iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** XI **Month of publication:** November 2026

DOI: https://doi.org/10.22214/ijraset.2026.85861

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 XI Nov 2026- Available at www.ijraset.com

AccessLearnAI: A Survey on Natural Language Processing and Deep Learning Techniques for Lecture Accessibility

Vishal Mundewadi[1], Saraswath H D[2], Siddesha C[3], Mr. Chandrasekhar V[4]
[1, 2, 3, 4]*Department of Computer Science and Engineering, Ballari Institute of Technology and Management, Ballari, Karnataka, India*
[4]*Assistant Professor, Department of Computer Science and Engineering, Ballari Institute of Technology and Management, Ballari, Karnataka, India*

Abstract: ***Inclusive education requires that lecture content be available in formats that students with visual, hearing, or cognitive impairments can use. Most learning systems offer only one such format, if any. Natural Language Pro-cessing (NLP) and Deep Learning (DL) now make it possible to turn a lecture into captions, simplified notes, audio nar-ration, and Braille. This paper surveys work on automatic speech recognition, text simplification and summarization, neural text-to-speech synthesis, and Braille conversion, with a focus on their use in education. We compare models such as Whisper, T5, BERT, Tacotron 2, and FastSpeech 2 in terms of reported performance, computational cost, and language support. Building on this review, we describe Ac-cessLearnAI, a framework that combines these components into a single pipeline, and a web prototype that produces a transcript, simplified notes, summaries, key points, MP3 narration, and a Braille-ready file from one lecture. We also discuss open problems: computational cost, limited sup-port for Indian languages, and the lack of shared evaluation benchmarks.***
Index Terms: ***Natural Language Processing, Deep Learning, Accessibility, Speech Recognition, Text-to-Speech, Assistive Technology, Inclusive Education, Transformer Models.***

I. INTRODUCTION

Education is recognized as a basic right, yet for many students with visual, hearing, or cognitive disabilities the lecture-based classroom remains hard to access. A student with low vision can-not read what is written on the board; a student with hearing loss cannot follow spoken instruction; a learner with dyslexia may struggle with dense academic text. Support for these learners has often been fragmented, underfunded, or missing altogether.

Recent progress in Artificial Intelligence (AI), and in partic-ular in NLP and DL, offers new options. Automatic Speech Recognition (ASR) can transcribe lecture audio with high ac-curacy [1]. Transformer-based language models can rewrite complex sentences in simpler language [4]. Neural Text-to-Speech (TTS) systems produce speech close to human record-ings [7, 8], and Braille translation software can render digital text for refreshable Braille displays and embossers [9].

Each of these technologies is mature on its own, but they are rarely combined. Most deployed tools handle one modality at a time, such as a captioning tool, a screen reader, or a summarizer. None of them addresses the overlapping needs of a student with more than one disability.

This paper reviews work in four areas (ASR, text simpli-fication, neural TTS, and Braille conversion), discusses their strengths and limitations in education, and describes AccessLear-nAI, a framework that brings these components together, along with a working web prototype. The rest of the paper is organized as follows. Section II states the problem. Section III reviews the literature. Section IV de-scribes the AccessLearnAI framework and prototype. Section V discusses findings and research gaps. Section VI lists limitations, and Section VII concludes.

II. PROBLEM STATEMENT

Students with disabilities need the same lecture content in sev-eral formats. A complete accessibility solution should cover at least four needs:

1) *Speech-to-text:* hearing-impaired students need accurate tran-scription of spoken lectures.
2) *Text simplification:* students with cognitive or learning diffi-culties benefit from simpler vocabulary and sentence structure, along with structured summaries.

©IJRASET: All Rights are Reserved

3) *Text-to-speech:* visually impaired students need clear audio narration of notes and transcripts.
4) *Braille output:* blind students need text in a Braille format that works with standard embossers and refreshable displays. Current systems address these needs separately. A student who is both hard of hearing and visually impaired must use at least two tools that were not designed to work together. In addi-tion, there are few evaluation benchmarks specific to educational accessibility, which makes it hard to compare systems or judge how usable they are in practice.

III. LITERATURE REVIEW

A. Automatic Speech Recognition for Lectures

ASR is the first stage of any lecture accessibility pipeline: sim-plification, narration, and Braille conversion all depend on a reliable transcript.

Radford et al. [1] released Whisper, an encoder-decoder Trans-former [14] whose training data consists of roughly 680,000 hours of web audio paired with loosely aligned transcripts in many languages. When applied to a new test set with no extra tuning, it performs on par with models trained specifically for that set, and its error rates come close to those of human listeners. Its multilingual training makes it a practical choice where more than one language is used in class. Self-supervised pre-training, as in wav2vec 2.0 [18], offers another way to reduce the amount of labeled speech needed.

Classroom use of ASR has a longer history. Kawahara et al. [20] created a live note-taking aid for deaf and hard-of-hearing students in Japanese university classes, tuning the recog-nizer to the vocabulary and speaking style of lectures. Ranchal et al. [2] evaluated speech recognition for both real-time caption-ing and post-lecture transcription in classrooms. In their classes, text generated after the session turned out cleaner and simpler to set up than on-screen live captions, because the live system had to trade accuracy for speed.

End-to-end neural models have since replaced hand-built pipelines. Deep Speech 2 [3] stacks convolutional layers, which pick up short-range patterns in the spectrogram, on top of recur-rent layers that track how speech unfolds over time. The same design was trained separately for English and for Mandarin, copes with background noise and accented speakers, and on some test sets matches crowd-sourced human transcription. The cost is scale: training used thousands of hours of transcribed speech and a large GPU cluster.

B. Text Simplification and Summarization

A transcript may still be hard to read for students with cognitive disabilities or limited literacy. Text simplification models rewrite complex text in a more readable form while keeping its meaning. Sheang and Saggion [4] adapted the pre-trained T5 model [16] to rewriting single sentences, prepending special tokens that let a user ask for shorter output or easier words depending on the audience. Their system scored slightly higher than the earlier BART-based state of the art. Alissa and Wald [5] built two encoder-decoder simplifiers, one of them using BERT [15] representations, motivated by readers who find web pages hard to follow, such as people with cognitive impairments, second-language learners, and young readers.

Summarization can also support accessibility by letting stu-dents grasp the main ideas of a lecture without reading the whole transcript. Miller [6] offered lecture summaries as a web service: each transcript sentence is turned into a BERT vector, the vectors are grouped with K-Means, and one representative sentence is taken from every group. The author reports encouraging output but also points to lecture passages the method handled poorly.

C. Neural Text-to-Speech Synthesis

TTS converts written content into speech, which helps visually impaired students and those who prefer to learn by listening. Tacotron 2 [7] first maps characters to mel spectrogram frames with an attention-based recurrent network and then turns those frames into a waveform with a WaveNet vocoder. In listening tests its ratings (MOS) were nearly as high as those for studio recordings, but producing audio one frame at a time makes it slow.

FastSpeech 2 [8] produces all frames in parallel and is given explicit targets for how long, how high, and how loud each sound should be. Compared with the first FastSpeech it needs about a third of the training time, listeners rate it as natural as Tacotron 2, and it generates audio far more quickly, which is useful when narration must keep pace with a lecture. VITS [17] goes a step further and outputs the waveform directly from text.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

D. Braille Conversion

Braille conversion is essential for blind students but has received less attention from the machine learning community than the other components. Blenkhorn and Evans [9] showed how Braille copies can be generated straight from ordinary word-processor files. In their view, turning plain text into uncontracted or con-tracted Braille is no longer the hard part; carrying headings, lists, and page layout across to the Braille version is. Mathemat-ical and scientific notation remains difficult in practice, which matters because this content is common in higher education.

Support for Indian languages is more limited. Dasgupta and Basu [21] wrote rule-based software that converts text in Indian scripts to Braille and back again, and paired it with speech output so a user can hear what has been entered.

E. Integrated Accessibility Systems

Some recent work goes beyond single-modality tools. Anderer et al. [11] began by asking 12 blind and low-vision students how AI could make lectures and lecture material easier for them to use. Based on those conversations they designed LectureAs-sistant, where the video sits next to a running transcript and a chat window; the chat can answer content questions from the transcript and describe whatever is on screen at that moment. Seven students tried the finished prototype, and every one of them rated it as helpful and easy to use.

Stelea et al. [22] describe AccessiLearnAI, an e-learning plat-form designed around accessibility from the start; it combines content delivery in several formats with personalization and offline use as a Progressive Web App, and its early evaluations report better engagement and learning outcomes. Despite the similar name, it is a separate project from the framework pro-posed here.

Broader reviews place this work in context. Patel et al. [10] examine classical classifiers such as SVMs and random forests for auditory assistive tools, detectors such as YOLO for object recognition and navigation aids for blind users, and newer gen-erative AI applications. Zdravkova et al. [12] focus on children, covering AI tools that help disabled pupils communicate and learn, and they close with the ethical risks of using such tools with young users. For Indian languages specifically, Bhogale et al. [13] collected 59 test sets covering 12 Indian languages under the name Vistaar. After further training Whisper on Indian speech, their IndicWhisper models had the best word error rate in 39 of those 59 tests. This shows both that general-purpose models leave room for improvement on Indian languages and that targeted fine-tuning helps.

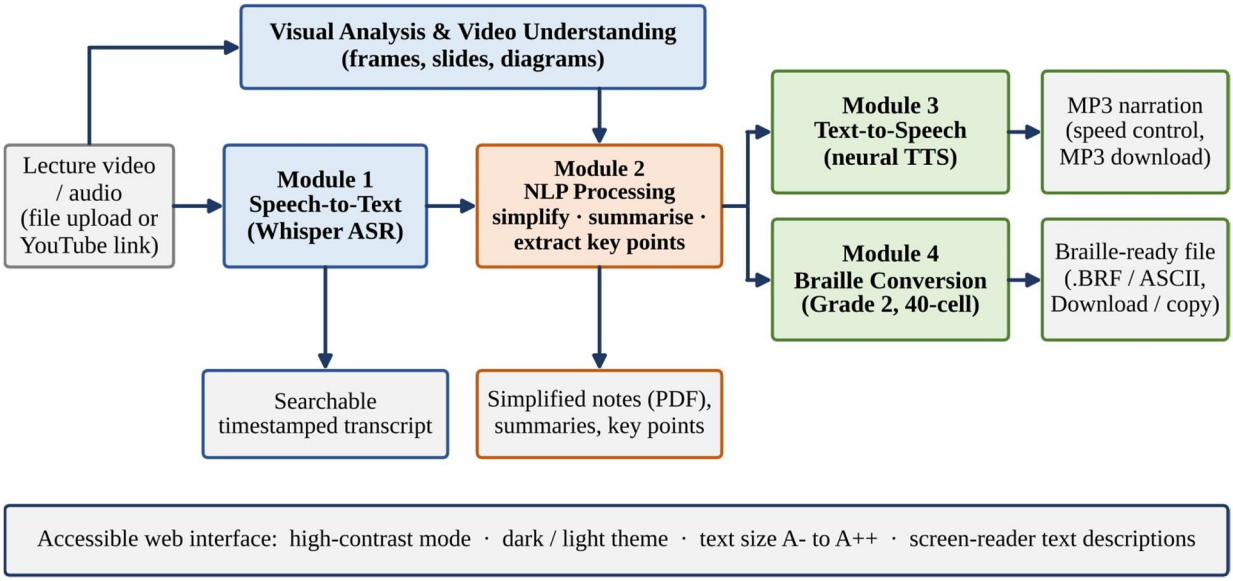


Figure 1. Overall architecture of AccessLearnAI: lecture input, four processing modules, visual analysis, accessible outputs, and the accessible web interface.

IV. PROPOSED FRAMEWORK: ACCESSLEARNAI

Based on the reviewed literature, this section describes Ac-cessLearnAI, a framework that combines these technologies into one pipeline for lecture accessibility, and the web prototype built to test it. The aim is simple: a student should receive the same lecture content in whichever format suits their needs, without switching between separate tools.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

A. System Architecture

The AccessLearnAI pipeline has four core processing modules and a supporting visual analysis stage, shown in Fig. 1:

1) *Speech-to-Text Module*. Lecture audio is transcribed by a Whisper-based ASR engine [1], chosen for its multilingual support and robustness to noise. The module outputs a times-tamped transcript.
2) *NLP Processing Module.* The transcript is passed to a T5-based simplification model [4]. Key concepts are identified with BERT embeddings and given plain-language explana-tions. In parallel, a summarization model produces structured lecture notes.
3) *Text-to-Speech Module*. The simplified text and summaries are sent to a neural TTS engine based on FastSpeech 2 [8], which produces audio narration for visually impaired stu-dents.
4) *Braille Output Module.* The simplified text is converted to Grade 2 Braille, producing a file compatible with standard embossers and refreshable Braille displays.

In addition, a *Visual Analysis and Video Understanding* stage samples video frames and uses a vision model to describe slides, diagrams, and other on-screen content. Its output is merged with the transcript before the NLP module, so that the notes can include flowcharts with text descriptions.

B. Integration and Output Formats

From a single lecture recording, AccessLearnAI produces a full transcript, simplified notes, short and detailed summaries with key points, an audio narration file, and a Braille-formatted document. Through a web interface, users upload lectures and choose one or more output formats. The framework is modular. Each module can be replaced as better models appear, and the pipeline can be extended to new outputs, such as sign language generation, once those technolo-gies mature.

C. Prototype Implementation

To show that the framework is feasible, we implemented a web-based prototype. A lecture can be supplied as a file upload or as a YouTube link (Fig. 3a). For links, the interface checks that the address belongs to youtube.com or youtu.be, downloads the audio, and checks it automatically before conversion.

The lecture then passes through the eight-stage pipeline shown in Fig. 2: upload, transcription, visual analysis, video understanding, simplified notes, summary and key points, ac-cessibility output generation, and completion. Transcription uses Whisper. The visual analysis and video understanding stages capture on-screen content that the audio does not convey, such as slides, diagrams, and flowcharts, in a similar spirit to the vision-language question answering used in LectureAssis-tant [11]. Where the notes include a visual element such as a flowchart, a text description is provided with it.

A progress bar and a labeled step indicator (Fig. 3b) show the current stage, for example “Transcribing Audio with Whisper”. The interface itself is designed for accessibility: a toolbar offers a high-contrast mode, a dark/light theme toggle, and four text sizes (A-, A, A+, A++), and each flowchart in the notes has an expandable description for screen readers (Fig. 4b).

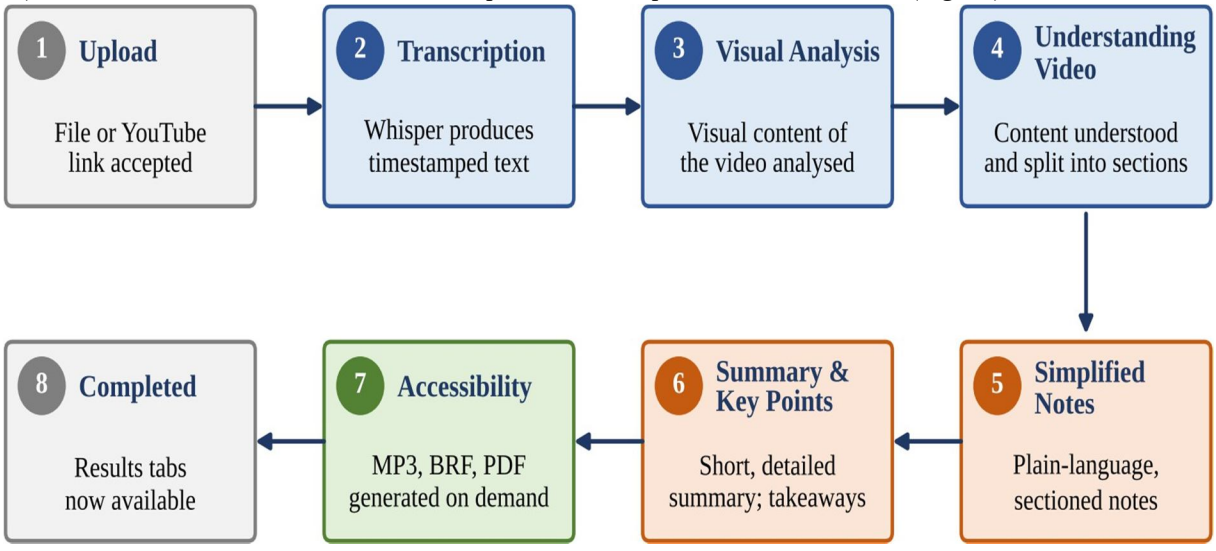
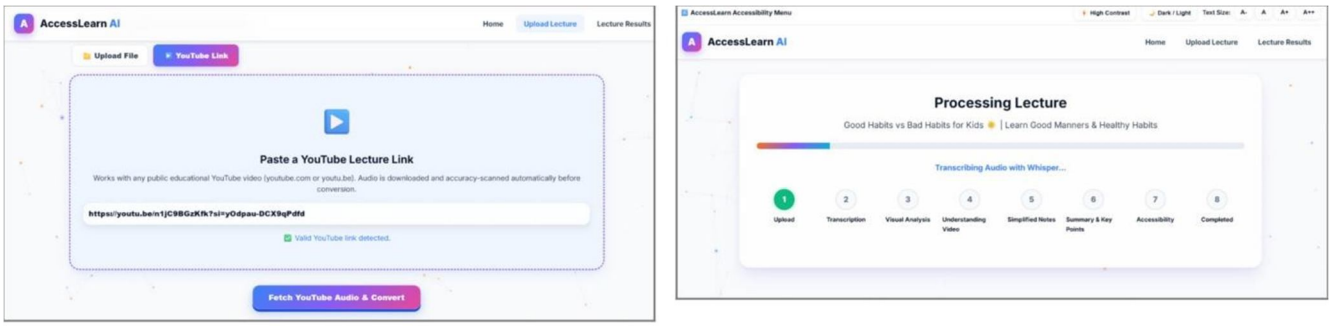


Figure 2. Eight-stage processing pipeline of the AccessLearnAI prototype, from upload to completion.

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 XI Nov 2026- Available at www.ijraset.com



(a) YouTube-link input screen (b) Eight-stage lecture processing view

Figure 3. Prototype input and progress screens: (a) YouTube-link input with link validation and (b) lecture processing view with step indicator and accessibility toolbar.

D. Output Interfaces

When processing finishes, results are shown in six tabs: Tran-script, Simplified Content, Summary, Key Points, MP3 Audio, and Braille (.BRF). Table II maps each output to the need it serves.

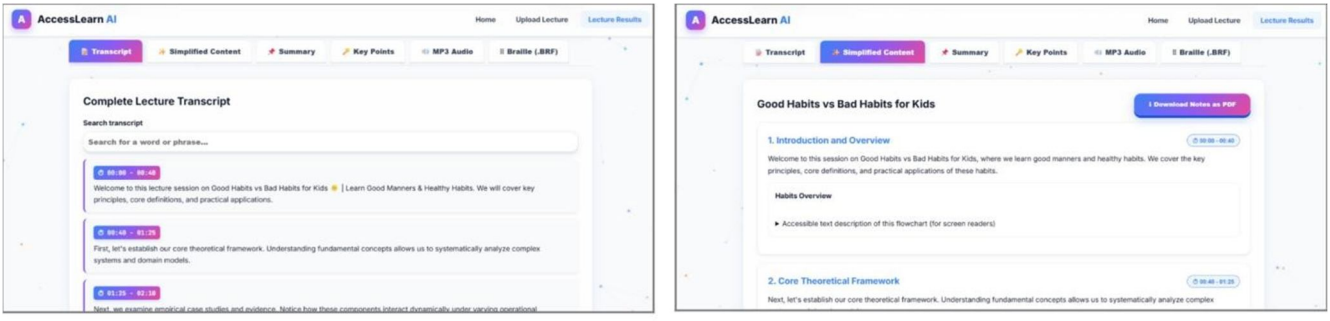
1) *Transcript.* The transcript is shown as timestamped segments (for example, 00:00–00:40) with a search box that filters by word or phrase (Fig. 4a). This directly supports hearing-impaired students and helps any student find a passage quickly.
2) *Simplified content.* The notes are organized into numbered sections, each with its time range, a plain-language explanation, and, where relevant, a flowchart with a text description for screen readers (Fig. 4b). The notes can be downloaded as a PDF.
3) *Summary and key points.* The Summary tab gives a short summary for quick orientation and a detailed summary that follows the structure of the lecture (Fig. 5a). The Key Points tab lists the main takeaways (Fig. 5b).
4) *Audio.* The user chooses which content to narrate (for exam-ple, the simplified notes) and generates speech, which plays in an accessible player with 10-second skip controls, a speed selec-tor, and a progress bar (Fig. 6a). The audio can be downloaded as an MP3 file.
5) *Braille.* The user chooses the content to convert and generates Braille. The interface shows the plain text beside the Braille, reports the output format (a 40-cell line, Grade 2), and offers a switch between Unicode Braille and ASCII BRF, a copy button, and a .BRF download for embossers and refreshable displays (Fig. 6b).

V. DISCUSSION

A. Key Findings

The reviewed literature shows that each technology used in AccessLearnAI is mature enough for educational use. Whis-per [1] is a strong general-purpose multilingual ASR model. Transformer-based simplification [4, 5] gives better results than earlier approaches. Neural TTS [7, 8] produces speech close to human quality, and FastSpeech 2 makes fast synthesis practi-cal. Rule-based Braille translation [9] works well for standard English prose.

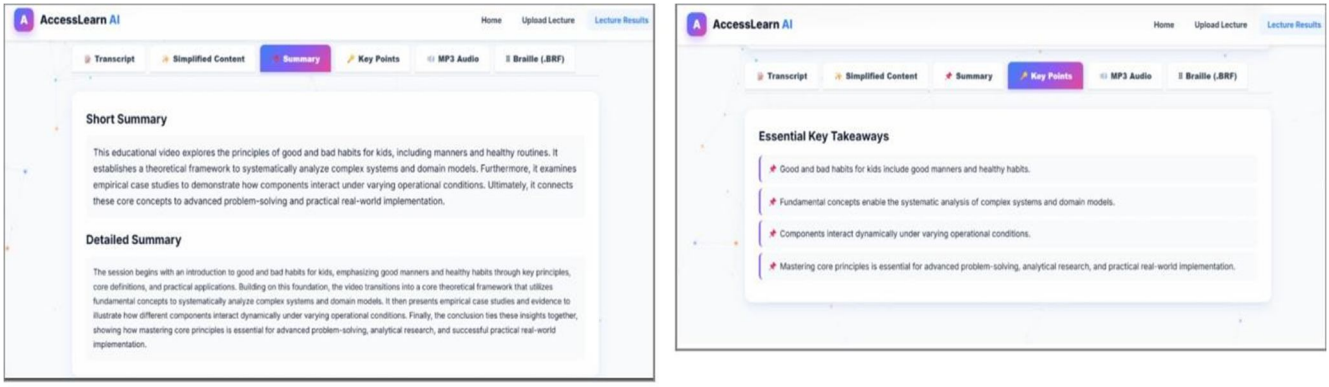
What we did not find is a system that combines all four capa-bilities in one pipeline. LectureAssistant [11] and AccessiLear-nAI [22] are the closest examples we reviewed. LectureAssistant



(a) Searchable timestamped transcript (b) Simplified content with accessible flowchart

Figure 4. Text outputs: (a) searchable timestamped transcript and (b) simplified content organized into timed sections with a PDF download option.

©IJRASET: All Rights are Reserved



(a) Short and detailed summary (b) Key takeaways

Figure 5. Summary outputs: (a) short and detailed summaries and (b) key takeaways.

focuses on blind and low-vision students, and AccessiLearnAI is a general e-learning platform; how far either covers Braille output and Indian languages for recorded lectures remains to be compared in detail. This gap between mature individual tools and the lack of integrated systems is the main finding of this sur-vey, and it is the problem AccessLearnAI is designed to address. Table I summarizes the techniques reviewed, with their reported strengths and limitations.

B. Research Gaps

Three gaps appear repeatedly in the reviewed literature.

1) *Language diversity.* Most systems perform best in English. Indian languages such as Kannada, Hindi, and Telugu are widely used as the medium of instruction, yet benchmarks such as Vistaar [13] show that general-purpose ASR leaves clear room for improvement on them, and Braille tools for these languages are less developed [21]. For AccessLearnAI to be useful in Indian institutions, support for regional languages is essential.
2) *Latency.* Real-time accessibility requires low latency across all four modules at once. Existing work addresses speed within individual components, for example non-autoregressive TTS [8], but we did not find a study that measures end-to-end latency for a system producing all four formats in parallel during a live lecture.
3) *Evaluation standards.* There is no shared benchmark for multi-format lecture accessibility. Each group evaluates on its own data with its own metrics, which makes cross-study comparison unreliable.

C. Ethical Considerations

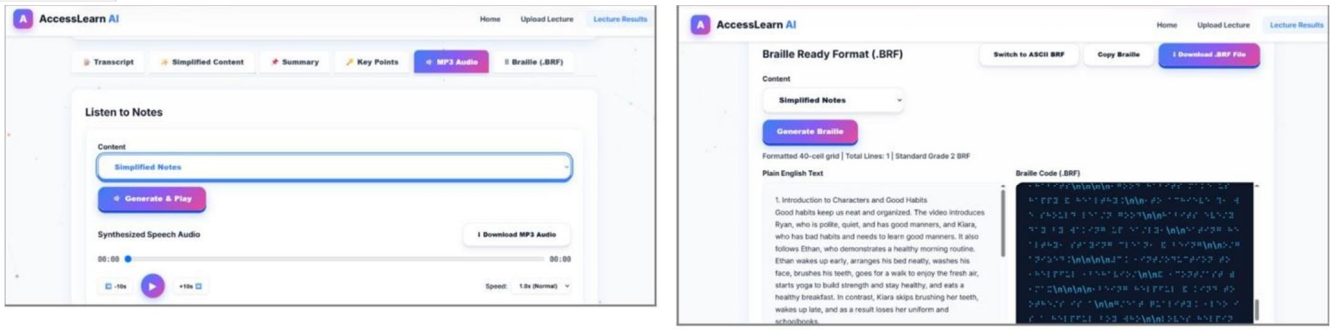
Any AI-based accessibility system must protect student data. Bias is a further concern: models trained mostly on English con-tent from well-resourced institutions may perform worse for stu-dents in other languages or less-resourced settings, which would reinforce the inequalities these tools are meant to reduce [12]. Reviews of Transformer and large language model deployment in other high-stakes domains [19] point to the same need for scrutiny. Responsible development of AccessLearnAI requires attention to training-data bias and evaluation that includes the students the system is meant to serve.

VI. LIMITATIONS

This study has several limitations.

First, its scope is limited to text, audio, and Braille. Sign language generation, which is essential for many deaf students for whom written language is a second language, is not covered. Second, most reviewed systems were evaluated on curated datasets. Real classrooms are noisier and less predictable, so performance in a live lecture at a typical Indian university is still an open question.

Third, user studies with students with disabilities are rare in the reviewed literature; Anderer et al. [11] is one of the few exceptions. A system that performs well technically but is confusing or frustrating for its users is not accessible. More



(a) MP3 audio generation and playback

(b) Braille Ready Format (.BRF) output

Figure 6. Non-visual outputs: (a) MP3 audio generation and playback and (b) Braille Ready Format (.BRF) preview with plain-text comparison.

TABLE I
COMPARISON OF KEY TECHNIQUES REVIEWED FOR LECTURE ACCESSIBILITY

Technique	Model / Approach	Task	Reported Strength	Limitation	Ref.
Speech recognition	Whisper (Transformer)	Multilingual ASR	Competitive zero-shot accuracy; approaches human robustness	Large models need significant compute	[1]
Speech recognition	Deep Speech 2 (CNN + RNN)	End-to-end ASR	Competitive with human transcribers on several benchmarks	Needs thousands of hours of labeled audio	[3]
Speech recognition	Lecture-adapted ASR	Classroom captioning	Supports hearing-impaired students in class	Live captions less accurate than post-lecture transcripts	[2, 20]
Text simplification	T5 with control tokens	Sentence simplification	Improved on BART+ACCESS on standard benchmarks	Works on single sentences; trained on general text	[4]
Text simplification	Transformer / BERT	Web text simplification	Targets readers with cognitive impairments	Not evaluated on lecture transcripts	[5]
Summarization	BERT + K-Means	Lecture summarization	Coherent extractive summaries	Struggles on some lecture content	[6]
Text-to-speech	Tacotron 2	Audio narration	MOS close to recorded speech	Slow autoregressive inference	[7]
Text-to-speech	FastSpeech 2	Audio narration	Comparable quality with much faster synthesis	Needs duration, pitch, and energy targets	[8]
Braille conversion	Rule-based translation	Grade 2 / Indian-language Braille	Mature for English prose	Layout and math notation; limited Indian-language tools	[9, 21]
Multi-format system	LectureAssistant (LLM + VLM)	Lecture access for blind and low-vision students	Rated useful by all participants	Small user study (7 students)	[11]

research should involve students with disabilities directly.

Fourth, the AccessLearnAI prototype is at an early stage. In initial runs on a short children's educational video, the notes for the opening section matched the video, but later sections, the summaries, and the key points contained generic wording about "complex systems and domain models" that did not match the video. This shows that the generation stages need stronger grounding in the transcript. In addition, the Braille preview currently shows literal newline escape sequences and reports a single output line, so line-break handling before Braille conver-sion must be fixed.

Finally, the prototype has not yet been evaluated empirically, and we have not yet benchmarked its processing time. A study of the full pipeline with real students in real classrooms, producing all output formats at once, is the most important direction for future work, and the authors intend to carry it out.

VII. CONCLUSION

This paper surveyed NLP and deep learning techniques for lec-ture accessibility, covering automatic speech recognition, text simplification and summarization, neural text-to-speech, and Braille conversion. We compared ten representative techniques in terms of reported performance, computational requirements, and suitability for education, and summarized them in Table I.

The individual technologies are mature: Whisper-based ASR, T5-based simplification, FastSpeech 2 TTS, and rule-based Braille conversion can each produce usable output in academic settings. What is missing is a system that brings them together, so students with multiple accessibility needs still have to switch between separate tools.

AccessLearnAI addresses this with a single pipeline that takes a lecture recording or a YouTube link and produces a transcript,

TABLE II
OUTPUT FORMATS PROVIDED BY THE ACCESSLEARNAI PROTOTYPE

Output	Primary Need Served	Prototype Features	Figure
Transcript	Hearing impairment	Timestamped segments (e.g., 00:00–00:40); keyword and phrase search	Fig. 4(a)
Simplified content	Cognitive or learning difficulties	Numbered, timed, plain-language sections; flowchart text descriptions for screen readers; PDF download	Fig. 4(b)
Summary and key points	Cognitive or learning difficulties; quick revision	Short and detailed summaries; list of key takeaways	Fig. 5
MP3 audio	Visual impairment	Selectable content; synthesized speech; 10 s skip, speed selector; MP3 download	Fig. 6(a)
Braille (.BRF)	Blindness	Grade 2 on a 40-cell line; plain text shown beside Braille; Unicode/ASCII BRF switch; copy and .BRF download	Fig. 6(b)

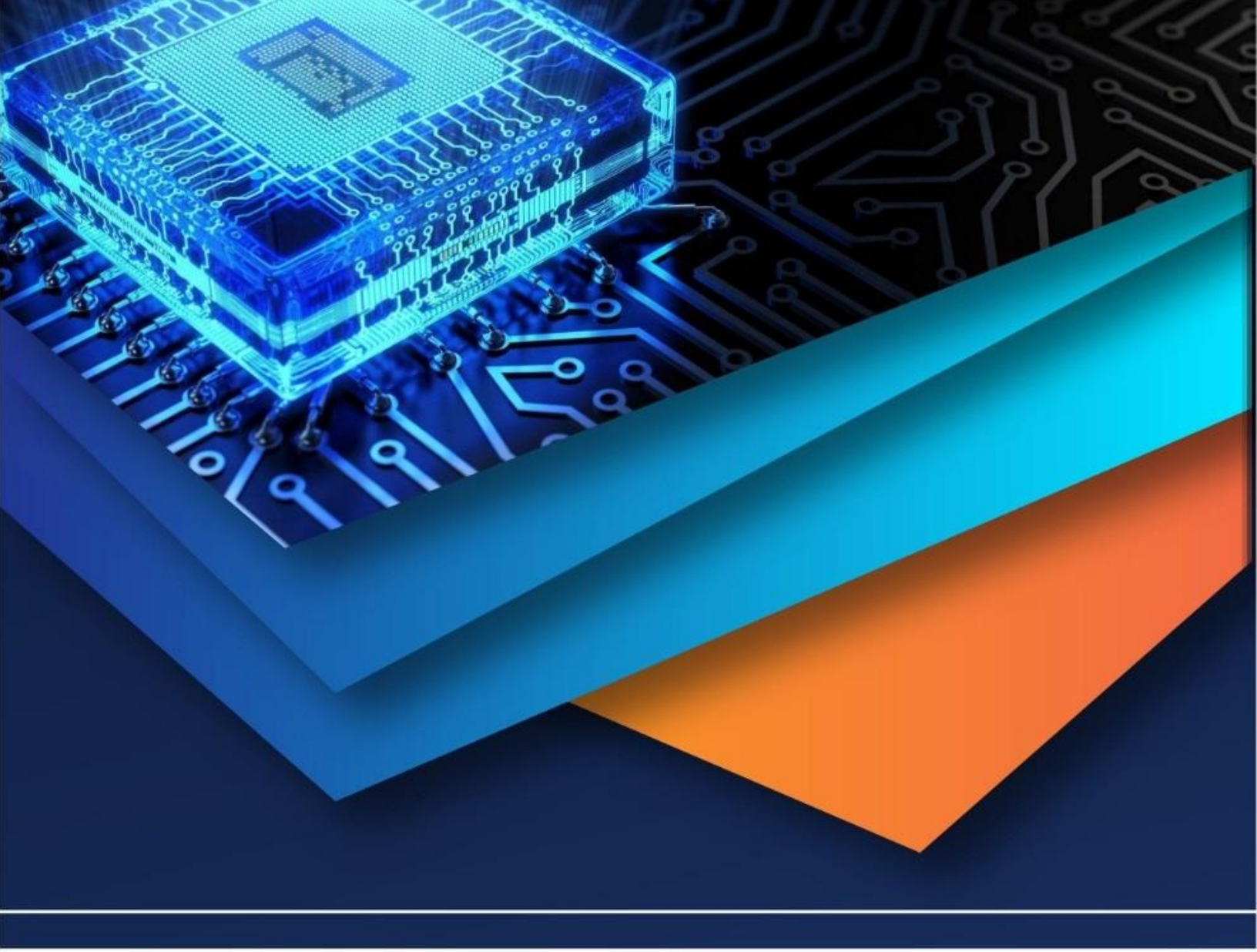
simplified notes, summaries and key points, audio narration, and a Braille file, through an interface with built-in accessibil-ity controls. Three open challenges set the agenda for future work: support for Indian languages, lower end-to-end latency for real-time use, and shared benchmarks for multi-format lec-ture accessibility. Progress on these will require collaboration between AI researchers, educators, accessibility specialists, and, above all, students with disabilities.

REFERENCES

[1] Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervi-sion," in Proc. 40th Int. Conf. Machine Learning (ICML), PMLR, vol. 202, pp. 28492–28518, 2023.
[2] R. Ranchal, T. Taber-Doughty, Y. Guo, K. Bain, H. Martin, J. P. Robinson, and B. S. Duerstock, "Using speech recognition for real-time caption-ing and lecture transcription in the classroom," IEEE Trans. Learning Technologies, vol. 6, no. 4, pp. 299–311, 2013.
[3] D. Amodei et al., "Deep Speech 2: End-to-end speech recognition in English and Mandarin," in Proc. 33rd Int. Conf. Machine Learning (ICML), PMLR, vol. 48, pp. 173–182, 2016.
[4] K. C. Sheang and H. Saggion, "Controllable sentence simplification with a unified text-to-text transfer transformer," in Proc. 14th Int. Conf. Natural Language Generation (INLG), Aberdeen, UK, 2021, pp. 341–352, doi: 10.18653/v1/2021.inlg-1.38.
[5] S. Alissa and M. Wald, "Text simplification using transformer and BERT," Computers, Materials & Continua, vol. 75, no. 2, pp. 3479–3495, 2023, doi: 10.32604/cmc.2023.033647.
[6] D. Miller, "Leveraging BERT for extractive text summarization on lec-tures," arXiv:1906.04165, 2019.
[7] J. Shen et al., "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 4779–4783.
[8] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," in Proc. Int. Conf. Learning Representations (ICLR), 2021
[9] P. Blenkhorn and G. Evans, "Automated Braille production from word-processed documents," IEEE Trans. Neural Systems and Rehabilitation Engineering, vol. 9, no. 1, pp. 81–85, 2001.
[10] P. Patel, S. Pampaniya, A. Ghosh, R. Raj, D. Karuppaih, and S. Kandasamy, "Enhancing accessibility through machine learning: A review on visual and hearing impairment technologies," IEEE Access, vol. 13, pp. 33286–33307, 2025, doi: 10.1109/ACCESS.2025.3539081.
[11] K. Anderer, K. Müller, L. Strobel, M. Wölfel, J. Niehues, and K. Gerling, "Making lecture videos accessible for students who are blind or have low vision through AI-assisted navigation and visual question answering," in Proc. 27th Int. ACM SIGACCESS Conf. Computers and Accessibility (ASSETS), Denver, CO, USA, 2025, doi: 10.1145/3663547.3746349.

[12] K. Zdravkova, V. Krasniqi, F. Dalipi, and M. Ferati, "Cutting-edge commu-nication and learning assistive technologies for disabled children: An arti-ficial intelligence perspective," Frontiers in Artificial Intelligence, vol. 5, art. 970430, 2022, doi: 10.3389/frai.2022.970430.
[13] K. Bhogale, S. Sundaresan, A. Raman, T. Javed, M. M. Khapra, and P. Kumar, "Vistaar: Diverse benchmarks and training sets for Indian language ASR," in Proc. INTERSPEECH, Dublin, Ireland, 2023.
[14] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, vol. 30, 2017. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
[15] C. Raffel et al., "Exploring the limits of transfer learning with a unified text-to-text transformer," J. Machine Learning Research, vol. 21, no. 140, pp. 1–67, 2020.
[16] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in Proc. Int. Conf. Machine Learning (ICML), 2021, pp. 5530–5540
[17] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A frame-work for self-supervised learning of speech representations," in Advances in Neural Information Processing Systems, vol. 33, 2020, pp. 12449–12460.
[18] S. Nerella, S. Bandyopadhyay, J. Zhang, M. Contreras, and P. Rashidi, "Transformers and large language models in healthcare: A review," Artifi-cial Intelligence in Medicine, vol. 154, art. 102900, 2024.
[19] T. Kawahara, N. Katsumaru, Y. Akita, and S. Mori, "Classroom note-taking system for hearing impaired students using automatic speech recog-nition adapted to lectures," in Proc. INTERSPEECH, Makuhari, Japan, 2010.
[20] T. Dasgupta and A. Basu, "A speech enabled Indian language text to Braille transliteration system," in Proc. Int. Conf. Information and Communication Technologies and Development (ICTD), Doha, Qatar, 2009, pp. 201–211.
[21] G. A. Stelea, D. Robu, and F. Sandu, "AccessiLearnAI: An accessibility-first, AI-powered e-learning platform for inclusive education," Education Sciences, vol. 15, no. 9, art. 1125, 2025, doi: 10.3390/educsci15091125.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)