iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.81578

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

AI-Based Adaptive Notes Generator from Lectures Audio and Video

Vericharla Ratnasri[1], Sanchula Siva Jyothi[2], Valluri Radhika[3], Murugula Rajeswari[4]
[1]*Assistant Professor Department of Computer Science and Engineering, Bapatla Women's Engineering College, Bapatla, India*
[2, 3, 4]*Department of Computer Science and Engineering, Bapatla Women's Engineering College, Bapatla, India*

Abstract: ***Efficient lecture understanding and personalized learning support are essential for improving student academic performance in modern digital education environments. Traditional note-taking methods are often inconsistent, incomplete, and ineffective in capturing key concepts from lectures, especially when delivered through audio or video formats. Additionally, existing summarization tools fail to adapt to different student learning levels and do not identify individual knowledge gaps. These limitations can result in poor concept clarity, reduced retention, and lack of personalized learning guidance. To address these challenges, this paper proposes an AI-Based Adaptive Lecture Notes Generator with Knowledge Gap Detection and Multi-Level Learning Modes, a web-based system designed to convert lecture audio and video into structured, personalized study material. The system integrates speech recognition techniques to transcribe lecture content, followed by Natural Language Processing (NLP) methods to generate organized notes. It introduces a multi-level learning framework that provides Beginner, Intermediate, and Advanced modes, enabling students to access content based on their level of understanding. A knowledge gap detection mechanism is implemented using automated quiz generation and semantic similarity-based evaluation to analyze student responses and identify weak concepts. Based on performance, the system adaptively regenerates simplified explanations to enhance understanding. The system is developed using Python and Flask for backend processing with a responsive frontend interface. Experimental evaluation demonstrates improved note quality, enhanced learning personalization, better concept clarity, and increased student engagement.***

Keywords: ***Adaptive Learning System, Lecture Notes Generation, Knowledge Gap Detection, Natural Language Processing (NLP), Speech Recognition, Semantic Similarity, Educational AI, Web-Based Learning System.***

I. INTRODUCTION

Educational institutions operate within a structured learning environment where effective understanding and retention of lecture content play a crucial role in improving academic performance. Lecture processing is not merely the conversion of audio into text. It involves careful analysis of lecture content, identification of key concepts, structuring of information, and adaptation based on different learning levels. As the volume of digital lectures increases through online platforms, classrooms, and recorded sessions, the complexity of extracting meaningful and personalized study material also grows significantly. Managing these processes manually often leads to inefficiencies and challenges. In many educational settings, note-taking is still performed manually or through basic transcription tools.

These traditional methods are time-consuming and highly dependent on student attention and comprehension during lectures. Even minor gaps, such as missed explanations or misunderstood concepts, may result in incomplete notes. This often leads to poor concept clarity, inconsistent learning, and difficulty in revision. Over time, such inefficiencies can negatively impact student performance and confidence. Another major challenge in modern learning is the inability to identify and address individual knowledge gaps. Generic notes do not adapt to different student learning levels, leading to either oversimplified or overly complex content. Additionally, the absence of real-time feedback restricts students from understanding their weak areas and improving effectively. Educational systems require an intelligent, adaptive solution that ensures clarity, personalization, and continuous learning support while reducing manual effort. To address these challenges, this project introduces an AI-Based Adaptive Lecture Notes Generator with Knowledge Gap Detection and Multi-Level Learning Modes. The system is designed as a web-based solution that automates lecture processing and note generation by applying advanced speech recognition and Natural Language Processing techniques.

The system incorporates an intelligent processing pipeline that evaluates multiple constraints such as lecture transcription, content structuring, multi-level note generation, and knowledge gap detection to generate accurate and personalized learning material. The system incorporates a multi-level learning framework that supports three primary modes: Beginner, Intermediate, and Advanced.

The Beginner mode provides simplified explanations, the Intermediate mode offers balanced content, and the Advanced mode delivers detailed conceptual understanding.

Students can upload lecture audio or video content and access structured notes, while also attempting automatically generated quizzes to evaluate their understanding. The system analyzes student responses using semantic similarity techniques to identify weak concepts. This structure ensures adaptive and personalized learning. Built with Python and Flask for backend processing, along with a responsive frontend interface, the system follows a scalable architecture. By automating lecture understanding and supporting adaptive feedback, it reduces manual effort, improves clarity, and enhances learning outcomes.

II. LITERATURE SURVEY

Chen et al. developed an automatic lecture summarization system using deep learning-based Natural Language Processing techniques. Their model effectively extracted key sentences from lecture transcripts to generate concise summaries. However, the system produced only a single-level summary and did not support adaptive learning for different student levels. In contrast, the proposed system generates multi-level notes (Beginner, Intermediate, Advanced) to support personalized learning. [1]

Khan et al. proposed a speech-to-text-based lecture transcription system using advanced speech recognition models. The system achieved high accuracy in converting lecture audio into text. However, it lacked structured note generation and learning assessment features. In contrast, the proposed system integrates transcription with structured notes and knowledge gap detection. [2]

Sharma et al. implemented an AI-based educational assistant that generates summaries from recorded lectures using NLP techniques. The system improved content accessibility but did not evaluate student understanding or provide feedback. In contrast, the proposed system includes quiz generation and semantic evaluation to identify knowledge gaps. [3]

Patel et al. developed a text summarization tool for academic content using extractive summarization methods. The system efficiently generated concise summaries but did not adapt content based on user knowledge levels. In contrast, the proposed system introduces multi-level learning modes for personalized content. [4]

Lee et al. proposed a lecture note generation system using transformer-based models. Their system improved contextual understanding but required high computational resources and lacked adaptive feedback. In contrast, the proposed system uses an architecture with adaptive learning support. [5]

Singh et al. developed an intelligent tutoring system that provided personalized learning recommendations based on student performance. The system effectively identified weak areas but did not integrate lecture-based note generation. In contrast, the proposed system combines note generation with knowledge gap detection in a unified framework.[6]

Zhang et al. implemented a deep learning-based speech recognition system for educational applications. The system provided accurate transcription of lecture audio. However, it did not perform content structuring or generate meaningful study material. In contrast, the proposed system converts transcribed text into structured and organized notes. [7]

Verma et al. designed a web-based learning system that provided static study materials and quizzes. While the system supported online learning, it lacked adaptive content generation based on student performance. In contrast, the proposed system dynamically adjusts content using performance-based feedback. [8]

Gupta et al. developed an automatic question generation system using NLP techniques. The system generated objective questions from text content effectively. However, it did not evaluate answers using semantic similarity or provide adaptive feedback. In contrast, the proposed system evaluates responses meaningfully and identifies knowledge gaps. [9]

Reddy et al. proposed an e-learning platform with basic content recommendation features. The system suggested learning materials based on predefined rules. However, it lacked real-time adaptation and personalized note generation. In contrast, the proposed system provides dynamic and personalized learning content. [10]

Li et al. introduced a semantic similarity model for evaluating student answers using NLP techniques. The system improved answer evaluation accuracy compared to traditional keyword-based methods by considering contextual meaning. It was effective in identifying partially correct responses and reducing evaluation bias. However, it was not integrated into a complete learning system. In contrast, the proposed system embeds semantic evaluation within an adaptive learning pipeline. [11]

Kumar et al. developed a lecture recording and playback system for educational institutions. The system improved accessibility to lecture content by allowing students to revisit recorded sessions at any time. It supported flexible learning but did not process or structure the content for better understanding. Additionally, it lacked mechanisms for summarization and knowledge assessment. In contrast, the proposed system transforms lecture recordings into structured notes and learning material. [12]

Ahmed et al. implemented a chatbot-based educational assistant to answer student queries. The system provided interactive responses and improved engagement through conversational interfaces. However, many responses were generic and lacked

structured explanation of concepts. It also did not evaluate student understanding or identify weak areas. In contrast, the proposed system delivers structured notes and targeted explanations based on knowledge gaps. [13]

Saha et al. proposed machine learning-based recommendation system for online courses. The system suggested courses based on user preferences and learning behavior, improving user engagement and course selection. However, it did not focus on lecture understanding, content structuring, or note generation from lecture data. In contrast, the proposed system focuses on lecture-based learning enhancement by generating structured notes and adaptive learning content. [14]

Das et al. developed hybrid NLP model for text classification and summarization in educational datasets. The system improved content categorization and summary generation. However, it lacked adaptive learning modes and performance-based content modification. In contrast, the proposed system integrates multi-level learning and adaptive feedback mechanisms. [15]

III. METHODOLOGY

The AI-Based Adaptive Lecture Notes Generator with Knowledge Gap Detection and Multi-Level Learning Modes is designed as a modular intelligent framework for lecture processing and personalized learning. It integrates speech recognition, Natural Language Processing, multi-level note generation, knowledge gap detection, quiz evaluation, and adaptive feedback within a web-based platform. The system processes lecture audio and video to generate structured notes and evaluate student understanding. Using Python and Flask for backend processing, along with a responsive frontend interface, the system provides a scalable and efficient learning support solution.

A. Lecture Input and Processing

The system begins with lecture input in the form of audio or video files provided by the user. The input is processed using speech recognition techniques to convert spoken content into text format. The generated transcript is then cleaned by removing noise, filler words, and irrelevant content. This ensures that the extracted text is meaningful and suitable for further processing. The structured transcript forms the foundation for note generation and analysis. Additionally, preprocessing improves transcription accuracy and prepares the data for efficient downstream processing. It also ensures consistency and reliability in handling different types of lecture inputs.

B. Natural Language Processing and Content Structuring

The processed text is analyzed using Natural Language Processing techniques to identify key concepts, keywords, and important sentences. Tokenization, filtering, and content segmentation are applied to organize the lecture content logically. The system extracts meaningful information and removes redundant data. This step ensures that the content is structured in a clear and readable format. It improves the quality and coherence of generated notes.

C. Multi-Level Notes Generation

The system generates notes in three different levels: Beginner, Intermediate, and Advanced. The Beginner mode provides simplified explanations for easy understanding, the Intermediate mode offers balanced and structured content, and the Advanced mode delivers detailed and in-depth conceptual information. The generation process adapts the complexity of content based on predefined rules. This ensures that students can access notes according to their level of understanding. It enhances personalized learning and knowledge accessibility. Additionally, this approach accommodates diverse learning capabilities and supports progressive knowledge building. It also enables students to transition between levels as their understanding improves.

D. Automated Quiz Generation and Evaluation

The system automatically generates quizzes from the processed lecture content to assess student understanding. Questions are created based on key concepts and important topics extracted from the lecture. Students can attempt these quizzes through the interface. The system evaluates responses using semantic similarity techniques instead of exact keyword matching. This ensures meaningful evaluation of answers. It provides accurate assessment of student understanding.

E. Knowledge Gap Detection and Adaptive Feedback

Based on quiz performance, the system identifies weak concepts and knowledge gaps in student understanding. It analyzes incorrect or partially correct responses to determine areas of difficulty. The system then regenerates simplified explanations or additional notes for those concepts.

This adaptive feedback mechanism ensures targeted learning improvement. It supports continuous learning by addressing individual weaknesses. Additionally, the system dynamically adjusts content difficulty to match learner progress. This enhances retention and promotes a more effective and personalized learning experience.

F. Structured Notes Delivery and User Interaction

The generated notes and feedback are delivered through a user-friendly interface. Students can select their preferred learning mode and access structured notes instantly. The system ensures that all updates, including regenerated content and feedback, are reflected in real time. The interface supports smooth navigation and organized presentation of content. This module enhances user experience and ensures effective learning support.

IV. SYSTEM IMPLEMENTATION

The proposed AI-Based Adaptive Lecture Notes Generator with Knowledge Gap Detection and Multi-Level Learning Modes is implemented as a modular web-based framework integrating speech recognition, Natural Language Processing, multi-level note generation, and adaptive learning mechanisms. Built using Python and Flask with a responsive frontend, the system generates structured notes and evaluates student understanding. The workflow ensures efficient content processing and personalized learning support.

G. Lecture Input Module

The module handles lecture input verification for audio and video files uploaded by users. It validates file format and quality before processing through the Flask backend. Upon successful validation, the lecture content is forwarded for speech-to-text conversion and further analysis.

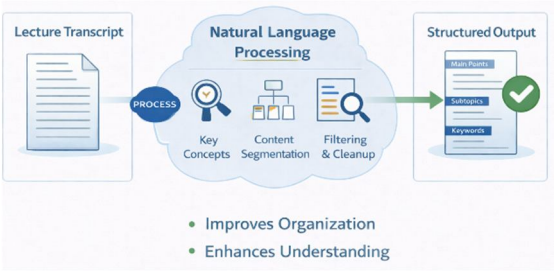


Fig. 1. Lecture Input workflow

H. Speech Recognition Module

The Speech Recognition Module converts lecture audio into text using speech-to-text techniques. It processes spoken content and generates a transcript for further analysis. The module handles variations in speech and removes unnecessary noise to improve accuracy. This module ensures reliable text extraction for downstream processing. The interaction and workflow are depicted in Figure 2.

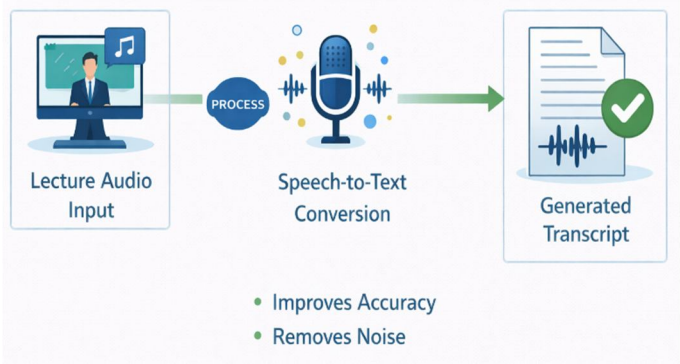


Fig. 2. Speech Recognition Module

I. Natural Language Processing Module

The Natural Language Processing Module provides centralized processing of lecture transcripts to extract meaningful information. It enables the system to identify key concepts, important sentences, and relevant keywords from the converted text. The module also supports content segmentation, filtering, and removal of redundant data to ensure structured and readable output. All processed data is prepared for multi-level note generation and further analysis.

This module also enhances content clarity and organization by refining the extracted information. It ensures that the structured text is logically arranged for better understanding and efficient processing in subsequent stages. These processing steps significantly improve content quality and reliability as shown in Figure 3.

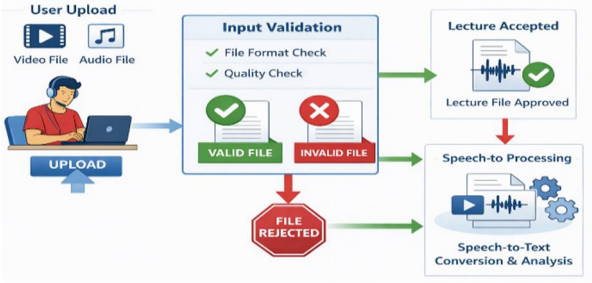


Fig. 3. NLP Module

J. Multi-Level Notes Generation Module

The Faculty Management Module allows faculty members to securely log in and access their personalized dashboard. Through this module, they can view assigned subjects, monitor teaching workload, and access month-wise timetables. The system retrieves schedule data from MongoDB and displays it dynamically using ReactJS, ensuring real-time visibility. The module also enables faculty to submit leave requests and track their approval status. Approved changes are reflected instantly in their timetable, ensuring clear communication and coordinated academic management, as depicted in Figure 4.

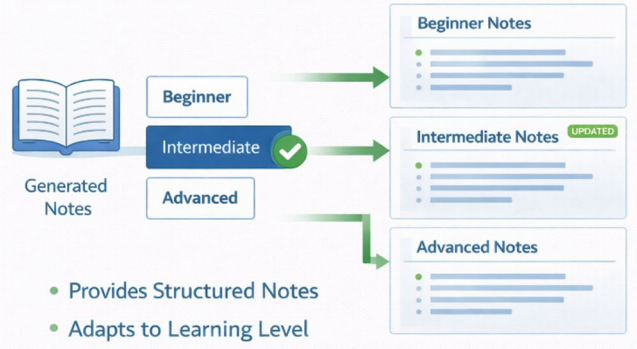


Fig. 4. Multi-Level Notes Generation Module workflow

K. Quiz Generation Module

The Quiz Generation Module provides structured assessment support by generating questions from lecture content. Students can access quizzes through their personalized interface and attempt them to evaluate their understanding. The system generates questions based on key concepts extracted from processed data. It ensures that quizzes are relevant and aligned with the lecture material. The module supports accurate and consistent assessment, enabling effective learning evaluation.

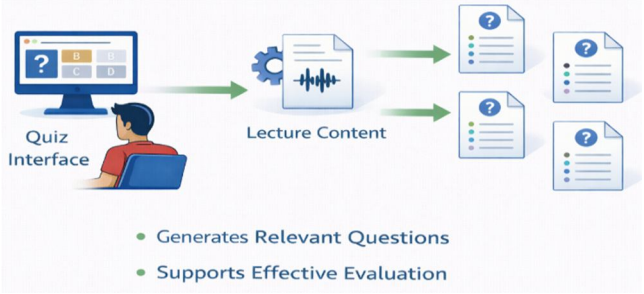


Fig. 5. Quiz Generation module

L. Semantic Evaluation Module

The Semantic Evaluation Module functions as the core assessment engine of the system. It evaluates student responses by analyzing semantic similarity between expected and submitted answers. The module considers contextual meaning instead of exact keyword matching to ensure accurate evaluation. It identifies correct, partially correct, and incorrect answers based on conceptual understanding. The evaluated results are stored and reflected in the system, ensuring reliable and meaningful assessment of student performance.

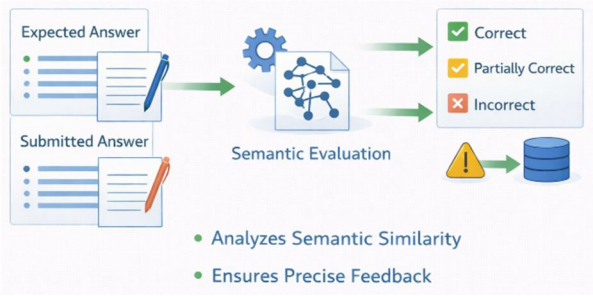


Fig. 6. Semantic Evaluation module

M. Knowledge Gap Detection Module

The Knowledge Gap Detection Module tracks student performance across quiz attempts to identify weak concepts. Before providing feedback, the system analyzes incorrect and partially correct responses to detect gaps in understanding. If gaps are identified, the system focuses on those topics for improvement. The module provides structured insights for targeted learning and content refinement. This mechanism ensures personalized learning support and improved concept clarity.

N. Adaptive Feedback and User Interface Module

The Adaptive Feedback Module enables the system to provide personalized learning support based on student performance.

V. RESULTS AND DISCUSSION

The developed AI-Based Adaptive Lecture Notes Generator with Knowledge Gap Detection and Multi-Level Learning Modes was evaluated using lecture audio and video datasets. The workflow, including speech recognition, note generation, quiz evaluation, and adaptive feedback, was tested across all modules. Performance was assessed based on note accuracy, content quality, knowledge gap detection efficiency, and system responsiveness.

A. User Interface

The interface combines the user dashboard and Adaptive Notes Studio of the AI-Based Adaptive Lecture Notes Generator system. The top section displays user details and system status, along with options such as History and Logout. The central section highlights key functionalities, including lecture upload, speech-to-text conversion, and structured note generation. The lower section provides three learning modes—Beginner, Intermediate, and Advanced—to support personalized learning. The clean and responsive design ensures secure access, real-time interaction, and an improved user experience, as shown in Figure 7.

.

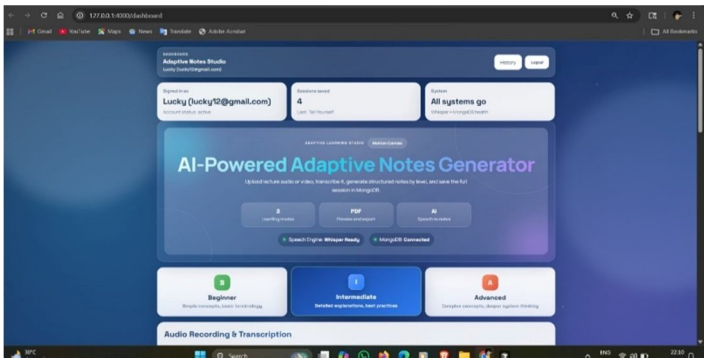


Fig. 7. User Interface.

B. Create Account

The Knowledge Gap Detection Module tracks student performance across quiz attempts to identify weak concepts. Before providing feedback, the system analyzes incorrect and partially correct responses to detect gaps in understanding. If gaps are identified, the system focuses on those topics for improvement. The module provides structured insights for targeted learning and content refinement. This mechanism ensures personalized learning support and improved concept clarity.

This interface represents the user registration and login section of the AI-Based Adaptive Lecture Notes Generator system. It allows users to create an account using basic details such as name, email, and password, ensuring secure access. The interface also provides login options for existing users to access their dashboard. This module supports secure authentication and smooth user onboarding.

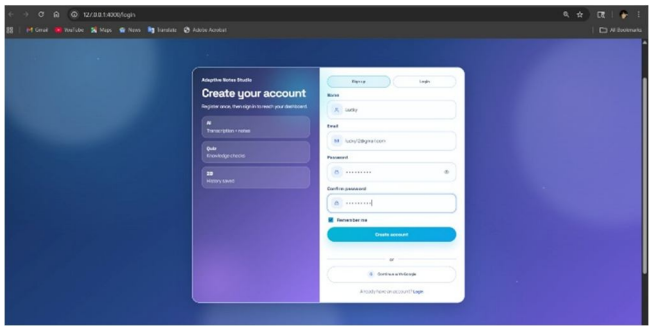


Fig. 8. Account Creation.

C. Generated Output Interface

The images show the Generated Output and Notes Preview modules of the AI-Based Adaptive Lecture Notes Generator system. The Generated Output section displays structured lecture content, including topic, transcription, introduction, summary, key highlights, and learning structure. The Notes Preview provides a detailed view of the generated notes in a clean and readable format. Users can review, organize, and export notes through available options such as PDF and DOCX. These features ensure structured content presentation, easy revision, and efficient learning support.

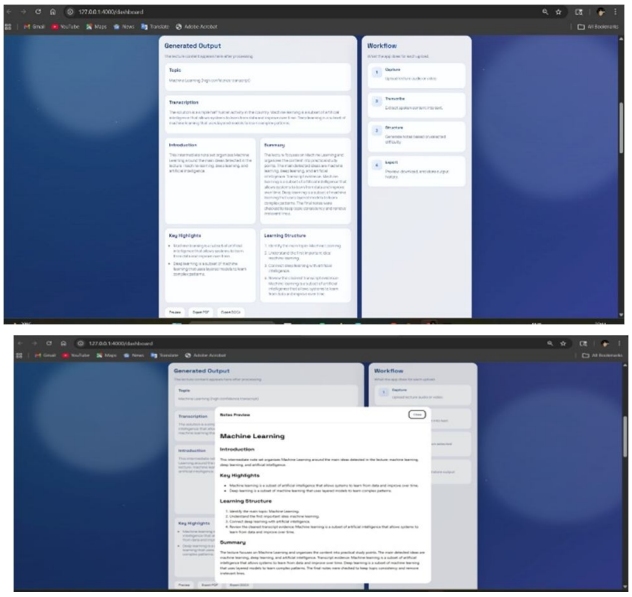


Fig. 9. Output Interface.

VI. CONCLUSION

This paper presented the design and implementation of an AI-Based Adaptive Lecture Notes Generator with Knowledge Gap Detection and Multi-Level Learning Modes to enhance lecture understanding and personalized learning. The system integrates speech recognition, Natural Language Processing, multi-level note generation, quiz evaluation, and adaptive feedback to provide structured and meaningful study material.

By processing lecture audio and video, the system generates organized notes and evaluates student understanding through semantic analysis, ensuring accurate knowledge assessment and improvement. Developed using Python and Flask for backend processing, along with a responsive frontend interface, the system follows a modular architecture that supports scalability and efficient learning support. Experimental evaluation demonstrated improved note quality, enhanced personalization, better concept clarity, and increased student engagement. Although large-scale implementation may require optimization techniques such as improved model efficiency and faster processing, the proposed system provides a practical and intelligent solution for modern digital education. Future enhancements will focus on integrating advanced AI models, real-time learning analytics, and multilingual support to further improve adaptability and learning effectiveness.

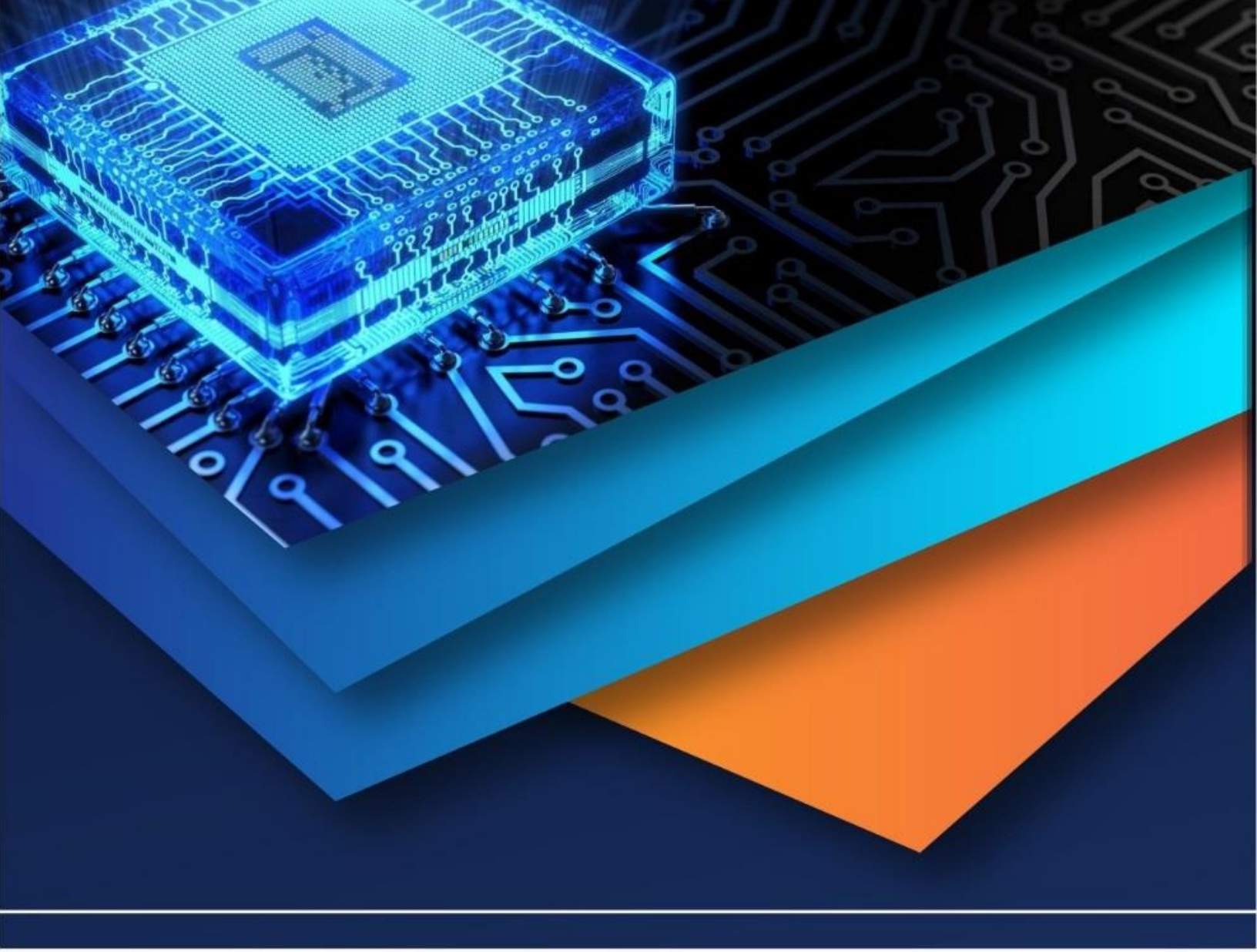
.

VII. ACKNOWLEDGMENTS

The authors would like to express their sincere gratitude to the faculty mentors and project guides for their valuable guidance and continuous support during the design and implementation of this project. Their suggestions were instrumental in refining the system approach and ensuring its effectiveness in enhancing learning outcomes. The authors also acknowledge the support of departmental staff and the use of open-source technologies such as Python and Flask, which significantly contributed to the successful development and evaluation of the proposed system.

REFERENCES

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” arXiv preprint arXiv:1301.3781, 2013.
[2] A. Vaswani et al., “Attention is all you need,” in Advances in Neural Information Processing Systems, vol. 30, pp. 5998–6008, 2017.
[3] D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2021.
[4] K. Hinton et al., “Deep neural networks for acoustic modeling in speech recognition,” IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82–97, 2012.
[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. NAACL-HLT, pp. 4171–4186, 2019.
[6] R. Nallapati, B. Zhou, C. dos Santos, Ç. Gülçehre, and B. Xiang, “Abstractive text summarization using sequence-to-sequence RNNs,” arXiv preprint arXiv:1602.06023, 2016.
[7] P. Liu and X. Qiu, “Text summarization with pretrained encoders,” IEEE Transactions on Knowledge and Data Engineering, vol. 33, no. 6, pp. 2474–2486, 2021.
[8] T. Brown et al., “Language models are few-shot learners,” in Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901, 2020.
[9] J. Pennington, R. Socher, and C. Manning, “GloVe: Global vectors for word representation,” in Proc. EMNLP, pp. 1532–1543, 2014.
[10] Y. Goldberg, Neural Network Methods in Natural Language Processing. San Rafael, CA, USA: Morgan & Claypool, 2017.
[11] M. Mohri, A. Rostamizadeh, and A. Talwalkar, Foundations of Machine Learning, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.
[12] A. Graves, “Speech recognition with deep recurrent neural networks,” in Proc. IEEE ICASSP, pp. 6645–6649, 2013.
[13] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[14] D. Blei, A. Ng, and M. Jordan, “Latent Dirichlet allocation,” Journal of Machine Learning Research, vol. 3, pp. 993–1022, 2003.
[15] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge University Press, 2008.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY