivates the literature examined in Section III.

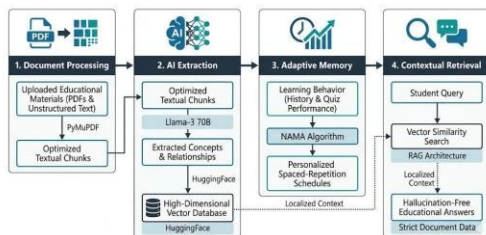


Fig. 1. The four-phase Neuroweave methodology

## III. LITERATURE SURVEY

### A. Graph-Based Adaptive Study Path Design

Authors and Year: Li, Li, and Gao (Electronics, 2026)

**Methodology:** The authors conduct a structured review of Concept Graph approaches to study path design. Their taxonomy covers learner profile modeling, prerequisite extraction, and resource sequencing. They compare graph traversal strategies and highlight how prerequisite relationships can guide curriculum ordering automatically from ontological content representations.

**Findings:** Graph-based path design consistently outperforms purely collaborative filtering approaches when prerequisite structures are accurately captured. Systems that incorporate explicit dependency edges achieve higher learner satisfaction and lower dropout rates in controlled evaluations.

**Limitation:** The majority of reviewed systems require richly annotated learner interaction logs that may not be available in early deployment phases. Graph construction quality varies significantly across subjects, and transferability to disciplines with less formalized prerequisite structures remains an open challenge.

### B. Grounded Educational Question Answering with Disciplinary Concept Maps

Authors and Year: MDKAG Research Group (Applied Sciences, 2025)

**Methodology:** This work constructs a multimodal disciplinary Concept Graph sourced from course syllabi, textbooks, and lecture transcripts. At query time, relevant graph sub-graphs are retrieved and provided as structured context to a language model fine-tuned on educational dialogue. Answer verification is performed by comparing generated spans against retrieved source passages.

**Findings:** Grounding generation in domain-specific graph context reduces hallucination rates by a measurable margin relative to graph-free RSG baselines. Question-answering accuracy improves particularly for multi-hop queries that require chaining across multiple conceptual dependencies.

**Limitation:** Constructing and maintaining a high-quality multimodal educational graph demands sustained curation effort. Automatic relationship extraction from diverse course materials introduces noise that can propagate into retrieval and generation quality.

### C. retrieval-supported generation for Knowledge-Intensive NLP Tasks

Authors and Year: Lewis et al. (NeurIPS, 2020)

**Methodology:** The RSG framework couples a dense passage retriever trained with in-batch negatives to a sequence-to-sequence generator. Given an input query, the retriever returns the top-k passages from a non-parametric document index; the generator conditions on both the query and the retrieved context to produce the output. Two variants are evaluated: RSG-Sequence and RSG-Token, which differ in how retrieval is marginalized during generation.

**Findings:** RSG achieves new state-of-the-art performance on open-domain question answering benchmarks and outperforms purely parametric baselines on knowledge-intensive tasks. The non-parametric memory component allows the model to access up-to-date information without retraining.

**Limitation:** Standard RSG operates over flat document chunks and does not capture hierarchical or relational structure between passages. In educational settings, this flat retrieval misses prerequisite relationships that are critical for pedagogically appropriate responses.

### D. Relationship-Aware Retrieval for Large Study Collections

Authors and Year: Edge et al. (2024)

**Methodology:** This approach converts large document collections into entity-link structures using automated extraction. Graph communities are identified through hierarchical clustering, and community summaries are generated offline. At query time, relevant communities are ranked and their summaries are passed to the generation model, enabling answers that synthesize information distributed across many documents.

**Findings:** For broad, thematic queries over large corpora, relationship-aware retrieval improves standard retrieval generation in comprehensiveness and diversity of response. The community-level summarization captures emergent themes that chunk-level retrieval cannot surface.

**Limitation:** Offline graph construction and community summarization adds significant computational overhead and storage requirements. System quality is sensitive to the accuracy of LLM-assisted entity extraction, which can be unreliable for specialized academic terminology.

### E. Efficient Context Linking for Retrieval

Authors and Year: Guo et al. (2024)

**Methodology:** This method combines relationship-based search with vector matching through two retrieval paths. Local retrieval identifies entity-specific context, while global retrieval identifies thematic relationships. The two paths are fused at inference time, balancing specificity with breadth.

**Findings:** Compared with plain retrieval generation, the method improves contextual coherence for complex queries requiring understanding of entity relationships. Retrieval latency remains acceptable for interactive applications due to lightweight graph indexing.

**Limitation:** Entity completeness depends on the quality of initial extraction. When source documents are poorly structured or use domain-specific abbreviations, extracted graphs are sparse, reducing the advantage of relationship-aware retrieval over flat vector search.

### F. Deep Knowledge Tracing with Recurrent Neural Networks

Authors and Year: Piech et al. (NeurIPS, 2015)

**Methodology:** Deep Knowledge Tracing (DKT) represents student knowledge as a latent hidden state maintained by a recurrent neural network. At each timestep, the model receives the most recent exercise identifier and correctness signal and updates its internal state. The probability of correctly answering any future exercise is predicted from this hidden state.

**Findings:** DKT outperforms Bayesian Knowledge Tracing on prediction accuracy across multiple learner datasets. The model captures complex temporal dependencies in learner behavior that parametric BKT models cannot represent.

**Limitation:** The latent state is not interpretable, making it difficult for educators to extract actionable insights about specific concept strengths. The model does not natively represent prerequisite relationships between skills, limiting its ability to recommend targeted interventions.

#### G. Adaptive Learning Through Educational Concept Graph Integration

**Authors and Year:** Zhang, Wang, Ma, and Wang (Frontiers in Educational Research, 2023)

**Methodology:** The proposed framework builds an educational Concept Graph from curriculum documents and links learner interaction records to graph nodes. Adaptive sequencing is achieved by selecting next-learning resources based on mastered concepts and pending prerequisite gaps identified through graph traversal. A learner disorientation metric is introduced to detect confusion states and trigger remedial content delivery.

**Findings:** Students using the graph-adaptive system demonstrated reduced study time for comparable achievement levels relative to linear curriculum controls. Disorientation events decrease over successive sessions as the system learns individual navigation preferences.

**Limitation:** The framework does not incorporate explicit memory decay modeling. As a result, the system may over-estimate mastery of concepts reviewed long ago. Richer temporal retention signals are needed to improve scheduling accuracy.

#### H. Concept Graph-Augmented Knowledge Tracing for Programming Education

**Authors and Year:** Applied Sciences Research Group (Applied Sciences, 2024)

**Methodology:** This study augments a knowledge tracing model with a programming-domain Concept Graph that encodes concept dependencies and common error patterns. Personalized answer sequences are constructed by routing learners through the graph according to their inferred mastery profile, prioritizing prerequisite concepts before advancing to dependent ones.

**Findings:** Graph-augmented knowledge tracing produces more accurate mastery predictions for programming competencies than graph-free baseline models. Personalized exercise sequences reduce the number of failed attempts before mastery is achieved.

**Limitation:** The approach is tuned to the structured prerequisite relationships inherent in programming education. Its effectiveness in domains with less formalized dependency graphs, such as humanities or interdisciplinary studies, has not been validated.

#### I. BGE Dense Embedding Models for Semantic Retrieval

**Authors and Year:** BAAI Flag Embedding Team (2023-2025)

**Methodology:** The BGE (BAAI General Embedding) model family produces dense vector representations of text through contrastive training on large-scale pairs. BGE-small-en-v1.5 produces 384-dimensional embeddings and is optimized for resource-constrained deployment. Retrieval performance is evaluated on BEIR benchmarks covering diverse domain-specific corpora.

**Findings:** BGE models achieve competitive retrieval performance relative to larger embedding models while maintaining inference latency suitable for real-time educational applications. The small variant in particular offers a favorable accuracy-cost trade-off for edge and server deployment.

**Limitation:** Dense embeddings measure semantic proximity but does not ensure logical correctness or educational accuracy of retrieved passages. Incorrect or outdated source material will be retrieved faithfully if it is semantically close to the query.

#### J. pgvector: Vector Similarity Search in Relational Databases

**Authors and Year:** pgvector Open-Source Project (ongoing)

**Methodology:** pgvector extends PostgreSQL with a native vector data type and supports approximate nearest-neighbor search using HNSW and IVFFlat indexes. Similarity functions include cosine distance, L2 distance, and inner product. The extension allows relational and vector queries to be combined in standard SQL, eliminating the need for a separate vector store.

**Findings:** For moderately sized educational knowledge bases, pgvector provides adequate retrieval throughput without the operational overhead of a dedicated vector database.

Combining relational filtering with vector search enables hybrid queries such as retrieving the most similar concept within a specific subject category.

Limitation: At very large scales, approximate indexing becomes critical and retrieval latency can degrade. Default index parameters may not be optimal for domain-specific embedding distributions, requiring empirical calibration.

#### IV. EXISTING SYSTEM

Contemporary digital learning environments occupy a broad spectrum from simple file repositories to sophisticated learning management systems, yet most share fundamental architectural limitations that constrain their effectiveness for personalized adaptive learning.

Learning management systems such as Moodle, Canvas, and Blackboard organize course content, distribute assignments, and record submission events. Their internal data model is fundamentally document-centric: lectures, slides, and reading materials are stored as files with minimal semantic annotation. Progress tracking is limited to binary completion flags or assessment grades, providing no signal about conceptual understanding depth or retention durability over time.

Standalone spaced-repetition applications apply forgetting-curve scheduling to discrete flashcard items. These tools are effective for vocabulary and fact memorisation but require learners to manually create card sets, assign difficulty ratings, and maintain card decks as course content evolves. Critically, they treat each card as an independent memory unit, discarding the relational structure between concepts that could inform more intelligent review prioritization.

Keyword-based search engines retrieve documents containing matching terms but cannot capture semantic equivalences, identify conceptually related passages using different vocabulary, or rank results by pedagogical relevance. A student searching for a concept using informal language may fail to retrieve authoritative explanations phrased in technical terminology.

General-purpose AI assistants based on large language models can engage in subject-matter dialogue but suffer from parametric hallucination when responding to specialized queries outside their training distribution. They retain no memory of individual learner history across sessions and cannot anchor responses to specific uploaded study materials unless explicitly provided with context in each interaction. The principal deficiencies of existing tools include absence of semantic concept modeling, lack of retention-aware scheduling, inability to represent prerequisite relationships, high manual data-entry burden, and insufficient factual grounding of AI-generated responses.

#### V. PROPOSED SYSTEM

Neuroweave is an AI-powered adaptive learning platform engineered to transform uploaded educational documents into a dynamic, queryable, retention-aware knowledge environment. The system addresses each identified limitation of existing tools through a cohesive integration of modern AI components.

Upon document upload, Neuroweave extracts raw text, segments it into semantically coherent chunks, identifies key concepts and their relationships using a large language model, and stores the resulting structured data as a Concept Graph in a relational-vector database. This graph persists across sessions and continuously updates as the learner adds new materials, completes quizzes, and requests study plans.

The frontend is implemented in React 18 with TypeScript and Vite, providing a responsive interface styled with Tailwind CSS and Motion for animations. Core interface modules include a personalized dashboard, an interactive 3D-style brain map for graph visualisation, an upload module, an insights panel displaying AI-generated recommendations, a study planner with weekly schedules, a memory heatmap, and an Ask Your Brain interface for grounded question answering.

The backend is a FastAPI application deployed on Render. It exposes REST endpoints for dashboard metrics, Concept Graph data, document uploads, study plan generation, memory analytics, AI insights, and RSG question answering. Supabase PostgreSQL with the pgvector extension serves as the primary data store, holding concept nodes, weighted relational edges, review logs, upload metadata, and 384-dimensional BGE-small vector embeddings.

Language model inference for concept extraction and AI response generation is provided by the Groq API using the Llama-3.3-70B model, which delivers sub-second response latency. Embedding generation uses the BGE-small-en-v1.5 model locally, ensuring low-latency semantic search without external API dependency. The Neuroweave Adaptive Memory Algorithm (NAMA) continuously updates per-concept memory strength scores, driving weak-area detection, brain-map color coding, and study-plan priority ordering.

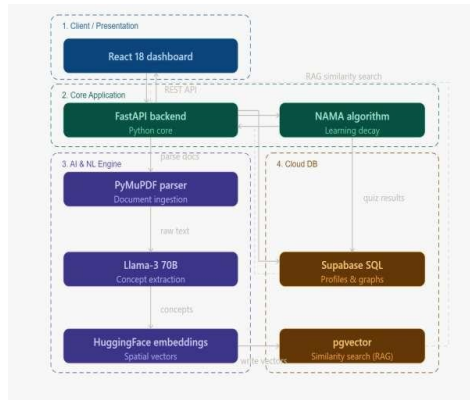


Fig.2. System Architecture Diagram of Neuroweave

## VI. METHODOLOGY

### A. User Interaction and Dashboard Layer

The learner's primary entry point is a personalized dashboard that surfaces a holistic view of their learning state. Displayed metrics include an overall knowledge score, subject-level retention rates, the count of mastered versus weak concepts, scheduled upcoming reviews, and AI-generated study recommendations tailored to the current memory profile. The brain map provides an interactive visual rendering of the Concept Graph, where node color encodes NAMA-computed memory strength: green for high retention, amber for moderate, and red for critically weak concepts requiring immediate review.

### B. Document Ingestion and Text Extraction

The ingestion pipeline accepts PDF files, plain text documents, Markdown files, and directly pasted content. For PDF inputs, the system applies a multi-strategy extraction approach: native digital text extraction is attempted first, with an OCR fallback for scanned or image-based pages. Extracted text is pre-processed to normalize whitespace, remove artifacts, and segment the document into overlapping chunks of configurable length. Overlapping windows ensure that concept references spanning chunk boundaries are captured in at least one chunk.

### C. AI-Assisted Concept Extraction

Each text chunk is submitted to the Llama-3.3-70B model with a structured prompt instructing the model to identify domain concepts, assign categorical labels, summarise key ideas, and infer directional relationships between identified concepts. The model returns a structured JSON object that is validated against a schema before database insertion. In the event of API unavailability or response validation failure, a deterministic fallback using TF-IDF keyword extraction ensures pipeline continuity, albeit with reduced semantic richness.

### D. Embedding Generation and Vector Storage

Concept labels, summaries, and source text snippets are independently embedded using BGE-small-en-v1.5, producing 384-dimensional float vectors. All embeddings are stored in the pgvector-enabled Supabase PostgreSQL database alongside their associated concept records. An HNSW index is maintained on the embedding column to support approximate cosine-similarity queries at retrieval time. This vector infrastructure enables both concept relationship discovery during graph construction and document-grounded answer retrieval during RSG question answering.

### E. Concept Graph Construction and Maintenance

Extracted concepts are stored as graph nodes with the following attribute schema: unique identifier, text label, domain category, source content snippet, current NAMA strength score, cumulative review count, weighted quiz score, and embedding vector. Edges between concepts are recreated in two ways: explicitly, where the LLM identifies a directed relationship between two extracted concepts, and implicitly, where the cosine similarity between concept embeddings exceeds a configurable threshold. Each edge carries a weight reflecting relationship strength. The graph is incrementally extended with each new document upload without duplicating nodes for concepts already present in the database.

#### F. NAMAMemoryStrengthComputation

The Neuroweave Adaptive Memory Algorithm computes a memory strength score  $S$  for each concept node by combining four weighted signals. The review component  $R_c$  increases logarithmically with cumulative review count, rewarding consistent engagement while exhibiting diminishing returns. The quiz performance component  $Q_c$  is the exponential moving average of quiz scores on exercises linked to the concept, reflecting demonstrated recall accuracy. The graph reinforcement component  $G_c$  is computed as the weighted average strength of all directly connected nodes, capturing the pedagogical principle that a concept embedded in a well-understood conceptual neighborhood is intrinsically more durable. The time decay component  $D_c$  applies an exponential decay function to the interval since the most recent review, modeling the Ebbinghaus forgetting curve. These components are combined as:  $S = w_1 * R_c + w_2 * Q_c + w_3 * G_c - w_4 * D_c$ , where the weights  $w_1$  through  $w_4$  are globally configurable parameters subject to empirical calibration.

Concepts with  $S$  below a weak-area threshold are flagged for priority scheduling.

#### G. RSG-Based Question Answering and Study Plan Generation

When a learner submits a question through the Ask Your Brain interface, the query text is embedded using BGE-small and compared against stored concept embeddings via cosine similarity. The top- $k$  most similar concept nodes are retrieved along with their source text snippets. This retrieved context is formatted into a structured prompt that explicitly instructs the language model to base its answer exclusively on the provided passages and to cite the relevant concept when doing so. The study planner reads NAMA strength scores nightly and constructs a weekly review schedule that allocates session slots to concepts proportionally to their weakness severity, grouping related concepts within the same session to leverage RSG graph reinforcement effects during active review.

## VII. DISCUSSION

Neuroweave advances the state of the art in adaptive learning by addressing the three structural deficiencies that characterise most existing platforms: absence of concept relationship modeling, inability to track retention trajectories, and lack of factual grounding for AI-generated responses. The unified architecture ensures that each system component reinforces the others: Concept Graph edges provide the graph reinforcement signal consumed by NAMA, NAMA scores guide both study planning and brain-map visualisation, and the same vector embeddings that power graph construction also underpin RSG retrieval.

The primary strength of the graph-centric design is its ability to surface second-order learning insights. A student who struggles with a specific concept can be shown not only the weak node but also the upstream prerequisite nodes whose incomplete mastery is likely responsible. This causal framing transforms remediation from a guessing exercise into a principled diagnostic activity.

Several implementation challenges require careful attention.

Concept extraction accuracy depends substantially on the semantic coherence of source documents; loosely structured notes yield noisier graphs than well-organized textbooks.

Relationship inference via embedding similarity introduces false edges when domain-adjacent concepts are semantically close but pedagogically independent. The current NAMA weight configuration was derived analytically rather than from empirical learner data, and may not optimally reflect the memory dynamics of all subject types or learner profiles.

Privacy is a salient concern given that the system processes and stores learners' personal study documents. Future versions must implement robust access controls, end-to-end encryption for uploaded content, and clear data retention policies.

Evaluation of AI-generated answers against expert-curated ground truth is also needed to quantify hallucination rates in practice and inform further prompt engineering.

## VIII. CONCLUSION

This survey has examined the theoretical landscape of AI-powered adaptive learning by reviewing foundational contributions in Concept Graph construction, retrieval-supported generation, vector search, neural knowledge tracing, and memory modeling. Existing work demonstrates the individual value of each component but rarely integrates them into a cohesive, deployment-ready system accessible to individual learners without institutional infrastructure.

Neuroweave addresses this gap by providing an end-to-end platform that transforms uploaded documents into a living Concept Graph, computes per-concept retention scores through NAMA, and grounds AI-generated answers in verified source material via vector retrieval and large language model generation. The system reduces learner effort for material organisation, increases revision targeting precision, and provides transparent, source-anchored responses to subject-matter queries.



Avenues for future research include longitudinal validation of NAMA with diverse real-world learner cohorts, automated weight calibration through reinforcement learning from learner outcome feedback, multimodal Concept Graph construction from lecture videos and diagrammatic content, privacy-preserving federated learning across institutional deployments, and integration of real-time collaborative learning features that allow peer Concept Graphs to inform individual study recommendations.

## REFERENCES

- [1] A. Li et al., *Electronics*, vol. 15, no. 1, 2026.
- [2] P. Lewis et al., *NeurIPS*, vol. 33, pp. 9459-9474, 2020.
- [3] Recent study on relationship-aware retrieval, 2024. \n\n[4] Recent study on efficient context linking, 2024.
- [4] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein, "Deep Knowledge Tracing," *Advances in Neural Information Processing Systems*, vol. 28, 2015.
- [5] S. Zhang, X. Wang, Y. Ma, and D. Wang, "An Adaptive Learning Method Based on Concept Graph," *Frontiers in Educational Research*, vol. 6, no. 6, 2023.
- [6] BAAI Flag Embedding Team, *embedding model documentation*, 2023.
- [7] PostgreSQL vector-search extension documentation.
- [8] J. Shi et al., *Applied Sciences*, vol. 15, no. 16, 2025.
- [9] Y. Zhou et al., *Applied Sciences*, vol. 14, no. 17, 2024.
- [10] A. C. Graesser, S. Lu, G. T. Jackson, H. H. Mitchell, M. Ventura,
- [11] Olney, and M. M. Louwerse, "AutoTutor: A Tutor with Dialogue in Natural Language," *Behavior Research Methods, Instruments, and Computers*, vol. 36, no. 2, pp. 180-193, 2004.
- [12] M. VanLehn, "The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems," *Educational Psychologist*, vol. 46, no. 4, pp. 197-221, 2011.



10.22214/IJRASET



45.98



IMPACT FACTOR:  
7.129



IMPACT FACTOR:  
7.429



# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24\*7 Support on Whatsapp)