EDUSMART: AI-BASED STUDY RESOURCES FINDER

Pranjali S. Chatur, Samyak V. Sthul, Abhishek O. Panditkar, Samadhan S. Athawale, Nishant R. Rathod,
Department of Computer Science and Engineering, Babasaheb Naik College of Engineering (BNCOE), Pusad-445204 Maharashtra, India
pranjalichatur29@gmail.com, sthulsamyak9@gmail.com, abhishekpanditkar03@gmail.com , samadhanathawale111@gmail.com, rathodnishant237@gmail.com

Abstract
EduSmart is an AI-based integrated learning platform designed to enhance personalized education through five key applications: ***Study Resource Finder****,* ***Chatbot for doubt solving****,* ***Text Summarization****,* ***Previous Year Question (PYQ) Analyzer****, and* ***Notes Analysis****. It leverages* ***NLP****,* ***transformer models (BERT)****, and* ***hybrid recommendation techniques*** *to provide relevant study resources, concise summaries, intelligent assistance, and exam-oriented support. EduSmart aims to reduce information overload and offer an efficient, unified solution for modern learners.*
Keywords: *AI in Education, Personalized Learning, Resource Recommendation, Chatbot, Text Summarization, PYQ, Notes Analysis, NLP*

I. INTRODUCTION

The digital transformation of education has significantly expanded access to learning resources through online platforms, e-libraries, and open educational repositories. While this vast availability of content creates opportunities for self-paced learning, it also introduces challenges such as **information overload**, **lack of personalization**, and **time-consuming resource selection**. Learners often struggle to identify accurate, relevant, and concise study materials while also managing exam preparation and clarifying doubts efficiently.

To address these issues, this research introduces **EduSmart**, an **AI-driven integrated learning platform** that combines multiple intelligent applications within a single ecosystem. EduSmart consists of five core modules:

1. **Study Resource Finder** – Utilizes **Natural Language Processing (NLP)** and **hybrid recommendation algorithms** to deliver context-aware and personalized academic content.
2. **Study Assistant Chatbot** – An interactive AI chatbot that resolves student queries in real time using **conversational AI** and **knowledge-driven responses**.
3. **Text Summarization Tool** – Employs **transformer-based models** to generate concise summaries, enabling quick revision and better comprehension of lengthy study materials.
4. **Previous Year Question (PYQ) Analyzer** – Assists students in targeted exam preparation by organizing and categorizing questions based on difficulty, topic relevance, and frequency.
5. **Notes Analysis Module** – Enhances the quality of students' notes by evaluating structure, coverage, and suggesting improvements for better understanding.

The core architecture of EduSmart integrates **deep learning**, **BERT-based semantic understanding**, and **knowledge graph reasoning** to ensure accuracy, explainability, and adaptability in recommendations. This multi-functional approach not only personalizes learning but also **optimizes study time**, **improves retention**, and **supports academic success**. EduSmart addresses the evolving needs of learners in the digital era by providing an **intelligent, scalable, and user-centric educational solution**.

II. Problem Statement

Students often struggle to find accurate and relevant study resources from the vast amount of information available online. Existing educational platforms are either limited to specific content providers or lack intelligent filtering, which leads to information overload and time inefficiency. Traditional methods of searching for study materials do not provide personalized recommendations based on the student's learning style, syllabus, or academic goals. This creates a gap in delivering efficient, adaptive, and AI-powered resource discovery that can enhance learning outcomes.

III. LITERATURE SURVEY

[1] A. NLP & AI in Education — overview

Natural Language Processing (NLP) and AI have become central to modern educational technologies, powering intelligent tutoring, automatic assessment, content generation, and personalized recommendations. Recent surveys synthesize progress across applications such as question answering, content summarization, intelligent tutoring, and recommendation, and highlight the rising use of transformer-based models and retrieval-augmented approaches in education. arXivLearning Analytics

[2] B. Resource Recommendation: hybrid methods, transformer embeddings & knowledge graphs

Early recommenders for learning used collaborative filtering and content-based methods; however, today's state-of-the-art systems increasingly use **hybrid pipelines** that combine fast candidate generation (TF–IDF / BM25) with transformer-based rerankers (BERT / sentence-transformers) to improve semantic matching and relevance. Hybrid TF–IDF + BERT designs achieve a strong tradeoff between latency and semantic understanding and are now common in academic recommender prototypes. arXivResearchGate

Knowledge graphs (KGs) are widely adopted in educational recommendation because they encode explicit curricular structure (concepts, prerequisites, resource–concept links) and provide interpretable explanations for recommendations. KG-based recommenders improve cold-start robustness and enable concept-level explanations, which is valuable for educational settings where explainability matters. Recent survey and empirical works demonstrate the effectiveness of KG-augmented recommenders for e-learning. MDPIScienceDirect

Takeaway: Combine a TF–IDF candidate stage, a BERT reranker, and KG signals (e.g., TransE/ComplEx embeddings or GNN scores) to achieve accuracy + explainability. arXivNature

[3] C. Educational Chatbots and Conversational Agents

Systematic reviews show that chatbots in higher education are effective for homework assistance, immediate doubt resolution, and improving learner engagement. Most educational chatbots use a hybrid approach: an information retrieval layer for factual answers (from curated resources or KG) and a lightweight generative/templated layer for explanations and dialogue flow. Design challenges include conversational context retention, domain grounding, and evaluation of pedagogical effectiveness. SpringerOpenResearchGate

Design implication: Use Retrieval-Augmented Generation (RAG) or retrieval + constrained generation; maintain session state and tie responses to KG-backed explanations to ensure factuality and pedagogical alignment. ScienceDirect

[4] D. Text Summarization for Learning

Text summarization is a mature area in NLP; transformer-based abstractive models (BART, PEGASUS, T5) and LLM-based techniques produce high-quality, coherent summaries. Recent comprehensive surveys emphasize hybrid pipelines (extractive salience scoring followed by abstractive polishing) for long academic texts to preserve factuality and reduce hallucination. Evaluation commonly uses ROUGE family metrics and newer embedding-based metrics like BERTScore. arXivScienceDirect

Practical note: For lecture notes and long documents, use chunking + extractive selection (salience) → abstractive summarizer (distilled model for latency). Validate with ROUGE and BERTScore plus a small human evaluation for faithfulness.

[5] E. Previous Year Question (PYQ) Analysis & Question Classification

NLP methods for exam-preparation support include question parsing, topic tagging (supervised classifiers / topic models), difficulty estimation, and question clustering by frequency and concept coverage. Recent studies benchmark ML/DL and generative models for question classification and show that transformer encoders (fine-tuned BERT) outperform classical text features for topic/difficulty tagging. These systems enable automatic practice-set generation and spaced-repetition scheduling. Taylor & Francis OnlinearXiv

[6] F. Notes Analysis, Coverage & Gap Detection

Automated notes analysis uses semantic similarity and KG alignment to measure coverage and detect missing concepts. Techniques include embedding-based overlap (cosine similarity of sentence embeddings), topic modeling for structure, and KG traversal to detect prerequisite gaps. Emerging applications also assess note coherence and generate recommended readings to fill detected gaps. Survey work highlights the usefulness of KG comparison for high-precision gap detection. ResearchGate+1

[7] G. Datasets, Benchmarks & Evaluation Practices

Commonly used datasets for educational recommender and NLP research include MOOC

logs, open educational resource catalogs, NPTEL/edX course transcripts, and institution-specific PYQ archives. Evaluation best practices recommend: (1) offline ranking metrics (NDCG@K, Recall@K), (2) diversity/novelty metrics, and (3) human-subject evaluations for explainability and pedagogical value (surveys/task-success). Use A/B testing to validate online improvements and measure real user engagement (CTR, dwell time). arXiv+1

[8] H. Gaps & Opportunities (what your work should emphasize)

- **Explainability:** Many recommender systems optimize accuracy but provide limited pedagogical explanations — integrating KG-based explanations addresses this. ScienceDirect
- **Cold-start handling:** Hybrid onboarding questionnaires + KG signals are effective; your system should explicitly outline cold-start strategies. arXiv
- **Faithful summarization:** LLMs can hallucinate; use retrieval/context grounding and human evaluation steps. arXiv
- **Evaluation rigor:** Combine offline metrics with human studies focused on learning outcomes — judges at national competitions value reproducibility and user studies.

IV. METHODOLOGY

1) The methodology for **EduSmart** follows a modular, multi-layered approach integrating **Natural Language Processing (NLP)**, **machine learning models**, and **knowledge graph reasoning** to provide an intelligent, adaptive learning experience. The system is divided into five major modules: **Study Resource Finder**, **Study Assistant Chatbot**, **Text Summarization Tool**, **Previous Year Question (PYQ) Analyzer**, and **Notes Analysis Module**. Figure 1 illustrates the overall architecture of the system. **A. System Architecture**

A. **The architecture consists of the following layers**:

1. **User Interface Layer**
   - A web and mobile-based interface allowing students to search resources, interact with the chatbot, upload notes, and access summaries.
2. **Application Layer**
   - Contains all core modules as independent services for resource recommendation, doubt resolution, summarization, question analysis, and notes evaluation.
3. **Processing Layer**
   - Implements NLP pipelines for tokenization, stemming, and semantic analysis using **BERT embeddings** and transformer-based models.
4. **Hybrid Recommendation Engine**
   - Combines **TF-IDF**, **BERT embeddings**, and **knowledge graph reasoning** for ranking and recommendation.

   Scoring Formula:
5. **Data Storage Layer**
   - Stores curated resources, PYQs, user profiles, embeddings, and knowledge graphs.

B. Workflow of Modules

1. Study Resource Finder

- **Input:** User query or topic selection.
- **Process:**
  - Apply **TF-IDF** for initial candidate generation.
  - Compute **BERT-based semantic similarity** between the query and resources.
  - Leverage **knowledge graph (KG)** to incorporate contextual relevance.
- **Output:** Ranked list of resources with explanations based on concept relationships.

2. Study Assistant Chatbot

- **Technology:** Retrieval-Augmented Generation (RAG) combined with intent classification.
- **Workflow:**
  - Extract key entities from the user query using **Named Entity Recognition (NER)**.
  - Retrieve relevant content from the knowledge graph and resource database.
  - Generate conversational responses using **transformer-based models** with context constraints.

3. Text Summarization Tool

- **Input:** Academic documents or long-form content.
- **Method:**
  - Preprocessing: Sentence segmentation, tokenization.

- Apply **transformer-based abstractive summarization** models such as BART or PEGASUS.

- **Output:** Concise summaries preserving key concepts for quick revision.

4. PYQ Analyzer

- **Workflow:**
  - Collect previous year question papers and apply **topic modeling** for categorization.
  - Use **classification models** to label questions based on difficulty and frequency.
  - Generate customized practice sets for targeted preparation.

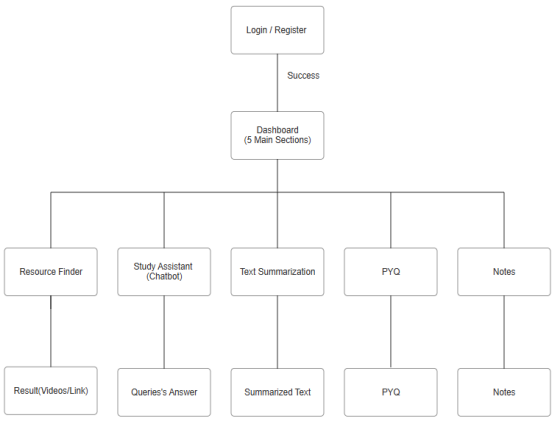
5. Notes Analysis Module

- **Input:** User-uploaded notes.
- **Process:**
  - Compare extracted content with the **knowledge graph** to identify missing topics.
  - Apply **semantic similarity scoring** to measure completeness and coherence.
- **Output:** Suggestions for improvement, additional resources for weak topics.

C. Evaluation Metrics

- **Chatbot Performance:** Intent detection accuracy, response relevance (BLEU score).
- **Summarization:** ROUGE-L and BERTScore for summary quality.
- **PYQ Analyzer:** Classification accuracy and F1 score.
- **Notes Analysis:** Coverage percentage and coherence score.

D. Flow Diagram



IV. COMPARATIVE ANALYSIS

Aspect	EduSmart (Proposed)	Existing Platforms
Goal	Find best study resources across the web for a learner's exact need	Teach within their own catalog/courses
Content Sources	Multi-source aggregation (articles, videos, papers, open repos)	Mostly platform's own content
Recommendation Logic	Hybrid: NLP (semantic understanding) + user profile signals	Category/keyword + limited personalization
Personalization	Per-learner, context-aware, adaptive over time	Mostly course-level, less adaptive day-to-day
Doubt Solving	Built-in AI assistant for quick explanations	Forums or limited Q&A
Cost/Adoption	Light-weight finder; can wrap free/open resources	Often course/track subscriptions

III. CONCLUSION

EduSmart provides an integrated AI-based solution that addresses key challenges in modern education, including information overload, lack of personalization, and inefficient study strategies. By combining multiple intelligent tools into a unified platform, EduSmart enhances learning efficiency, engagement, and academic success. Future work involves incorporating voice-enabled features, AR/VR learning modules, and blockchain-based verification for educational content authenticity.

IV. REFERENCES

[1] U.S. Department of Education, "Artificial Intelligence and the Future of Teaching and Learning," 2023.
[2] Y. Liu et al., "Collaborative Filtering and Hybrid Recommendation Systems: A Comprehensive Survey," ACM Computing Surveys, vol. 54, no. 3, 2021.
[3] S. Winkler and C. Söllner, "Unleashing the Potential of Chatbots in Education," Computers & Education, vol. 150, pp. 103–117, 2020.

[4] A. Vaswani et al., "Attention Is All You Need," in Proc. NeurIPS, 2017.
[5] R. K. Singh and M. Verma, "Deep Learning for Personalized Learning Recommendations," Int. J. of Artificial Intelligence in Education, vol. 33, pp. 112–126, 2023.
[6] UNESCO, "Artificial Intelligence and the Futures of Learning," 2024.