

INTELLICLASS AI: An AI-Powered Virtual Classroom System

¹B.E Ramesh, ²Shreyas B Acharya, ³Naveen V, ⁴Sumanth R K, ⁵Ummar Farukh

¹Associate Professor, Dept. of CS&E, SJMIT, Chitradurga, Karnataka, India.

²Seventh semester Student, SJMIT Chitraturga, Karnataka, India.

³Seventh semester Student, SJMIT Chitraturga, Karnataka, India.

⁴Seventh semester Student, SJMIT Chitraturga, Karnataka, India.

⁵Seventh semester Student, SJMIT Chitraturga, Karnataka, India.

Abstract : Artificial Intelligence (AI) has created new opportunities for developing intelligent and interactive educational systems, but conventional online learning platforms largely depend on static materials, recorded lectures, or basic conversational tools. This paper presents IntelliClass AI, an AI-powered virtual lecturer designed to transform instructor-provided educational content into interactive, AI-generated lectures. The system processes PDF course materials by extracting and organizing textual and visual information, converts the extracted text into searchable vector representations, and uses multimodal Retrieval-Augmented Generation (RAG) with a Gemini-based generative AI model to generate course-grounded teaching scripts. The generated lecture is divided into structured speech, pause, engagement, image, and video events and processed through a unified Text-to-Speech (TTS) architecture supporting Head TTS/ Kokoro ONNX and Eleven Labs. Speech alignment and phoneme information are converted into a common viseme representation to synchronize the lecturer's facial and mouth movements. The generated audio, subtitles, media, and timing information are coordinated through Supabase Realtime, while the lecturer browser acts as a conductor and student browsers locally render the interactive three-dimensional classroom and avatar using Three.js and React Three Fiber. The system also supports student questions, AI-generated responses, classroom media projection, speech caching, and synchronized lecture control.

IndexTerms - Artificial Intelligence, Virtual Lecturer, Multimodal Retrieval-Augmented Generation, Generative AI, Text-to-Speech, HeadTTS, 3D Avatar, Lip Synchronization, Viseme, Three.js, WebGL, Supabase Realtime, Intelligent Education.

I. INTRODUCTION

The rapid development of Artificial Intelligence (AI), Natural Language Processing (NLP), and real-time computer graphics has created new opportunities for improving digital education. Conventional online learning systems mainly depend on static study materials, recorded lectures, and instructor-led sessions, which may provide limited personalization and real-time interaction. To address these limitations, IntelliClass AI is proposed as an intelligent virtual lecturer that can transform instructor-provided educational content into an interactive lecture experience using generative AI and a real-time three-dimensional avatar.

The proposed system uses Retrieval-Augmented Generation (RAG) to ground AI-generated lectures in the instructor's course material. Educational PDF documents are processed to extract textual and visual information, while the extracted text is divided into meaningful chunks and stored as searchable vector representations. When a lecture is requested, relevant course information is retrieved and provided to a Gemini-based generative AI model to generate a structured teaching script. The system can also incorporate extracted diagrams and images into the virtual classroom to support visual learning.

The generated lecture is converted into speech using a unified Text-to-Speech (TTS) architecture supporting HeadTTS/Kokoro ONNX and ElevenLabs. Speech alignment and phoneme information are converted into visemes to control the facial and mouth movements of the virtual lecturer. The three-dimensional classroom and avatar are rendered locally in the student's browser using Three.js and React Three Fiber, while Supabase Realtime synchronizes audio, subtitles, avatar animation, media, and classroom events between the lecturer and connected students.

Real-time lecture delivery requires efficient processing of speech and classroom events to maintain a smooth learning experience. IntelliClass AI addresses this requirement by dividing generated lecture content into smaller natural-language speech units and using lookahead processing to prepare upcoming audio while the current segment is being played. Each speech segment is associated with timing information, sequence data, and viseme information, allowing connected clients to maintain synchronized audio and avatar animation. This approach reduces initial speech latency, minimizes interruptions between lecture segments, and enables the system to respond more effectively to student interactions during an ongoing virtual lecture.

Thus, IntelliClass AI integrates multimodal RAG, generative AI, neural speech synthesis, lip-synchronized 3D avatar animation, real-time communication, and interactive student participation into a unified virtual teaching environment. The system aims to provide a more engaging and intelligent alternative to conventional digital lectures while reducing the manual effort required for lecture preparation and delivery. The implemented system is evaluated using real speech synthesis, viseme, bandwidth, and AI generation latency benchmarks to demonstrate its practical feasibility.

II. NEED OF THE STUDY.

There is a growing need for intelligent educational systems that can transform conventional course materials into interactive and personalized learning experiences. Existing digital learning platforms largely depend on static documents, recorded lectures, or basic online communication, requiring significant manual effort from instructors and offering limited real-time interaction. IntelliClass AI addresses this need by combining instructor-provided course content with Retrieval-Augmented Generation, generative AI, neural speech synthesis, and a real-time 3D virtual lecturer, enabling course-grounded lectures to be automatically generated, spoken, visually presented, and synchronized for students. The system also provides interactive features such as student questioning, classroom media presentation, and synchronized lecture control, making it a practical approach for developing more engaging and intelligent virtual learning environments.

III. RESEARCH METHODOLOGY

3.1 System Architecture

The proposed IntelliClass AI system follows a modular polyglot architecture.

The major components are:

Table 3.1: system architecture components

Component	Technology	Function
Frontend	Next.js 16, React 19, TypeScript	Application and classroom UI
3D Engine	Three.js, React Three Fiber	Avatar and classroom rendering
State Management	Zustand	Client-side classroom state
AI Backend	Python, FastAPI	PDF processing and AI operations
PDF Processing	PyMuPDF	Text and image extraction
Generative AI	Gemini	Lecture and answer generation
TTS Provider 1	HeadTTS + Kokoro ONNX	Local neural speech synthesis
TTS Provider 2	ElevenLabs	Optional cloud speech synthesis
Database	Supabase PostgreSQL	Application and vector data
Storage	Supabase Storage	PDFs, media and generated files
Realtime	Supabase Realtime	Classroom event synchronization

3.2 Overall Data Flow

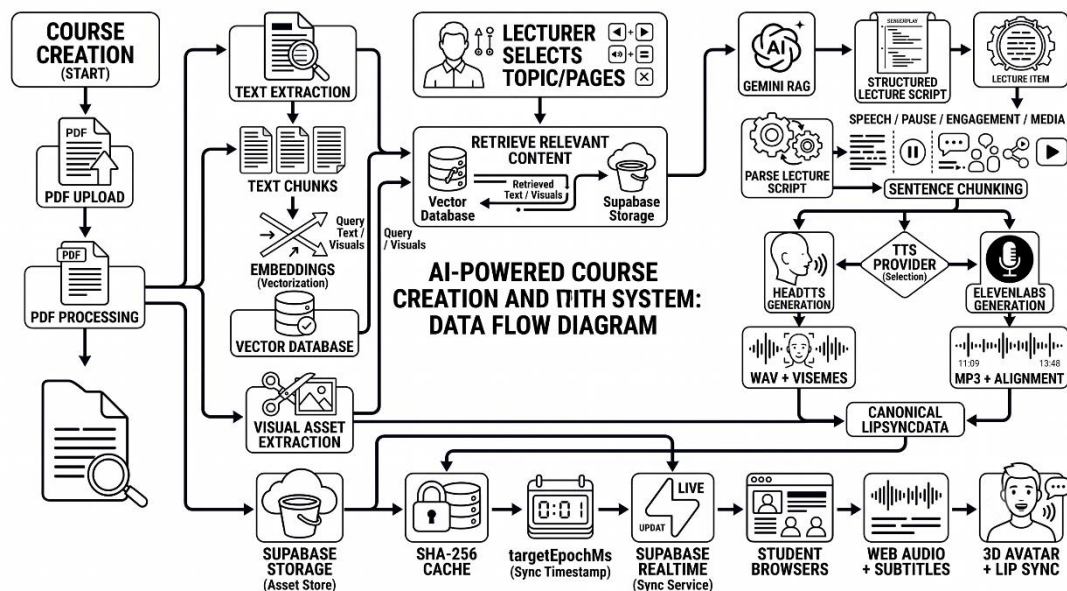


Figure 3.2: AI-powered course creation and delivery system

3.3 PDF Course Material Processing

The PDF Course Material Processing module is the first major stage of IntelliClass AI, where instructor-provided educational documents are converted into structured knowledge that can be used by the AI lecturer. When a course PDF is uploaded, the system analyzes the document and separates its educational content into **textual information and visual resources** such as images, diagrams, and figures. The extracted text is cleaned and divided into smaller meaningful sections so that individual concepts can be efficiently searched and retrieved later. Each section is transformed into a numerical representation called a **vector embedding**, which captures the semantic meaning of the content and allows the system to find information based on meaning rather than only exact keywords. These embeddings and their associated course information are stored in the database for later retrieval.

At the same time, important visual elements from the PDF are preserved separately so they can be displayed during the lecture when relevant. This processing creates a searchable course knowledge base that enables IntelliClass AI to generate lectures using the instructor's actual learning material rather than relying only on the general knowledge of the AI model.

3.4 Text Chunking and Embeddings

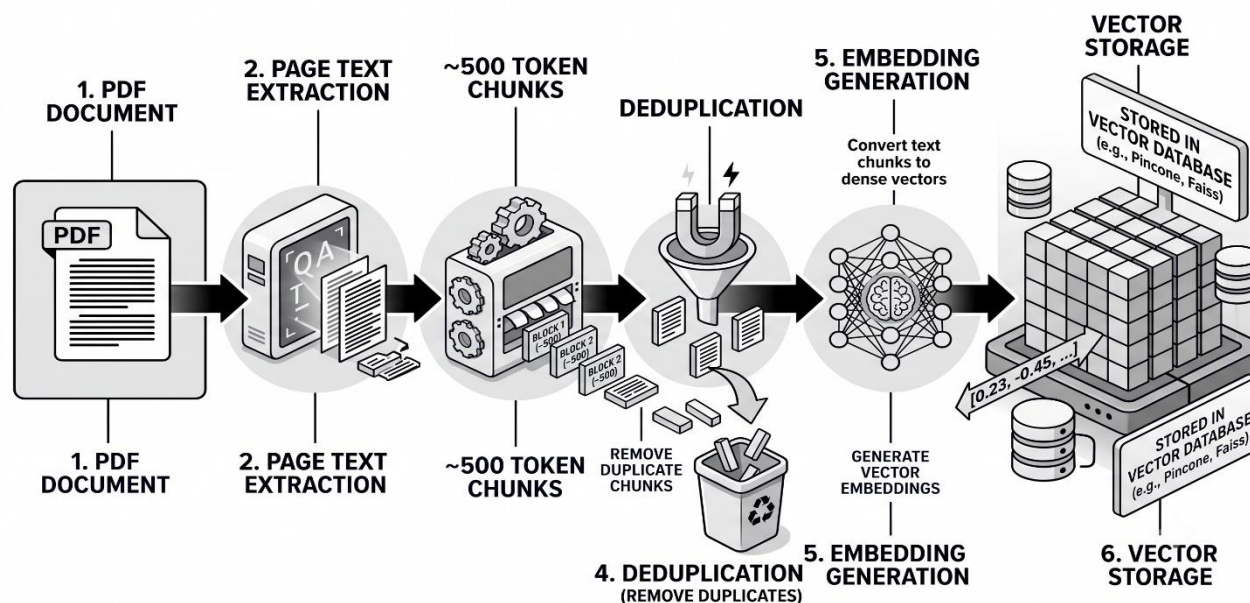


Figure 3.4: text chunking and embeddings flow

Large documents cannot efficiently be provided to an LLM as a single context. Therefore, extracted text is divided into smaller chunks. A vector embedding converts text into a numerical representation. Similar concepts produce vectors that are close in the embedding space, allowing relevant course material to be retrieved during RAG.

3.5 Multimodal Retrieval-Augmented Generation

The **Multimodal Retrieval-Augmented Generation (RAG)** module is the core intelligence layer of IntelliClass AI. It connects the processed course material with the generative AI model so that the virtual lecturer can generate lectures based on the instructor's actual content rather than relying only on general AI knowledge.

3.5.1 Course Knowledge Retrieval

When a lecture is requested, the system first identifies the relevant information from the stored course knowledge base. The student's or instructor's selected topic is used to retrieve semantically related sections from the processed PDF content. This allows the system to locate the most relevant concepts even when the words used in the request are different from the exact wording in the document.

3.5.2 Semantic Vector Search

The extracted course content is represented using **vector embeddings**, where each content section is converted into a numerical representation of its meaning. When a topic or query is provided, it is also represented in the same semantic space. The system compares these representations to identify the most relevant course sections. This semantic retrieval approach helps IntelliClass AI retrieve meaningful educational information instead of depending only on keyword matching.

3.5.3 Multimodal Content Integration

IntelliClass AI does not treat the course as text alone. Along with textual knowledge, the system maintains visual resources such as **images, diagrams, and figures** extracted from the educational PDFs. When relevant visual material is associated with the retrieved course content, it can be incorporated into the generated classroom presentation. This enables the virtual lecturer to explain a concept while displaying supporting visual material.

3.5.4 Context Preparation

The retrieved course information is organized into a contextual input for the generative AI model. The selected content provides the model with the knowledge required to generate an explanation for the requested topic. This process helps keep the generated lecture aligned with the instructor-provided learning material and reduces the possibility of generating content unrelated to the selected course.

3.5.6 AI Lecture Generation

The prepared context is provided to a **Gemini-based generative AI model**, which generates the lecture content in a teaching-oriented format. Instead of returning only a simple answer, IntelliClass AI can generate structured instructional content containing explanations and classroom events such as **speech, pauses, engagement activities, images, and videos**. These events are subsequently processed by the classroom and avatar modules for lecture delivery.

3.5.7 Grounded and Interactive Learning

The combination of retrieval and generation allows IntelliClass AI to provide lectures that are both **course-grounded and interactive**. The RAG layer supplies relevant educational knowledge, while the generative AI layer converts that knowledge into understandable lecture content. This creates a continuous flow from **course material** → **relevant knowledge** → **AI-generated explanation** → **virtual lecturer** → **student interaction**, forming the foundation of the IntelliClass AI virtual classroom.

3.6 Lecture Script Generation and Speech Processing

After the relevant course content is retrieved, IntelliClass AI generates a **structured lecture script** using the Gemini-based generative AI model. The generated script is designed specifically for virtual teaching and can contain spoken explanations along with classroom events such as **pauses, student engagement activities, images, and videos**. These events allow the system to control not only what the virtual lecturer says but also what happens in the classroom during the lecture. For example, an image can be displayed when a concept requires visual explanation, while a pause or engagement event can provide students with time to understand the topic or participate in the session. Before speech synthesis, the generated lecture text is processed and divided into smaller sentence and clause-level segments. Punctuation such as periods, commas, questions, and other natural boundaries is preserved so that the TTS system can produce more natural speech and appropriate pauses. The resulting speech segments are generally maintained within approximately **60–75 characters**, providing a balance between natural speech flow and fast synthesis. Very small or punctuation-only segments are removed to prevent empty or invalid audio generation. The system further uses a **lookahead processing strategy**, where upcoming speech segments are prepared while the current segment is being played. Since the HeadTTS synthesis process operates faster than real-time playback, this approach allows subsequent audio to be available before it is required, reducing noticeable gaps between sentences. Each generated speech segment is also associated with sequence and timing information, allowing audio playback, subtitles, classroom events, and avatar visemes to remain synchronized. This processing pipeline enables IntelliClass AI to deliver long AI-generated lectures smoothly while also supporting interruption, student interaction, and continuous real-time avatar animation.

3.7 TTS Provider Architecture

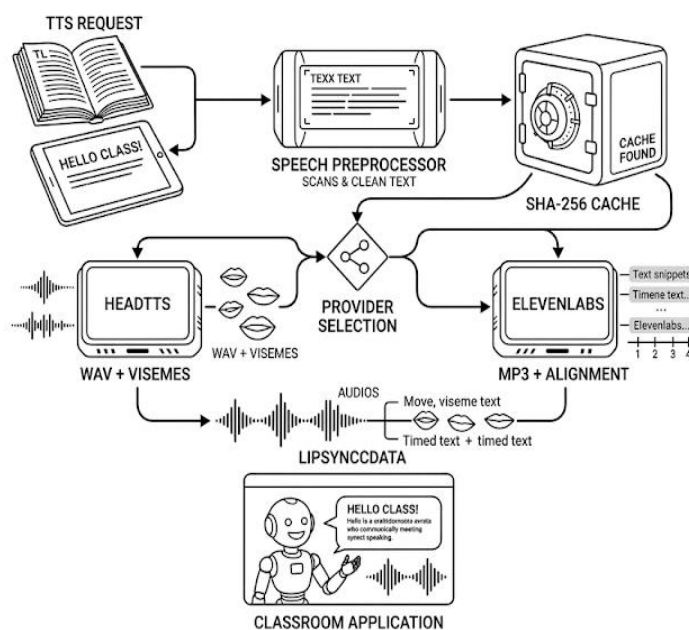


Figure 3.7: tts provider architecture

Before speech synthesis, each lecture segment is preprocessed and passed through the selected Text-to-Speech (TTS) provider, such as HeadTTS/Kokoro ONNX or ElevenLabs, to generate the corresponding audio and timing information. To improve performance and avoid repeated synthesis of identical speech, IntelliClass AI uses SHA-256-based caching, where the text and relevant synthesis parameters are used to generate a unique identifier for each audio segment; if the same content is requested again, the previously generated audio and alignment data can be reused instead of performing TTS computation again. Along with the synthesized audio, the system obtains phoneme or character-level timing information required for avatar animation. In the HeadTTS pipeline, phoneme-level information is directly available, while alignment information from other TTS providers can be processed into phoneme representations. These phonemes are then mapped to the 15 Oculus viseme categories, such as **sil, PP, FF, TH, DD, kk, CH, SS, nn, RR, aa, E, ih, oh, and ou**. Each viseme is associated with its corresponding start time and duration, allowing the 3D avatar to select the appropriate mouth shape during speech playback. The resulting audio and viseme timeline form a synchronized representation that is sent to the classroom clients, enabling the virtual lecturer's mouth movements to follow the generated speech in real time. This combination of preprocessing, caching, phoneme conversion, and viseme timing improves system efficiency while providing consistent and synchronized lip animation across the virtual classroom.

3.8 3D Avatar Rendering

The **3D Avatar Rendering** module provides the visual representation of the virtual lecturer in IntelliClass AI. The classroom environment and avatar are rendered locally in the student's browser using **Three.js and React Three Fiber**, allowing the system to display the lecturer without continuously streaming a pre-rendered video. The avatar receives the synchronized speech and viseme timeline generated by the TTS processing layer and changes its mouth shape according to the active viseme at the corresponding playback time. This creates synchronized lip movements that follow the generated speech. The avatar can also participate in the overall classroom presentation by displaying appropriate facial and animation states during lecture playback, pauses, and interactions. Since rendering is performed on the client side, each connected student independently generates the 3D scene while receiving only the required classroom events and synchronization information, reducing the computational and network requirements compared with transmitting a complete video stream of the virtual lecturer.

3.9 Conductor-Based Classroom Synchronization and Clock Synchronization

IntelliClass AI uses a **conductor-based synchronization architecture** to coordinate the activities of the virtual classroom across the lecturer and multiple student clients. The lecturer browser acts as the **classroom conductor**, controlling the sequence of lecture events and broadcasting information such as speech segments, subtitles, media instructions, viseme data, sequence numbers, and target playback timestamps through **Supabase Realtime Broadcast**. Each event contains timing information that enables student clients to determine when the corresponding audio and animation should be played. To reduce differences between individual client clocks, the system uses clock synchronization information to estimate the timing relationship between the lecturer and student browsers. Instead of depending only on when a network message is received, clients use the provided target timestamp to schedule playback according to the synchronized timeline. This approach helps maintain consistent timing between **audio, subtitles, avatar lip movements, images, videos, pauses, and classroom interactions**, even when small network delays occur. As a result, multiple students can experience the same lecture progression while rendering the 3D classroom locally on their own devices.

3.10 Lookahead TTS Prefetching

To provide continuous and responsive speech, IntelliClass AI uses a **lookahead TTS prefetching mechanism** during lecture playback. Instead of waiting for the current speech segment to finish before generating the next segment, the system prepares upcoming speech chunks in advance while the current chunk is being played. Multiple future chunks can be placed in a playback queue and synthesized or retrieved from the cache before they are required. Since the measured HeadTTS pipeline operates faster than real-time, this approach allows speech generation to stay ahead of playback and significantly reduces gaps between consecutive sentences. The mechanism also works together with the SHA-256-based audio cache, allowing previously generated segments to be reused immediately. This improves the overall responsiveness of the virtual lecturer and supports smooth delivery of long AI-generated lectures.

3.11 Student Question and AI Interruption

IntelliClass AI also supports **interactive student participation** during an ongoing lecture. When a student raises a question or initiates an interaction, the classroom controller can interrupt the current lecture at an appropriate speech boundary instead of waiting for the entire generated lecture to finish. The current playback state is controlled by the classroom conductor, allowing the avatar, audio, subtitles, and related classroom events to be paused consistently. The student's question can then be processed by the AI system using the available course knowledge and relevant retrieved context to generate a response. After the interaction is completed, the classroom can resume the lecture from the appropriate point in the sequence. This mechanism enables IntelliClass AI to behave more like an interactive lecturer rather than a fixed prerecorded presentation, allowing students to participate and receive AI-generated explanations during the lecture.

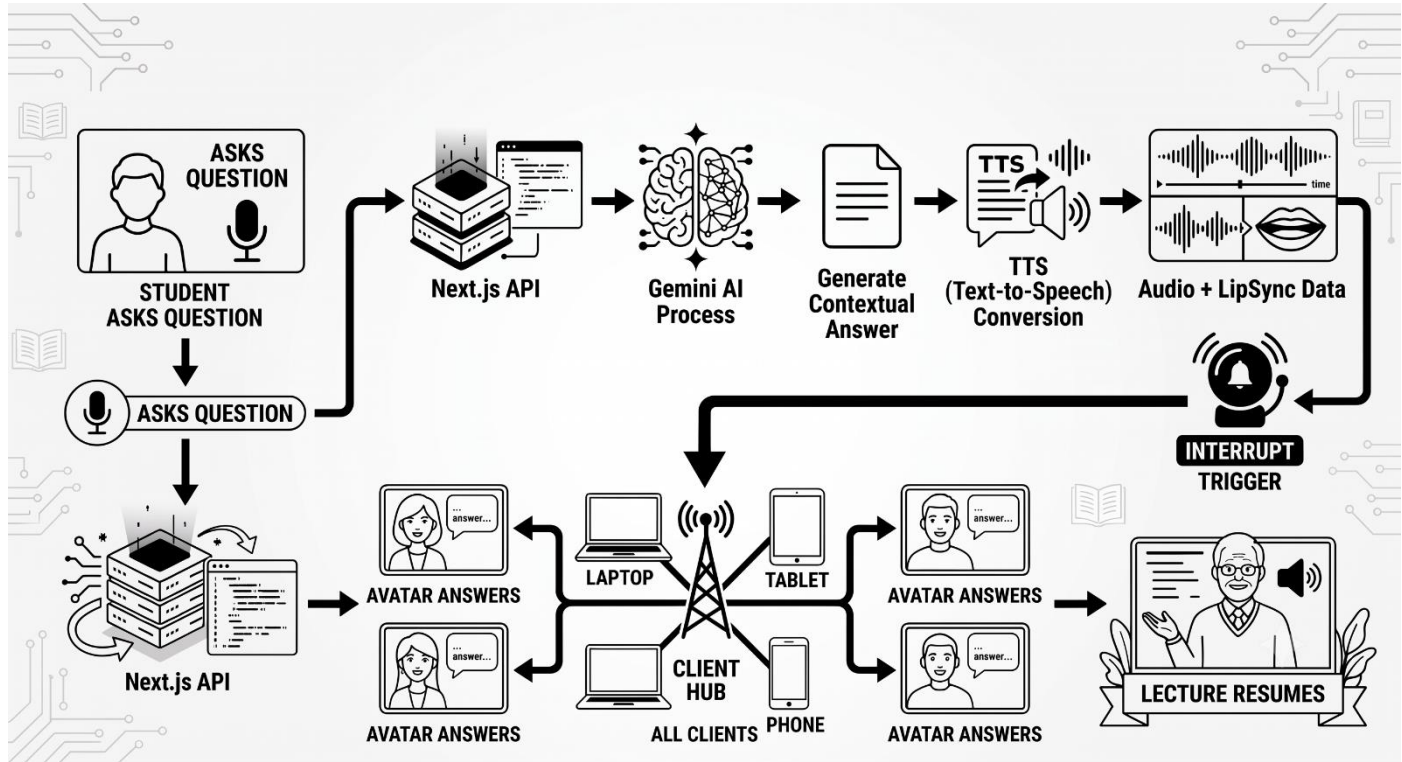


figure 3.11: student question and ai interruption

IV. RESULTS AND DISCUSSION

4.1 Experimental Setup

The quantitative benchmark experiments were performed on **September 6, 2026** using live benchmark scripts against running IntelliClass AI development services. The HeadTTS experiment used:

```
POST /v1/synthesize
localhost:8882
```

Three speech samples of increasing length were evaluated after a mandatory warm-up request. Synthesis time was measured using performance.now(). Audio duration was derived from the returned WAV payload.

4.2 HeadTTS Performance

table 4.1: headtts performance benchmarks

Sample	Words	Synthesis Time	Audio Duration	RTF	Visemes	Accuracy
Short greeting	10	1,118.9 ms	4.80 s	0.233	45	91.1%
Medium explanation	25	2,321.7 ms	9.43 s	0.246	91	83.5%
Long paragraph	50	5,423.7 ms	19.93 s	0.272	217	81.1%
Average	—	2,954.8 ms	—	0.250	—	85.2%

The average RTF was:

$$\text{RTF} = 0.250$$

$$\text{RTF} = 0.250$$

Since:

$$\text{RTF} < 1 \text{RTF} < 1$$

the speech synthesis process is faster than real-time. An RTF of 0.250 means that the system requires approximately one-quarter of the generated audio duration to synthesize the speech under the tested conditions.

4.3 Viseme Accuracy

The strict average direct viseme accuracy was calculated as:

$$(91.1\% + 83.5\% + 81.1\%) / 3 = 85.2\%$$

The strict metric counts phonemes that directly correspond to the selected Oculus viseme labels. Some HeadTTS phonemes are represented using IPA symbols, for example: ϵ , η , $\text{d}\zeta$.

These do not necessarily have a one-to-one string match with the 15 selected viseme labels.

Therefore, **85.2% should be reported as strict direct-label accuracy**, while the frontend's IPA-to-viseme lookup provides functional mapping for supported phonemes.

4.4 Audio Bandwidth

The current benchmarked implementation generates uncompressed mono PCM WAV audio.

Given:

SampleRate=24,000 Hz
BitDepth=16
Channels = 1

the theoretical bitrate is:

$$24,000 \times 16 \times 1 = 384,000 \text{ bits/sec}$$

Therefore:

$$384,000 \text{ bits/sec} = 384 \text{ kbps}$$

Table 4.4: audio bandwidth

Parameter	Measured/Calculated Value
Sample rate	24,000 Hz
Bit depth	16-bit
Channels	Mono
PCM bitrate	384 kbps
Measured bitrate	384.0–384.1 kbps
Estimated HTTP overhead	~7%
Audio + overhead	~410.9 kbps
Viseme metadata	~2.6 kbps
Approximate total	~413.4 kbps

Compared with a 3,250 kbps 1080p video baseline:

$$\text{Reduction} = (1 - (413.4 / 3250)) \times 100 \approx 87.3\%$$

Thus, under the specified comparison assumptions, the current architecture provides approximately **87.3% lower downstream bandwidth** than the selected 1080p video baseline.

V CONCLUSION

IntelliClass AI presents a unified intelligent virtual teaching framework that combines multimodal Retrieval-Augmented Generation (RAG), Gemini-based generative AI, neural Text-to-Speech, phoneme-to-viseme conversion, real-time 3D avatar rendering, and classroom synchronization. The system transforms instructor-provided PDF materials into a searchable course knowledge base containing textual and visual information, retrieves relevant content for generating course-grounded lectures, and converts the generated script into structured speech and classroom events. Speech chunking, lookahead TTS processing, and SHA-256-based caching improve lecture responsiveness, while phoneme-to-viseme mapping enables synchronized avatar lip movements. Three.js and React Three Fiber provide local 3D classroom rendering, and Supabase Realtime with a conductor-based architecture synchronizes audio, subtitles, avatar animation, multimedia, pauses, and student interactions across connected clients. The system therefore extends AI education beyond text-based question answering toward an interactive virtual lecturer capable of delivering structured and multimedia-supported lectures.

The experimental results demonstrate the feasibility of the implemented system under the tested conditions, with HeadTTS achieving an average synthesis time of 2,954.8 ms, an average Real-Time Factor of 0.250, and 85.2% strict direct viseme-label accuracy, while the uncompressed PCM audio stream required approximately 384 kbps. Gemini generation tests produced successful response latencies between 3,157 ms and 10,336 ms under the tested API conditions. Although further evaluation is required for RAG recall, human-perceived speech quality, and large-scale end-to-end classroom performance, the current results establish a practical foundation for an AI-powered virtual classroom. Future improvements can focus on compressed audio streaming, reduced AI generation latency, improved multimodal retrieval, enhanced viseme mapping, and larger-scale user evaluation, further strengthening IntelliClass AI as a scalable, interactive, and engaging platform for AI-assisted education.

Acknowledgment

The authors would like to express their sincere gratitude to Prof. Ramesh B E, Computer Science and Engineering, and SJM Institute of Technology for their valuable guidance, encouragement, and support throughout the development of the IntelliClass AI project. The authors also acknowledge the open-source communities and technologies that contributed to the development of the proposed system.

REFERENCES

- [1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). "Attention Is All You Need." *Advances in Neural Information Processing Systems (NeurIPS)*, 30.
- [2] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." *Advances in Neural Information Processing Systems (NeurIPS)*, 33.

- [3] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." *Proceedings of NAACL-HLT 2019*.
- [4] Reimers, N., & Gurevych, I. (2019). "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks." *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- [5] Johnson, J., Douze, M., & Jégou, H. (2019). "Billion-Scale Similarity Search with GPUs." *IEEE Transactions on Big Data*, 7(3), 535-547.
- [6] van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016). "WaveNet: A Generative Model for Raw Audio." *arXiv preprint arXiv:1609.03499*.
- [7] Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerry-Ryan, R. J., Saurous, R. A., Agiomyriannakis, Y., & Wu, Y. (2018). "Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions." *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
- [8] Ren, Y., Ruan, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T. Y. (2019). "FastSpeech: Fast, Robust and Controllable Text to Speech." *Advances in Neural Information Processing Systems (NeurIPS)*, 32.
- [9] Suwajanakorn, S., Seitz, S. M., & Kemelmacher-Shlizerman, I. (2017). "Synthesizing Obama: Learning Lip Sync from Audio." *ACM Transactions on Graphics (TOG)*, 36(4), 1-13.
- [10] Thies, J., Elgharib, M., Tewari, A., Theobalt, C., & Nießner, M. (2020). "Neural Voice Puppetry: Audio-driven Facial Reenactment." *European Conference on Computer Vision (ECCV)*.
- [11] Karras, T., Aila, T., Laine, S., Herva, A., & Lehtinen, J. (2017). "Audio-Driven Facial Animation by Joint End-to-End Learning of Pose and Emotion." *ACM Transactions on Graphics (TOG)*, 36(4), 1-12.
- [12] Ekman, P., & Friesen, W. V. (1978). "Facial Action Coding System: A Technique for the Measurement of Facial Movement." *Consulting Psychologists Press, Palo Alto, CA*.
- [13] VanLehn, K. (2011). "The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems." *Educational Psychologist*, 46(4), 197-221.
- [14] Woolf, B. P. (2010). *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing E-Learning*. Morgan Kaufmann Publishers.
- [15] Card, S., Cheng, L., Klemmer, S. R., & Fox, A. (2011). "PDF Document Analysis and Text Extraction Techniques for Structured Content Retrieval." *ACM Symposium on Document Engineering (DocEng)*.

Copyright & License:

© Authors retain the copyright of this article. This work is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.