Intelligent Course Assistant: A User-Centric AI Learning Platform

Su Yongyuan 3036661020 Ren Liubo 3036654675 Chen Zhixin 3036654053

Xue Chaowei 3036657536 Yu Erfei 3036657732

Abstract

This proposal introduces an intelligent course assistant that empowers graduate students to take control of their learning journey through adaptive AI support. Unlike fully automated AI tutors, our platform operates on a human-AI collaboration model where students maintain decision-making authority while AI provides enhanced capabilities. Students choose when to access authoritative course materials, decide when community discussion is needed, and customize their study resources—all within a 24/7 available intelligent environment. The system addresses critical gaps in current course learning: fragmented resources, lack of instant feedback, and absence of personalized study tools. Expected impact includes improving student response time from hours to seconds, enhancing learning efficiency by 30%, and creating a sustainable knowledge repository for Data Mining and future graduate courses.

1 Problem Statement

1.1 Current Pain Points in Graduate Course Learning

Resource Fragmentation Students navigate scattered lecture PDFs, past exam papers, and supplementary materials without unified access. Critical information (formula derivations, solution approaches) is buried across 200+ slides. An estimated 30% of study time is wasted on searching rather than learning.

Lack of Immediate Learning Support Students struggle with concepts outside of class hours without instant clarification. There is no personalized feedback when working through practice problems independently. Waiting for next lecture or office hours interrupts learning momentum.

Absence of Personalized Learning Tools One-size-fits-all materials don't accommodate diverse learning paces. Students lack structured methods to identify and strengthen weak areas. Traditional study aids (handwritten notes, generic cheat sheets) are inefficient and incomplete.

Lost Knowledge & Community Disconnect Valuable student questions don't reach course forums due to posting friction. Future students repeatedly encounter same obstacles without benefit of past discussions. High-quality discussions that could benefit the learning community remain in private messages.

2 Proposed Solution

2.1 System Overview

We propose a **user-centric intelligent course assistant** combining Agentic AI with Retrieval-Augmented Generation (RAG). The system operates on a core principle: *"AI provides capabilities, students retain control."*

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

2.2 Why Agent + RAG Architecture?

From a product perspective, our architecture offers:

- **Contextual Intelligence**: Agent understands when to pull exact course content vs. engage in exploratory discussion
- **Transparent Provenance**: Every answer citing course materials shows exact PDF page or exam question number
- **Adaptive Response Strategy**: Handles diverse queries—from definition lookups to comparative analysis requiring synthesis
- **User-Directed Retrieval**: Students explicitly choose when to use course-specific knowledge vs. general LLM reasoning

2.3 Core Design Philosophy

Unlike traditional chatbots with black-box responses, our system emphasizes transparency and user control. Students aren't passive recipients—they actively direct all key functions through explicit choices.

3 Dataset Description

3.1 Overview

Our intelligent course assistant is built upon a comprehensive knowledge base constructed from official HKU course materials.

3.2 Data Sources

The knowledge base consists of two primary categories:

Course Lecture Materials (Moodle Platform) All official lecture content published on the university's Moodle learning management system:

- Lecture slides in PowerPoint format (PPT/PPTX)
- Lecture notes and supplementary materials in PDF format

Historical Examination Papers (ExamBase) Past final exam papers from the university's official ExamBase repository:

- Previous years' final examination papers
- Covering multiple academic years

3.3 Data Characteristics

- **Authority**: All materials officially published by course instructors
- **Traceability**: Every retrieved answer maintains source citation (lecture number, slide number, exam year)
- **Comprehensiveness**: Covers complete course curriculum

4 System Features

4.1 Feature 1: Intelligent Knowledge Q&A (Agent + RAG)

Product Description *"You control the knowledge boundary—toggle course materials for authoritative answers, or explore freely with open-ended AI discussion."*

Core Capability Students interact through a conversational interface with a **prominent knowledge base toggle**:

- **Knowledge Base ON**: System retrieves from lecture PDFs and past exams with direct citations
- **Knowledge Base OFF**: General LLM reasoning for exploratory questions

Example Interaction

```
Student: "Explain the difference between decomposition-first
        and interleaved decomposition in agent planning"

[Knowledge Base: ON]

System Response:
According to Lecture 8 (Slides 26-27), there are two main
approaches:

Decomposition-First Method
* Strategy: Split entire task first, execute sequentially
* Advantages: Clear objectives, parallel processing
* Disadvantages: Low fault tolerance, poor adaptability

Interleaved Decomposition Method
* Strategy: Alternate between decomposition and execution
* Advantages: Dynamic adaptability, high fault tolerance
* Disadvantages: Lower efficiency, difficult parallelization

[Source: Data Mining Lecture 8, Slides 26-27]
```

4.2 Feature 2: Forum Intelligent Publishing Tool

Product Description *"Transform complex questions into high-quality forum posts with one click—let community wisdom accelerate your learning."*

Example - Auto-Generated Forum Post

```
MOODLE FORUM POST DRAFT

Subject: Understanding orchestrator-worker architecture in
        DeepResearch case study

Message:
I'm reviewing the DeepResearch case study (Lecture slides
73-75) and have questions about the orchestrator-worker
architecture.

**What I understand:**
- LeadResearcher agent plans strategy and spawns specialized
  subagents
- Sub-agents run in parallel with own context

**My confusion:**
How does the LeadResearcher decide how many subagents to
spawn? Is it based on query complexity, or predefined rules?

**Chatbot response summary:**
System explained it's dynamic based on task decomposition,
but I'm unclear on the exact decision mechanism.

Would appreciate clarification. Thanks!

[Preview] [Edit] [Publish to Forum]
```

4.3 Feature 3: Personalized Study Report Generator

Product Description *"Generate custom study materials in seconds—from comprehensive cheat sheets to targeted concept reviews."*

Example - RAG Fundamentals Review Sheet

```
DATA MINING - RAG Architecture Review Sheet
Generated for: Lectures 7-8

1. WHAT IS RAG?
RAG = LLMs + External Retrieval
Purpose: Enhance LLMs, reduce hallucination
Flow: Query -> Retriever -> Generator -> Answer

2. THREE-STEP PROCESS
1. INDEXING: Split docs -> Embeddings -> Vector DB
2. RETRIEVAL: Query vector -> Search chunks
3. GENERATION: Query + Context -> LLM -> Answer

3. ARCHITECTURE COMPONENTS
[Documents] -> Sanitizer -> Splitter -> Knowledge DB
[Query] -> Processor -> Retriever -> Generator -> [Answer]

Key Roles:
- Sanitizer: Cleans data
- Splitter: Breaks into chunks
- Knowledge DB: Stores vectors
- Retriever: Finds relevant chunks
- Generator: LLM synthesis

4. EMBEDDINGS
Vector representations (BERT, OpenAI)
Enable semantic search beyond keywords

5. KEY ADVANTAGES
Reduces LLM hallucinations through grounded retrieval
Enables access to updated/domain-specific knowledge
Preserves context while managing token limits
Supports scalable knowledge management

6. COMMON CHALLENGES
Context length limitations for complex queries
Hallucination reduction (requires better evidence structuring)
Retrieval accuracy (relevance vs. similarity trade-offs)
Scaling training paradigms for improved reasoning
```

5 User Journey Scenarios

5.1 Scenario 1: Exam Preparation

Studying Agent Architectures

- **Initial Learning (KB ON)**: Gets detailed answer citing DeepResearch case study
- **Exploration (KB OFF)**: Discusses real-world scenarios without course constraints
- **Community**: Publishes auto-drafted forum post for peer/TA input

5.2 Scenario 2: Assignment Support

Working on RAG Implementation

- **Quick Lookup (KB ON)**: Instant cited definition from lecture slides
- **Implementation (KB OFF)**: Python code guidance for his specific task

6 System Architecture

6.1 High-Level Architecture

The system consists of layers:

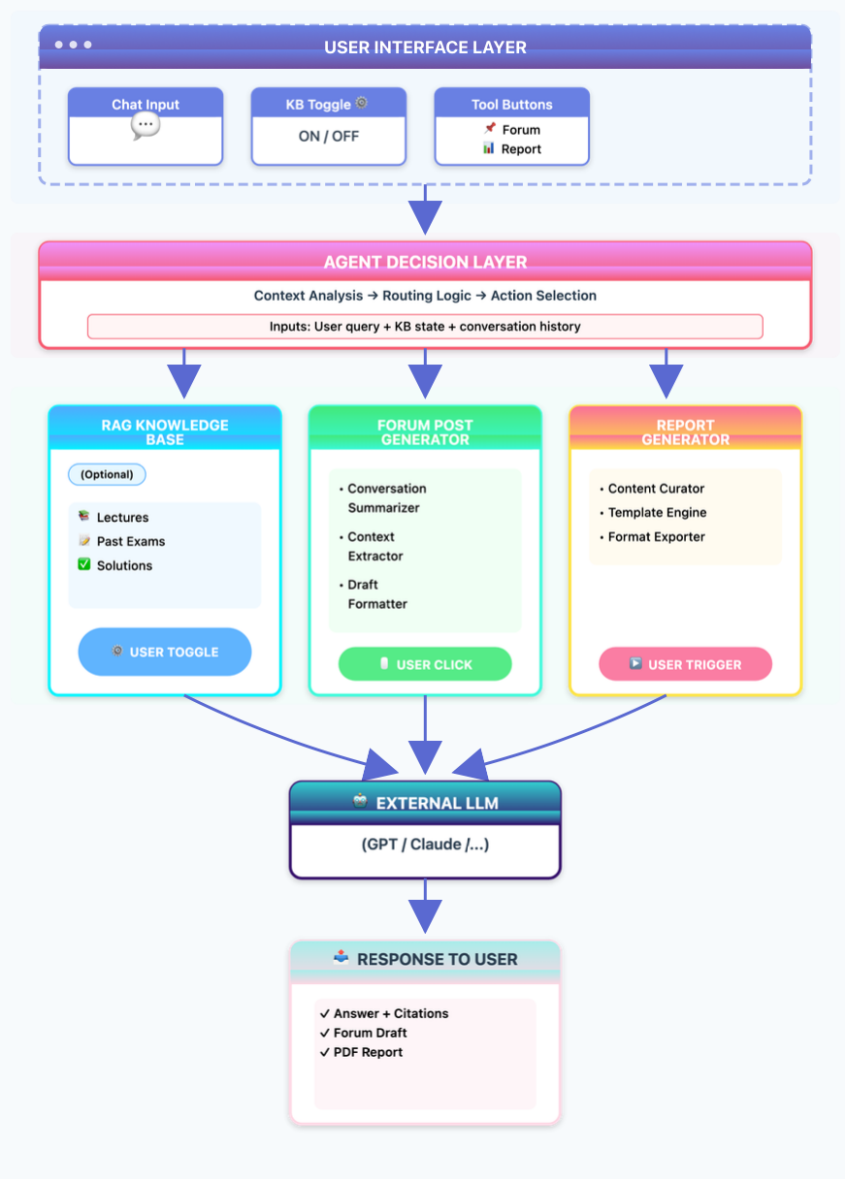

Figure 1: Architecture Diagram

6.2 User Interaction Flow

The interaction follows a user-centric decision flow:

Figure 2: User Interaction Flow Diagram

7 Conclusion

This intelligent course assistant reimagines AI learning tools through **user-centric design**. By prioritizing student autonomy over automation, we create a system that enhances rather than replaces human judgment.

Core Differentiators

1. User-directed knowledge retrieval (control precision vs. exploration)
2. Human-AI collaboration (bridges individual and community learning)
3. Customization over automation (reports to student specifications)