

StudyBuddy: An AI-Powered RAG Learning Ecosystem with Emotion-Aware Adaptive Hypothesis Generation

Aakash Dilip Jadhav, Dr. Chandrika.J, Chandan HK, Dhanush CT, Jaydev H S

Department of Computer Science and Engineering
Malnad College of Engineering, Hassan
(An Autonomous Institution under VTU,
Belagavi) Hassan, Karnataka – 573201, India

Abstract—Traditional e-learning platforms deliver static educational content without adapting to learner context, emotional state, or individual knowledge gaps. This paper presents StudyBuddy, a full-stack, AI-powered learning ecosystem that addresses these limitations through three integrated intelligent subsystems: (1) a Retrieval-Augmented Generation (RAG) engine built on FAISS vector indexing and LangChain for context-aware document querying; (2) a Random Forest-based emotion detection model trained on a Kaggle text emotion dataset to infer the learner's affective state from conversational input; and (3) a Find-S algorithm-based hypothesis learner that constructs the most-specific consistent hypothesis from the student's interaction history, which is then forwarded alongside the user's query to the primary LLM for emotionally and contextually enriched responses. The platform supports multi-mode search (PDF-core, web intelligence, and fast research), a voice-first “Voice Orb” interface powered by low-latency Groq STT/TTS, and persistent session memory via Supabase PostgreSQL. Experimental evaluation demonstrates high retrieval fidelity from uploaded academic PDFs, strong emotion classification accuracy on the Kaggle benchmark, and positive qualitative feedback from student users regarding personalised response quality. StudyBuddy demonstrates that combining RAG, affective computing, and symbolic hypothesis learning produces a materially richer tutoring experience than RAG alone.

Index Terms—retrieval-augmented generation, emotion detection, random forest, Find-S algorithm, adaptive learning, FAISS, LangChain, voice interface, personalised education, LLM

I. INTRODUCTION

Students in higher education increasingly rely on digital resources to supplement classroom instruction, yet most platforms treat every learner identically — delivering the same content in the same tone regardless of whether the student is confused, frustrated, confident, or curious [6]. This one-size-fits-all paradigm limits engagement, reduces retention, and wastes the considerable potential of modern large language models (LLMs) to serve as personalised digital tutors.

Recent advances in Retrieval-Augmented Generation (RAG) [1] address the factual accuracy problem by grounding LLM responses in verified source documents retrieved at inference time. Meanwhile, affective computing research has shown that recognising a learner's emotional state and adjusting instructional tone accordingly leads to measurably better out-

comes [3]. What has been missing is a platform that unifies these capabilities — high-fidelity document retrieval, emotion awareness, and adaptive hypothesis tracking — into a cohesive, deployable learning environment.

StudyBuddy was built to fill that gap. It is a full-stack web and voice platform in which a student can upload academic PDFs, ask questions in text or speech, and receive answers that are grounded in their own study materials, calibrated to their detected emotional state, and shaped by a running hypothesis about their knowledge profile. The system achieves this through three independently trained components wired together at the inference pipeline level:

- A **RAG engine** using LangChain's `RecursiveCharacterTextSplitter` for document segmentation, FAISS for per-user vector indexing [2], and Llama 3.3 70B via Groq for contextual synthesis.
- An **emotion detection model** — a Random Forest classifier trained from scratch on a Kaggle text emotion dataset — that labels each incoming message with an affective class (e.g., joy, sadness, anger, fear, surprise, neutral).
- A **Find-S hypothesis learner** that maintains the most-specific consistent hypothesis over the student's session features (topics queried, emotion trajectory, search modes used) and appends that hypothesis to each LLM prompt, enabling the model to tailor explanatory depth and tone continuously.

The remainder of this paper is structured as follows. Section II surveys related work across RAG, emotion detection, and adaptive learning. Section III describes the system architecture and each technical component. Section IV details the dataset and training methodology for the emotion model and hypothesis learner. Section V presents experimental results. Section VI concludes and outlines future work.

II. LITERATURE REVIEW

We reviewed work across five areas directly relevant to StudyBuddy: RAG for education, vector similarity search, text-based emotion detection, adaptive learning systems, and voice-enabled educational interfaces.

A. Retrieval-Augmented Generation in Education

Lewis et al. [1] introduced RAG as a framework that combines parametric memory (the LLM's weights) with non-parametric memory (a retrieved document index) to reduce hallucination and support domain-specific question answering. Klesel and Wittmann [5] extended this analysis to educational settings, showing that RAG-based FAQ systems markedly outperform static chatbots in factual accuracy on course-specific content. Zamfirescu-Pereira et al. [4] conducted a systematic survey of RAG applications in interactive learning systems and found that retrieval-grounded generation significantly reduces the rate of factually incorrect answers compared to pure generative approaches. A limitation common to these works is that they focus on retrieval fidelity alone, without incorporating learner affect into the generation pipeline — a gap StudyBuddy directly addresses.

B. Vector Similarity Search

Douze et al. [2] described the FAISS library as a toolkit of indexing methods for large-scale approximate nearest-neighbour search over dense embeddings, published in IEEE Transactions on Big Data. FAISS underpins the per-user vector stores in StudyBuddy. Riyadh et al. [10] demonstrated that combining LLM reasoning with vector similarity search improves recall for complex or nuanced queries where semantic proximity alone is insufficient, directly motivating our use of LangChain retrieval chains atop FAISS.

C. Text-Based Emotion Detection

Macho'va' et al. [3] trained and evaluated multiple classifiers — including Random Forest, SVM, Naive Bayes, and LSTM-based deep networks — for text emotion detection using Kaggle conversational data. Their finding that Random Forest achieves performance competitive with deep models for six-class emotion recognition on moderate-sized datasets motivated our choice of Random Forest as the primary classifier in StudyBuddy. The Springer review by Acheampong et al. [9] surveyed emotion detection methods from 2013 to 2023, confirming that ensemble methods such as Random Forest remain strong baselines, particularly when training data is limited and interpretability is valued.

D. Adaptive and Personalised Learning

Gligorea et al. [6] reviewed AI/ML integration in e-learning systems from 2010 to 2023 and found that personalising content delivery to individual learner profiles — rather than presenting a uniform curriculum — consistently improves engagement and performance. Alshammari et al. [7] conducted a systematic literature review of ML-based personalised adaptive learning platforms and found that combining behavioural signals (interaction patterns, performance history) with content-based retrieval produces superior personalisation compared to either approach alone. The Find-S hypothesis learning component in StudyBuddy operationalises this insight by maintaining a symbolic profile of each student's inferred knowledge state and forwarding it to the LLM.

E. Voice-Enabled Educational Interfaces

Neumann et al. [8] built and evaluated MoodleBot, an LLM-powered chatbot for higher-education database courses, and found that conversational access — rather than query-box access — substantially increased student interaction frequency. Their work underscores the importance of low-friction interfaces in educational AI. The StudyBuddy “Voice Orb” extends this principle to speech, using Groq's Whisper-based STT and Llama-based TTS to minimise latency.

F. The Gap This Paper Addresses

Reviewing the above work reveals a consistent pattern: RAG systems improve factual grounding but ignore learner affect; emotion detection systems identify affective states but do not feed them into retrieval-augmented pipelines; adaptive systems personalise content but rarely integrate real-time emotion signals or symbolic hypothesis tracking. StudyBuddy unifies all three into a single deployable platform.

III. METHODOLOGY

A. System Architecture

StudyBuddy follows a layered, service-oriented architecture. The frontend is a React 18 single-page application built with Vite and Tailwind CSS v4. The backend is a FastAPI server (Python 3.10+) that exposes asynchronous REST endpoints consumed by the frontend. Persistent data — user profiles, document metadata, session transcripts, and flashcards — is stored in a Supabase PostgreSQL instance with row-level security policies applied to every table. Figure 1 illustrates the full-stack architecture.

At inference time, each user message passes through three sequential stages before reaching the LLM:

- 1) **Emotion classification:** the Random Forest model assigns an affective label to the message text.
- 2) **Hypothesis update:** the Find-S learner updates the student's hypothesis object with the new interaction features.
- 3) **RAG retrieval:** the FAISS index for the student's uploaded documents is queried for semantically relevant chunks, which are prepended to the prompt along with the emotion label and hypothesis.

B. RAG-Based Knowledge Extraction

PDF documents are uploaded via `POST /api/upload`. The backend uses LangChain's `RecursiveCharacterTextSplitter` to segment each document into overlapping chunks (chunk size: 1000 tokens, overlap: 200 tokens). Each chunk is embedded using a sentence-transformer model and stored in a FAISS index keyed to the student's UUID, persisted under `vectorstore/{student_id}/`. At query time, the top- k ($k=5$) most semantically similar chunks are retrieved and injected into the LLM prompt as grounded context. This architecture follows the standard dense-retrieval RAG pattern described by Lewis et al. [1], ensuring that every LLM

TABLE I
SYSTEM COMPONENT SUMMARY

Component	Technology	Role
Frontend	React 18, Vite, Tailwind CSS v4	UI & Client-side State
Backend	FastAPI, Python 3.10+	Async REST API
Vector Engine	FAISS, LangChain	Embeddings & Similarity Search
Primary LLM	Llama 3.3 70B (Groq)	Contextual Synthesis
Database	Supabase (PostgreSQL)	Persistence & Auth
Voice	Groq Whisper / Llama TTS	STT/TTS Processing
Emotion Model	Random Forest (scikit-learn)	Affective State Inference
Hypothesis	Find-S Algorithm	Student Knowledge Profile

response is traceable to a specific passage in the student's own study materials.

C. Emotion Detection Model

1) *Dataset*: The emotion classifier is trained on the *Emotion Detection from Text* dataset available on Kaggle [12]. The dataset contains approximately 40,000 labelled English sentences drawn from social-network conversational content, annotated with six emotion classes: *joy*, *sadness*, *anger*, *fear*, *surprise*, and *neutral*. We use an 80/20 train-test split, with stratified sampling to preserve class distribution.

2) *Preprocessing*: Each sentence is lowercased, stripped of punctuation, and tokenised. Stop words are removed using NLTK. The text is then vectorised using TF-IDF with a maximum vocabulary of 10,000 features and unigram/bigram n -gram range.

3) *Model Training*: A Random Forest classifier (100 estimators, *gini* criterion, *max_depth=None*) is trained on the TF-IDF feature matrix. Random Forest was chosen following Macho'va' et al. [3], who demonstrated that it achieves near-LSTM accuracy on this type of data while remaining fast and interpretable. The trained model and TF-IDF vectoriser are serialised with *joblib* and loaded into the FastAPI server at startup.

4) *Inference Integration*: At inference time, the user's raw message text is passed through the vectoriser and classifier. The predicted emotion label and its confidence score are appended to the system prompt forwarded to the LLM, e.g.: "[Student emotion: *sadness* (confidence: 0.81)]. Adjust tone accordingly — be encouraging and patient."

D. Find-S Hypothesis Learner

1) *Motivation*: The Find-S algorithm [11] is a symbolic concept-learning procedure that, given a set of positive training examples, finds the most-specific hypothesis consistent with all of them. In the StudyBuddy context, we treat each student

session as a stream of positive examples of the student's learning profile, and Find-S is used to progressively specialise a hypothesis about what the student knows and how they prefer to learn.

2) *Feature Space*: Each interaction is represented as a feature vector with the following attributes:

- **Search mode**: *pdf*, *web*, or *research*
- **Detected emotion**: one of the six classes from the emotion model
- **Topic cluster**: coarse topic label derived by keyword extraction from the user query (e.g., *mathematics*, *biology*, *programming*, *general*)
- **Query complexity**: *simple*, *moderate*, or *complex* (classified by query length and syntactic depth)
- **Session stage**: *early*, *mid*, or *late* (based on interaction count in the session)

3) *Algorithm Operation*: The hypothesis is initialised to the most-specific state, $h = [\emptyset, \emptyset, \emptyset, \emptyset, \emptyset]$. For each incoming interaction (treated as a positive example), the hypothesis is generalised minimally: if a feature value in h is $\emptyset$, it is set to the observed value; if it already holds a value that differs from the observation, it is replaced with the wildcard "?". This produces the most-specific hypothesis consistent with all observed interactions in the session.

4) *Forwarding to the LLM*: After each update, the current hypothesis is serialised into a natural-language descriptor and prepended to the LLM system prompt, e.g.:

"Student profile hypothesis: prefers PDF-based search; currently experiencing sadness; primarily studying programming topics at moderate complexity; mid-session. Adjust explanation depth and tone accordingly."

This allows the LLM to tailor vocabulary, example choice, and instructional scaffolding to the inferred student profile without requiring explicit user configuration.

E. Multi-Mode Search

StudyBuddy exposes three search modes via POST `/api/notes/generate`:

- **PDF Core**: retrieves chunks from the student's FAISS vector store and synthesises a response grounded entirely in their uploaded documents.
- **Web Intelligence**: uses Groq's built-in web search tool to fetch real-time external content before synthesis, suitable for topics not covered in uploaded PDFs.
- **Fast Research**: performs rapid chunked analysis of the uploaded documents for immediate extraction of key facts, definitions, and summaries.

F. Voice-First Interface

The "Voice Orb" interface captures audio from the browser microphone and streams it to Groq's Whisper-based STT endpoint. The transcribed text enters the full inference pipeline (emotion detection → hypothesis update → RAG retrieval → LLM synthesis). The LLM response text is then streamed

back to the Groq TTS endpoint and played to the student. End-to-end voice latency is minimised by Groq’s hardware-accelerated inference stack.

G. Database Schema

All persistent state is stored in Supabase PostgreSQL with the schema summarised in Table II.

TABLE II
SUPABASE DATABASE SCHEMA

Table	Purpose
students	User profiles, authentication, hash-based IDs
documents	Uploaded PDF metadata, storage paths, processing status
talks	Chat and voice interaction history for session memory
flashcards	AI-generated learning units linked to document contexts

IV. DATASET AND TRAINING DETAILS

A. Emotion Detection Dataset

The Kaggle *Emotion Detection from Text* dataset [12] contains 40,000 English sentences annotated with six emotion labels. Table III shows the class distribution after stratified splitting.

TABLE III
EMOTION DATASET CLASS DISTRIBUTION (TRAINING SPLIT)

Emotion Class	Training Samples	Proportion (%)
Joy	6,870	21.5
Sadness	6,430	20.1
Anger	5,120	16.0
Fear	4,860	15.2
Surprise	4,570	14.3
Neutral	4,150	12.9
Total	32,000	100.0

B. Find-S Hypothesis Initialisation

The hypothesis learner requires no separate training dataset. It is initialised fresh at the start of each session and updated incrementally as interactions occur. The feature schema described in Section III-D is fixed at system design time. No offline training phase is required, which keeps the system lightweight and stateless across sessions.

V. RESULTS AND DISCUSSION

A. Emotion Detection Performance

Table IV reports classification metrics for the Random Forest emotion model on the 8,000-sample test split.

The macro-averaged F1 of 0.83 is consistent with results reported by Macho’va’ et al. [3] for Random Forest on analogous Kaggle conversational data. Joy and sadness — the

TABLE IV
EMOTION DETECTION — RANDOM FOREST TEST RESULTS

Class	Precision	Recall	F1	Support
Joy	0.89	0.91	0.90	1,718
Sadness	0.87	0.85	0.86	1,608
Anger	0.83	0.81	0.82	1,280
Fear	0.82	0.84	0.83	1,215
Surprise	0.80	0.78	0.79	1,143
Neutral	0.78	0.80	0.79	1,036
Macro Avg	0.83	0.83	0.83	8,000

most frequent classes — achieve the highest scores, while surprise and neutral, which share overlapping lexical features, are slightly lower.

B. RAG Retrieval Quality

In a user study with 30 students who uploaded their own course PDFs and posed ten questions each, 87% of LLM responses were rated as “relevant” or “highly relevant” by the student, and 91% cited a passage traceable to the uploaded document. These figures are consistent with benchmarks reported for similar FAISS-backed RAG systems in educational contexts [4].

C. Hypothesis-Augmented Response Quality

To evaluate the contribution of the Find-S hypothesis, a within-subjects comparison was conducted where 20 students received responses both with and without the hypothesis prepended. Students rated hypothesis-augmented responses as “better personalised” in 74% of paired comparisons. Qualitative comments indicated that the adapted tone and targeted examples were the primary drivers of this preference, consistent with findings by Gligorea et al. [6] on the value of learner profile personalisation.

D. Voice Interface Usability

Voice Orb transcription accuracy was measured against manual transcripts of 200 student utterances. Word Error Rate (WER) was 6.2%, and end-to-end voice latency (from end of utterance to start of synthesised response) averaged 1.8 seconds, within the 2-second threshold identified as perceptually acceptable in interactive voice applications.

E. Discussion of the Reviewed Works in Context

The RAG literature [1], [4], [5] establishes that document-grounded generation is a robust baseline for educational QA. StudyBuddy confirms this but adds that grounding alone is insufficient when learner state varies widely — the emotion and hypothesis layers materially improve subjective quality ratings. Macho’va’ et al. [3] justified our choice of Random Forest over deep networks for emotion detection; our results reproduce their finding that RF achieves >0.80 macro-F1 on Kaggle conversational data with far lower inference overhead than LSTM alternatives. Alshammari et al. [7] and Gligorea

et al. [6] both argued that combining behavioural signals with content retrieval outperforms either strategy alone; the within-subjects comparison above provides empirical support for this claim in the StudyBuddy context. The FAISS benchmarks in Douze et al. [2] guided our index configuration (IVF with flat quantiser) to balance search latency and recall for per-user indices of the sizes typical in a semester's worth of PDFs.

VI. CONCLUSION

This paper presented StudyBuddy, an AI-powered learning ecosystem that integrates three complementary intelligent sub-systems into a unified educational platform: a FAISS-backed RAG engine for document-grounded question answering, a Random Forest emotion classifier trained on Kaggle text data, and a Find-S algorithm hypothesis learner that builds and forwards a continuously updated student knowledge profile to the LLM. Together, these components enable emotionally calibrated, personalised, and factually grounded responses that adapt dynamically to each student's affective state and learning trajectory.

Experimental results confirm strong emotion classification (macro-F1: 0.83), high RAG relevance (87% user rating), and a clear user preference for hypothesis-augmented responses (74% of paired comparisons). The voice interface achieves a 6.2% WER and 1.8-second latency, making speech interaction practical in everyday study sessions.

Future work includes: semantic topic modelling to replace the keyword-based topic clustering in the hypothesis feature space; adaptive difficulty adjustment in flashcard generation based on the emotion and hypothesis state; multi-modal input (images, lecture slides); federated learning to improve the emotion model across institutions while preserving data privacy; and integration of an intelligent scheduling module to recommend study times based on the learner's emotional trajectory across sessions.

REFERENCES

- [1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Ku'ttler, M. Lewis, W.-T. Yih, T. Rockta'schel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 9459–9474, 2020.
- [2] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazare', M. Lomeli, L. Hosseini, and H. Je'gou, "The Faiss Library," *IEEE Transactions on Big Data*, 2024. DOI: 10.1109/TBDATA.2024.3348836.
- [3] K. Macho'va', M. Szabo'ova', J. Paralic', and M. Mic'ko, "Detection of Emotion by Text Analysis Using Machine Learning," *Frontiers in Psychology*, vol. 14, 2023. DOI: 10.3389/fpsyg.2023.1190326.
- [4] Y. Zhao, J. Wang, Q. Li, and X. Chen, "Retrieval-Augmented Generation for Educational Application: A Systematic Survey," *Computers and Education: Artificial Intelligence*, 2025. DOI: 10.1016/j.caeai.2025.100578.
- [5] M. Klesel and H. F. Wittmann, "Retrieval-Augmented Generation (RAG)," *Business & Information Systems Engineering*, vol. 67, no. 4, pp. 551–561, 2025.
- [6] I. Gligorea, M. Cioca, R. Oancea, A.-T. Gorski, H. Gorski, and P. Tudorache, "Adaptive Learning Using Artificial Intelligence in e-Learning: A Literature Review," *Education Sciences*, vol. 13, no. 12, p. 1216, 2023. DOI: 10.3390/educsci13121216.
- [7] M. Alshammari, R. Alshammari, and M. Hooper, "Personalised Adaptive Learning Technologies Based on Machine Learning Techniques to Identify Learning Styles: A Systematic Literature Review," *IEEE Access*, vol. 11, 2023. DOI: 10.1109/ACCESS.2023.3276439.
- [8] A. Neumann, Y. Yin, S. Sowe, S. Decker, and M. Jarke, "An LLM-Driven Chatbot in Higher Education for Databases and Information Systems," *IEEE Transactions on Education*, vol. 68, pp. 103–116, 2025. DOI: 10.1109/TE.2024.3469120.
- [9] F. A. Acheampong, H. Nunoo-Mensah, and W. Chen, "A Review on Emotion Detection by Using Deep Learning Techniques," *Artificial Intelligence Review*, 2024. DOI: 10.1007/s10462-024-10831-1.
- [10] M. Riyadh, M. Li, F. H. Lie, J. L. Loh, H. Mi, and S. Bohra, "LLM-Assisted Vector Similarity Search," arXiv preprint arXiv:2412.18819, 2024.
- [11] T. M. Mitchell, *Machine Learning*. New York: McGraw-Hill, 1997.
- [12] P. Gupta, "Emotion Detection from Text," Kaggle Dataset, 2021. [Online]. Available: <https://www.kaggle.com/datasets/pashupatigupta/emotion-detection-from-text>