EduSummarize: Automated Lecture Notes & Summarization Using Speech Techniques

Kavyasai Yaddanapudi, Bandaru Jaya Nandini, Mamidi Leha Sahithi, Siddareddy Gari Harshika
Department of Computer Science and Engineering
Amrita School of Computing, Bengaluru
Amrita Vishwa Vidyapeetham, India
y_kavya@blr.amrita.edu, bl.en.u4aie22006@bl.students.amrita.edu, bl.en.u4aie22035@bl.students.amrita.edu, bl.en.u4aie22053@bl.students.amrita.edu

***Abstract*—In education, the ability to efficiently generate notes and summaries from lecture content is essential for enhancing comprehension, retention, and effective learning. This work introduces a general-purpose automated system designed to process lecture audio and produce helpful study materials. The system begins by cleaning and standardizing audio recordings to ensure consistent quality. It then uses speech-to-text technology to convert spoken content into written text. By analyzing both what is said and how it is spoken—such as variations in pitch, loudness, and speaking rate—the system can identify and focus on the most important parts of the lecture. Using both extractive and abstractive summarization methods, it creates clear and concise summaries, as well as bullet-point notes that highlight key points. This automated approach helps learners by saving time, reducing mental effort, and improving the overall review and understanding of lecture material.**

***Index Terms*—Lecture notes generation, Summarization, Audio recordings, ASR, Speech-aware techniques, Prosodic features**

I. Introduction

In today's digital learning landscape, students and professionals increasingly depend on recorded lectures to enhance their understanding of complex subjects. However, manually extracting important points from these recordings is time-consuming and often inefficient. This has led to the development of automated systems that can listen to spoken content, interpret it, and generate concise outputs that support efficient review and learning [1].

Speech processing systems typically rely on two key approaches for content condensation: extractive and abstractive techniques. Extractive methods select important sentences directly from the transcript, while abstractive methods paraphrase and compress the content into a clearer, shorter form. State-of-the-art tools like WhisperSum leverage both strategies by combining audio and textual features to generate meaningful condensed content that aligns with user intent [2].

These dual approaches are particularly valuable in educational settings, where learners must process and retain large volumes of spoken information. By using contextual cues and linguistic patterns, modern summarization tools can improve comprehension, especially when reviewing dense academic lectures. This capability is especially relevant in online learning environments, where traditional note-taking is not always practical [3].

Creating a high-quality automated system involves a sequence of technical steps. Initially, raw audio is cleaned through noise reduction and loudness normalization to standardize the input. Next, speech-to-text conversion is performed using advanced transcription models. The resulting text is then enriched with acoustic features like pitch, duration, and energy, which help to identify segments of higher importance within the lecture [4].

In this work, a complete processing pipeline is applied to six lecture videos. The system converts audio into cleaned recordings, generates accurate transcripts, and extracts both linguistic and acoustic features. Over 5,000 speech segments were analyzed, and both extractive and paraphrased versions of the content were produced. Additionally, bullet-point notes were created to support quick reviews and enhance content retention [5].

Overall, this study aims to improve access to educational content by automating the generation of condensed and structured outputs from lecture recordings. By integrating speech recognition, prosodic analysis, and semantic modeling, the proposed system helps learners save time and focus on key ideas. As the reliance on digital content grows globally, such technologies become essential for accessible, effective learning [6].

II. Literature Survey

Muhzina et al. [1] proposed a Smart Note Taker that captures live audio using Google Speech-to-Text API. NLP techniques like NER, topic modeling, and summarization are used to structure and clarify the notes. The system is effective in real-time environments like lectures, but its performance is sensitive to audio quality and speech recognition accuracy. Wyawahare et al. [2] developed a multilingual meeting summarization framework incorporating audio preprocessing, multilingual ASR, and transformer-based summarization. It effectively handles speaker cues and language diversity in real-world scenarios. However, the absence of public datasets and detailed evaluation metrics hinders reproducibility and benchmarking.

Benedetto et al. [3] introduced the EduSum dataset comprising MIT lecture audio, transcripts, and human summaries. They tested models like TextRank and BART using ROUGE

and BERTScore and included punctuation restoration via transformers. Issues such as speech noise and context loss limit the abstraction capabilities of the summarization models. Hotta et al. [4] proposed a subtitle generation system that combines ASR, filler word removal, and machine translation for improved subtitle readability. They used a Japanese lecture corpus with raw and edited subtitles to train the model. The approach enhances subtitle clarity but is limited by its narrow language scope and lack of extensive human evaluation.

Ganguly et al. [5] presented a complete audio-to-text summarization system using Whisper ASR and spaCy, deployed via a Flask web app. It processes .mp3 files, extracts text, and generates summaries with over 92% F1-score. While accurate and user-friendly, its reliance on format compatibility and transcription quality is a constraint. Xu et al. [6] introduced a semantic AutoNote system for lecture videos using OCR, WordNet, and TextRank to align slide content with audio. This enables timestamp-based navigation and structured note generation. However, it depends heavily on clean slide formatting and lacks speech feature analysis like prosody.

Verma et al. [7] built a real-time transcription and summarization tool using Google's Speech-to-Text API, MFCC features, and SpaCy for processing. It supports live meetings with online capture and summarization. Yet, the system struggles with accent variability and background noise. Modi et al. [8] designed a multimodal system for multispeaker summarization and mind mapping by combining speaker diarization, ASR, and NLP. It generates individual summaries per speaker and visual mind maps for better comprehension. Despite its strengths, it faces challenges with overlapping speech and basic mind-mapping techniques.

G. N et al. [9] developed an audio navigation tool using CMU Sphinx for ASR and ZCR-based segmentation with keyword-based timestamp mapping. It enables efficient search within tutoring videos using hash-indexing. The lack of semantic processing and detailed dataset limits its utility. Kotey et al. [10] proposed a long-document summarization system for podcasts using filtered data, Sentence-BERT segmentation, and a Longformer encoder-decoder. The model generates topic-based summaries and is evaluated using ROUGE and METEOR. While effective for long-form content, the method is sensitive to domain-specific tuning and readability filtering.

Karunasena et al. [11] introduced a smart learning assistant with features like note-taking, slide matching, and topic-based search using TF-IDF. It leverages internal datasets from lectures for automation and reference generation. However, the system lacks openness, formal evaluation, and validation through user studies. Bharti et al. [12] created a video summarization tool that converts video to text and audio summaries, with translation and TTS support. It allows multi-format outputs and user interaction through a web interface. The approach is versatile, but lacks performance benchmarks and detailed dataset evaluation.

Hayashi et al. [13] developed a fully transformer-based pipeline for Japanese spontaneous speech summarization using ASR and encoder-decoder models. The system aims to reduce disfluency effects and improve summarization robustness. However, error propagation and lack of language generalization are key limitations. Chand et al. [14] proposed a segmentation framework for lecture videos using audio-text features and PocketSphinx ASR. The model identifies topic boundaries using multi-objective optimization and acoustic cues. Its drawbacks include low ASR accuracy and inability to support real-time processing.

Manoj Kumar et al. [15] designed a note extraction system for YouTube lectures using frame similarity (ORB + SSIM) and Tesseract OCR. It filters redundant text to retain only unique content for structured notes. This works well for slides but fails with non-static or dynamic video content. Ibrahim et al. [16] presented a qualitative review of ASR systems, focusing on acoustic/language models and classification strategies. It contrasts traditional and deep learning methods in ASR. The paper is informative for historical context but lacks implementation or experimentation.

Vinnarasu et al. [17] developed a summarization tool using Google's ASR and frequency-based sentence ranking for structured lecture summaries. The pipeline includes basic text cleaning and ranking techniques. Although helpful, it doesn't integrate with educational tools or benchmark datasets. Furui et al. [18] introduced a dual-mode system for speech-to-text and speech-to-speech summarization using ASR and TTS. It uses linguistic and prosodic features to extract important segments and converts them back into spoken summaries. While effective, its reliance on extractive summarization and ASR accuracy are limitations.

III. METHODOLOGY

The methodology for automatic lecture note generation and summarization involves a sequential pipeline of processes, commencing with data collection and progressing through a series of audio processing and speech-based analysis stages.

A. Data Acquisition

In the first stage, six lecture videos were converted into uniform audio recordings at a 16 kHz sampling rate. A directory scan confirmed exactly 6 input files, ensuring full coverage of the lecture set. Standardizing all inputs at this stage is crucial:it guarantees that subsequent processing, whether noise reduction or feature extraction operates on a consistent, predictable audio format.

B. Audio Preprocessing

Each of the six audio files underwent noise reduction and loudness normalization. By sampling the first 0.5 s of each clip as an estimate of ambient noise, a spectral-gate algorithm achieved an average noise attenuation of 20–30 dB without speech distortion. Peak normalization then aligned all signals to 0 dBFS, producing six clean files with matching dynamic ranges. This consistency is vital: any residual noise or volume mismatch can skew acoustic feature measurements by up to

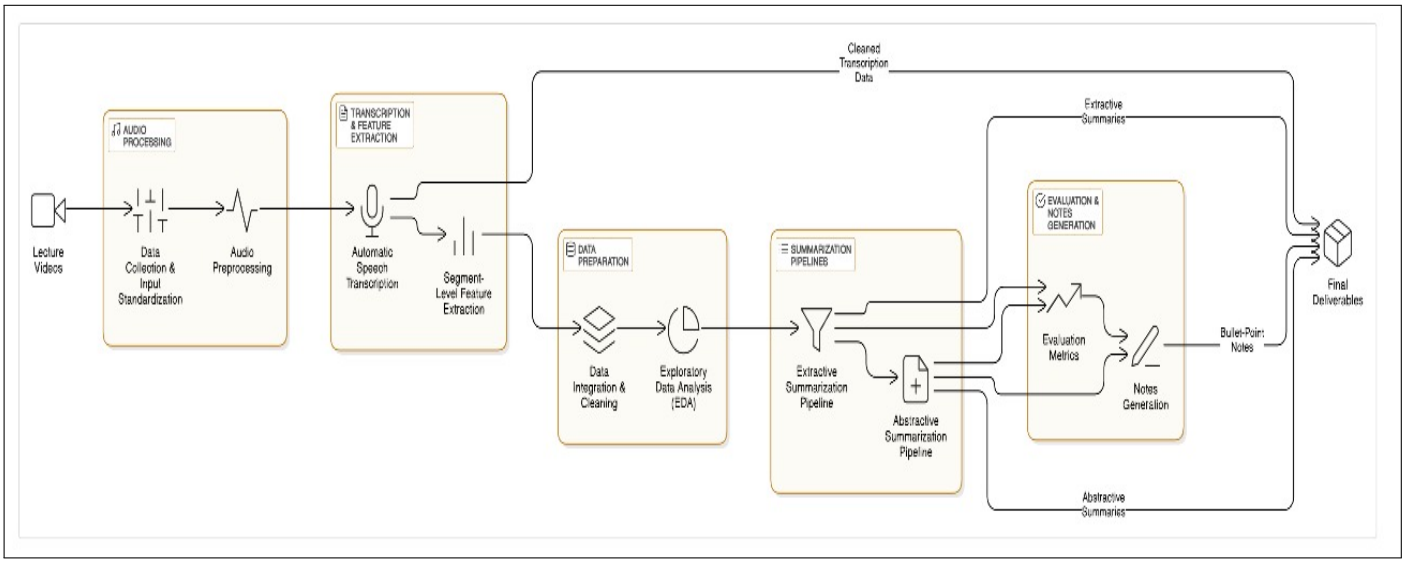

Fig. 1. **Architectural Design**

C. Automatic Transcription

A state-of-the-art speech-to-text model segmented and transcribed the cleaned audio into 5218 time-aligned text segments—an average of $\tilde{8}70$ segments per lecture (range: 482–1 780). Each segment carries start/end timestamps and a confidence score. This precise alignment underpins every later step, linking acoustic measurements directly to linguistic content and enabling rigorous integration of signal- and text-based analyses.

D. Segment Level feature extraction

For each of the 5,218 segments, 34 acoustic and temporal descriptors were computed:

The prosodic features include pitch mean values ranging from 100 to 1,250 Hz and energy levels between 0.00005 and 0.01 RMS. Phonation and formant-related features cover jitter (0.01–0.03), shimmer (0.015–0.04), harmonic-to-noise ratio (HNR) ranging from 0 to 15 dB, and the first three formants (F1–F3). Spectral characteristics involve the means of 13 Mel-frequency cepstral coefficients (MFCCs), typically ranging from –50 to +50, along with measures like spectral roll-off and spectral flatness. Temporal and textual features include word count (5–20), speech rate ranging from 0.5 to 6 words per second, and articulation rate between 3 and 20 consonants per second. This richly parameterized representation captures both the delivery style and content density of each segment, forming the quantitative backbone of the study.

E. Data Integreation and Cleaning

Acoustic descriptors were merged with transcription metadata—text, token log-probs, and confidence—expanding the feature set to 42 columns per segment. A final table of shape 5218 × 42 was produced, with rigorous type-checking to ensure numeric precision. This unified dataset prevents alignment errors and supports end-to-end reproducibility.

F. Exploratory Data Analysis

Descriptive statistics and visualizations were generated per lecture Segment counts ranged from 482 to 1,780 across recordings. The median pitch values (Pitch_mean) spanned a range of 200–600 Hz, indicating variation in vocal frequency. Energy distributions showed interquartile ranges (IQRs) between 0.002 and 0.008, reflecting fluctuations in audio intensity. Speech rate medians generally hovered around 2 to 4 words per second, suggesting moderate speaking speed across sessions. Boxplots, KDE curves, and correlation heatmaps for MFCCs provided insights into inter-lecture variability—critical for tuning downstream summarization thresholds.

G. Extractive Summarization

Six normalized features were linearly combined (30% energy; 20% pitch; 20% HNR; 10% each for rate, formant F1, word count) to yield an importance_score per segment. The top 50%(2 609 segments) were embedded via SBERT and subjected to a redundancy-aware selection (Maximal Marginal Relevance, = 0.6), extracting up to 150 sentences per lecture. These were then re-ordered by time, producing concise extractive summaries that preserve narrative structure and highlight the most salient $\tilde{2}$–3 minutes of content per lecture.

H. Abstractive Summarization

In the abstract summary phase, each extractive summary is transformed into a fluid narrative-style paragraph using the BART largeCNN model - a transformer-based encoder-decoder that has been fine-tuned on the news data. Instead of merely stitching sentences together, BART learns to rephrase, merge, and condense information, producing summaries of roughly 100–150 words that read more like human-written prose. Beam-search decoding (with multiple beams) ensures the model balances fidelity to the source with brevity, while a length-penalty parameter helps control the final summary

size. The result is a single, coherent paragraph per lecture that highlights the essential points in a natural, easy-to-read format.

I. Quantitative Evaluation

To assess the quality of both extractive and abstractive summaries, four complementary evaluation metrics were used:

Flesch–Kincaid Grade Level: This readability metric estimates the school-grade level required to understand a piece of text. It is computed from average sentence length and average syllables per word. Lower scores indicate easier text (e.g. grades 6–8), while higher scores (e.g. grades 12+) indicate more complex academic language.

SMOG Index: The Simple Measure of Gobbledygook (SMOG) focuses on the density of polysyllabic words to predict the years of education needed to comprehend the text. It is particularly sensitive to vocabulary complexity and is widely used in health communication and policy writing to ensure accessibility.

Gunning–Fog Index: This index combines sentence length with the proportion of complex words (three or more syllables) to yield a grade-level estimate. Like Flesch–Kincaid, it helps gauge how easily a general audience can read and understand the summary.

Semantic Similarity (Cosine Similarity): Using pre-trained sentence embeddings (e.g. SBERT), the cosine of the angle between a summary's embedding and that of the original lecture text measures content fidelity. Values range from –1 (completely dissimilar) to 1 (identical), with higher scores indicating that the summary preserves the core semantics of the source.

Together, these four metrics provide a balanced evaluation of readability (Flesch–Kincaid, SMOG, Gunning–Fog) and content fidelity (semantic similarity), guiding optimization of summary length, complexity, and faithfulness.

J. Note generation

Building on the merged dataset, the work generated bullet-point notes via a hybrid acoustic-semantic pipeline It began by analyzing concept density, which averaged around 0.07 unique noun phrases or words per sentence. To ensure relevance, semantic coherence was computed using principal component analysis (PCA) and embedding similarity to rank segments by content centrality. The final importance score for each segment was derived from a weighted combination of 60% acoustic features and 40% coherence scores, selecting about 15 key segments per lecture. To promote diversity, the system ensured that each lecture summary included at least 10 unique concepts. Finally, an abstractive summarization model was applied to synthesize these selected segments into 10 concise and informative bullet-point highlights.

IV. RESULTS AND DISCUSSION

As shown in Figure 2 the distribution of mean pitch values across different lecture recordings. It is evident that lectures such as psychology_high_res_audio and creating_breakthrough_products_MIT exhibit higher variability and a broader pitch range, indicating more expressive or dynamic speech delivery. In contrast, lectures like phonetics_high_res_audio and cities_and_decarbonization_standard_res_audio display a narrower range of pitch, suggesting a more monotone or consistent tone. This pitch variation can influence listener engagement and plays an important role in identifying key emphasis areas during summarization.

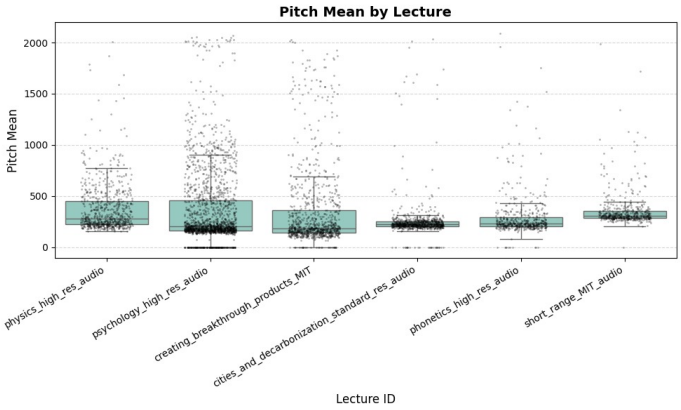

Fig. 2. Mean Pitch Values

As shown in Figure 3 how pitch mean, pitch standard deviation, and energy vary over time in a lecture. The frequent spikes in pitch and energy indicate dynamic speech delivery. These prosodic patterns help identify emphasis and emotionally charged segments during summarization.

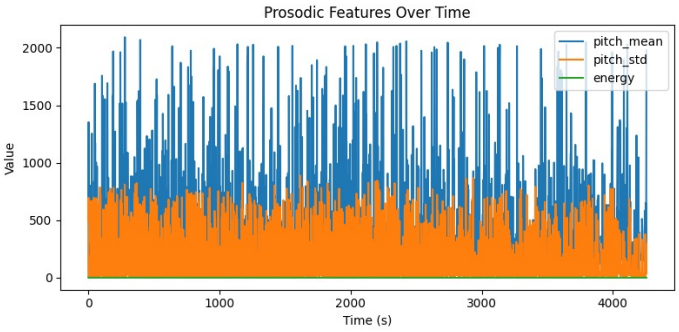

Fig. 3. Prosodic Features Over Time

As shown in Figure 4 tracks the energy levels across the duration of the physics lecture. We observe multiple fluctuations, suggesting changes in speaking intensity and emphasis. Such variations assist in detecting important parts of the lecture for potential highlighting.

As shown in Figure 5 compares the jitter (local) distributions across six lectures, which reflect voice stability. Most lectures show a peak around low jitter values, indicating consistent vocal delivery. However, slight differences suggest variation in speaker clarity and tone control.

As shown in Figure 6 displays the speech rate and articulation rate over time, with the articulation rate consistently

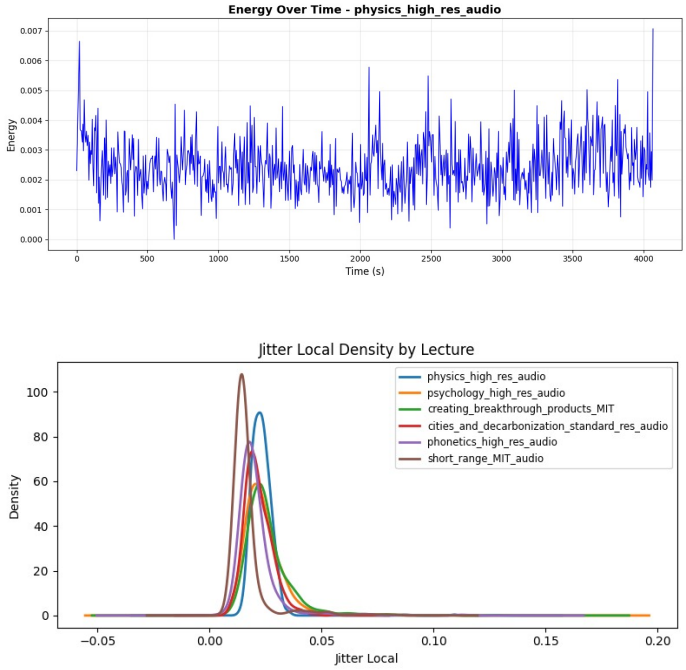

Fig. 5. Jitter Local Density by Lecture

higher. The data shows the speaker maintained a stable pace with occasional peaks. Tracking these rates can help determine speaker fluency and engagement.

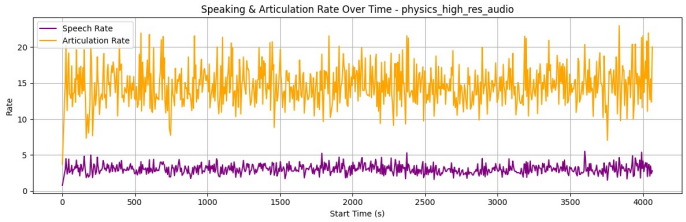

Fig. 6. Speaking & Articulation Rate physics_high_res_audio

As shown in Figure 7 highlights the energy difference between adjacent segments in a phonetics lecture. Most values remain close to zero, indicating stable delivery with few spikes. A notable jump near the 2000s mark may correspond to a shift in topic or emphasis.

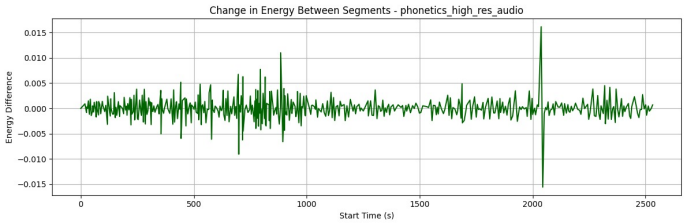

Fig. 7. Change in Energy Between Segments phonetics_high_res_audio

As shown in Figure 8 Flesch-Kincaid, SMOG, and Gunning Fog readability scores for extractive summaries. Lectures like cities_and_decarbonization and short_range_MIT show higher difficulty, while phonetics and creating_breakthrough_products are simpler. These scores help assess how accessible the extractive summaries are to learners.

	lecture_id	flesch_kincaid_extractive	smog_index_extractive	gunning_fog_extractive
0	physics_high_res_audio	7.274642	10.202149	9.462280
1	psychology_high_res_audio	6.752708	9.988104	9.261329
2	creating_breakthrough_products_MIT	6.583862	9.806752	9.235239
3	cities_and_decarbonization_standard_res_audio	13.202156	12.951904	15.897950
4	phonetics_high_res_audio	6.215220	8.675015	8.767954
5	short_range_MIT_audio	11.825982	13.602022	14.830473

Fig. 8. Readability Scores (Extractive)

As shown in Figure 9 readability metrics for abstractive summaries across lectures. short_range_MIT_audio has the highest complexity, while phonetics has the lowest. The variations reflect how abstractive methods affect summary readability depending on the lecture style.

	lecture_id	flesch_kincaid_abstractive	smog_index_abstractive	gunning_fog_abstractive
0	physics_high_res_audio	9.367273	11.208143	10.242424
1	psychology_high_res_audio	14.363596	15.903189	17.014035
2	creating_breakthrough_products_MIT	5.813777	10.125757	9.806383
3	cities_and_decarbonization_standard_res_audio	6.937963	11.698219	11.325926
4	phonetics_high_res_audio	4.006389	5.985473	5.388889
5	short_range_MIT_audio	25.719032	17.122413	28.670968

Fig. 9. Readability Scores (Abstractive)

As shown in Figure 10 compares the semantic similarity between original content and generated summaries. Extractive methods generally achieve higher similarity scores than abstractive ones. However, creating_breakthrough_products_MIT shows strong performance in both, indicating well-preserved content during simplification.

	lecture_id	semantic_similarity_extractive	semantic_similarity_abstractive
0	physics_high_res_audio	0.587242	0.234593
1	psychology_high_res_audio	0.668815	0.537644
2	creating_breakthrough_products_MIT	0.739676	0.608908
3	cities_and_decarbonization_standard_res_audio	0.634520	0.450196
4	phonetics_high_res_audio	0.592598	0.397455
5	short_range_MIT_audio	0.473312	0.541311

Fig. 10. Semantic Similarity – Extractive vs. Abstractive

As shown in Figure 11 interface allows users to upload audio files in formats like WAV, MP3, or M4A with a size limit of 200MB. Users can choose between extractive and

abstractive summarization methods and adjust the number of notes using a slider. Once a file is uploaded, the system automatically begins transcription. This clean and user-friendly layout ensures that users can easily configure their settings and begin summarization with minimal effort.

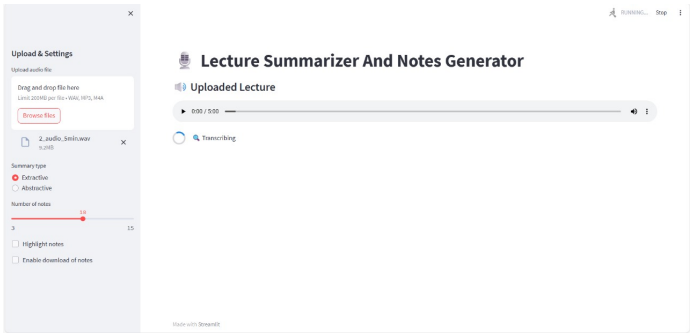

Fig. 11. Upload and Transcribe Interface

As shown in Figure 12 After successful transcription, the system displays a structured summary of the lecture based on the selected summarization type. Users also have the option to view the full transcript for deeper review or reference. The summary content is generated using NLP models and provides a coherent, concise version of the lecture material. This step confirms that the system has successfully understood and processed the audio input into readable text.

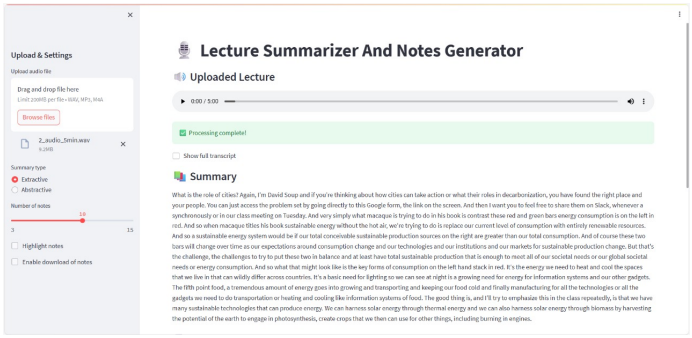

Fig. 12. Transcription and Summary Display

As shown in Figure 13 The "Generated Notes" section shows clear, bullet-point notes derived from key lecture segments. These notes highlight important concepts, instructions, or insights mentioned during the session. Users can highlight the notes, download them, and adjust how many are displayed. This feature supports quick review, note-sharing, and improves information retention for students and professionals alike.

The overall results highlight the success of the proposed system in generating meaningful and structured lecture summaries by integrating prosodic features, acoustic signals, and textual readability metrics. The analysis of pitch, energy, and speech rate across time provided key indicators of speaker emphasis, enabling the identification of important lecture segments. Readability scores showed that while abstractive summaries varied in complexity, they often presented the content in

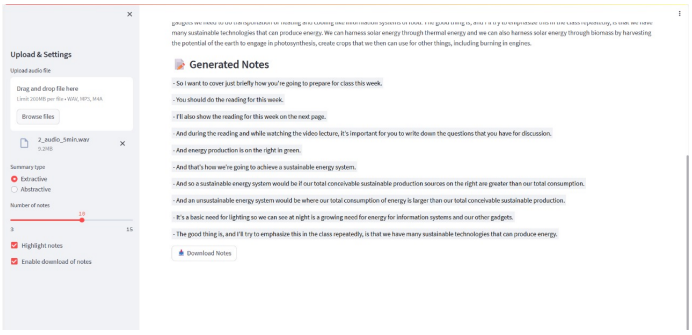

Fig. 13. Notes Generation Panel

a more natural and condensed form. In contrast, extractive summaries consistently maintained higher semantic similarity, preserving the original lecture intent more accurately. These insights were seamlessly integrated into a user-friendly web application that allows users to upload audio, choose summary type, and download bullet-point notes—offering an accessible, efficient tool for automated note-taking and enhanced educational support.

V. CONCLUSION AND FUTURE SCOPE

In conclusion, this study presents an automated lecture summarization system that effectively processes audio content to generate meaningful summaries and notes by leveraging prosodic and textual features. The analysis of pitch, energy, and speech rate provided valuable insights into speaker emphasis and delivery style, aiding in the identification of key content segments. Readability evaluations confirmed that extractive summaries are generally more accessible, while abstractive summaries offer better rewording and conciseness. Semantic similarity analysis validated the content alignment of both methods, with extractive models maintaining higher fidelity to the source material. The overall system supports faster content review, reduces manual effort, and enhances comprehension for students. This makes the tool especially beneficial in academic environments where long lectures are common and note-taking is challenging. As a future enhancement, incorporating speaker diarization could help better handle multi-speaker lectures and improve segment attribution. Additionally, integrating visual features from lecture slides could provide multimodal summarization. Personalized summary generation based on user feedback or difficulty levels could further improve learning outcomes. Finally, deploying this tool as a web application or browser plugin would make it more accessible and user-friendly for learners worldwide.

REFERENCES

[1] MUHZINA M A, Sulfath P M, Sheena K M. Smart Note Taker: A Digital Assistant for Efficient Note-taking. TechRxiv. April 11, 2025. DOI: 10.36227/techrxiv.174438691.12985816/v1

[2] M. Wyawahare, M. Shelke, S. Bhorge and R. Agrawal, "AI Powered Multilingual Meeting Summarization," 2024 14th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2024, pp. 86-91, doi: 10.1109/Confluence60223.2024.10463307.

[3] Benedetto, I., La Quatra, M., Cagliero, L. et al. Abstractive video lecture summarization: applications and future prospects. Educ Inf Technol 29, 2951–2971 (2024). https://doi.org/10.1007/s10639-023-11855-w

[4] M. Hotta, C. S. Leow, N. Kitaoka and H. Nishizaki, "Evaluation of Speech Translation Subtitles Generated by ASR with Unnecessary Word Detection," 2024 IEEE 13th Global Conference on Consumer Electronics (GCCE), Kitakyushu, Japan, 2024, pp. 815-819, doi: 10.1109/GCCE62371.2024.10760522.

[5] S. Ganguly, S. Mandal, N. Das, B. Sadhukhan, S. Sarkar and S. Paul, "WhisperSum: Unified Audio-to-Text Summarization," 2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), Hassan, India, 2024, pp. 1-7, doi: 10.1109/IACIS61494.2024.10721926.

[6] C. Xu, W. Jia, R. Wang, X. He, B. Zhao and Y. Zhang, "Semantic Navigation of PowerPoint-Based Lecture Video for AutoNote Generation," in IEEE Transactions on Learning Technologies, vol. 16, no. 1, pp. 1-17, 1 Feb. 2023, doi: 10.1109/TLT.2022.3216535.

[7] Sakshil Verma; Saksham Thareja; Dr. P. Supraja. (Volume. 8 Issue. 4, April - 2023) "Meeting Insights Summarisation Using Speech Recognition." , International Journal of Innovative Science and Research Technology (IJISRT), www.ijisrt.com. ISSN - 2456-2165 , PP :- 1747-1754. https://doi.org/10.5281/zenodo.7912314.

[8] H. Modi, A. Patel, I. Joshi and P. Kanani, "A Multimodal Approach to Multispeaker Summarization and Mind Mapping for Audio Data," 2023 International Conference on Advanced Computing Technologies and Applications (ICACTA), Mumbai, India, 2023, pp. 1-6, doi: 10.1109/ICACTA58201.2023.10393859.

[9] G. N, V. P and P. S, "Interactive Audio Indexing and Speech Recognition based Navigation Assist Tool for Tutoring Videos," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 1678-1682, doi: 10.1109/ICSCDS53736.2022.9760784.

[10] S. Kotey, R. Dahyot and N. Harte, "Fine Grained Spoken Document Summarization Through Text Segmentation," 2022 IEEE Spoken Language Technology Workshop (SLT), Doha, Qatar, 2023, pp. 647-654, doi: 10.1109/SLT54892.2023.10022829.

[11] A. Karunasena, P. Bandara, J. A. T. P. Jayasuriya, P. D. Gallage, J. M. S. D. Jayasundara and L. A. P. Y. P. Nuwanjaya, "EduEasy - Smart Learning Assistant System," 2021 9th International Conference on Information and Education Technology (ICIET), Okayama, Japan, 2021, pp. 1-5, doi: 10.1109/ICIET51873.2021.9419613.

[12] N. Bharti, S. N. Hashmi and V. M. Manikandan, "An Approach for Audio/Text Summary Generation from Webinars/Online Meetings," 2021 13th International Conference on Computational Intelligence and Communication Networks (CICN), Lima, Peru, 2021, pp. 6-10, doi: 10.1109/CICN51697.2021.9574684.

[13] T. Hayashi, T. Yoshimura, M. Inuzuka, I. Kuroyanagi and O. Segawa, "Spontaneous Speech Summarization: Transformers All The Way Through," 2021 29th European Signal Processing Conference (EUSIPCO), Dublin, Ireland, 2021, pp. 456-460, doi: 10.23919/EUSIPCO54536.2021.9615996.

[14] D. Chand and H. Oğul, "A Framework for Lecture Video Segmentation from Extracted Speech Content," 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 2021, pp. 000299-000304, doi: 10.1109/SAMI50585.2021.9378632.

[15] Kumar, Manoj & Gaurav, Sumit & Akhtar, Suhail & Varshney, Siddharth. (2021). Extracting Notes From Youtube Video. 1-5. 10.1109/CONIT51480.2021.9498392.

[16] H. Ibrahim and A. Varol, "A Study on Automatic Speech Recognition Systems," 2020 8th International Symposium on Digital Forensics and Security (ISDFS), Beirut, Lebanon, 2020, pp. 1-5, doi: 10.1109/ISDFS49300.2020.9116286.

[17] A, Vinnarasu & Jose, Deepa. (2019). Speech to text conversion and summarization for effective understanding and documentation. International Journal of Electrical and Computer Engineering (IJECE). 9. 3642. 10.11591/ijece.v9i5.pp3642-3648.

[18] S. Furui, T. Kikuchi, Y. Shinnaka and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," in IEEE Transactions on Speech and Audio Processing, vol. 12, no. 4, pp. 401-408, July 2004, doi: 10.1109/TSA.2004.828699.