

An Edge AI-Based Offline System for Educational Technology in Resource-Constrained Regions

Irin Jose¹, Nickolas Kenyeres¹, Kailey Cozart Quesada¹

¹College of Computing, Georgia Institute of Technology

Correspondence: ijose3@gatech.edu, nkenyeres3@gatech.edu, kcozart6@gatech.edu

Abstract

Many foundational studies have highlighted how AI can improve education, from automated grading to adaptive, real-time instruction. However, these benefits remain inaccessible to millions of students due to digital inequality. Traditional AI grading systems depend on constant cloud connectivity and high-power hardware, creating barriers for schools with limited electricity or minimal technological infrastructure. We present a proof of concept for an offline, browser-based educational assessment system that leverages on-device AI to provide automated grading without requiring internet connectivity or cloud infrastructure. The system combines WebAssembly-compiled optical character recognition with Google’s Gemini Nano language model, embedded directly in the Chrome browser, to process handwritten student submissions locally on a user’s device. This work grounds the discussion of offline, privacy-preserving educational AI in a proof-of-concept implementation. We also reflect on the ethical and societal implications of automated grading, including the risk of automation complacency and the limits of offline access in addressing educational inequity.

1 Introduction

Educational technology has the potential to transform learning access globally, yet the AI systems driving its most capable features increasingly depend on cloud infrastructure that millions of students cannot reliably reach. As of 2023, approximately 2.6 billion people, which is about one-third of the global population, remain entirely offline ([International Telecommunication Union, 2023](#)). Mobile internet exclusion is even broader. Around 3.45 billion people do not use mobile internet, of whom 3.1 billion live within existing network coverage but do not use it, while 350 million more lack coverage entirely ([GSMA Intelligence, 2024](#)), demonstrating that physical infrastructure alone is

insufficient to ensure digital inclusion. Electricity access compounds this divide. While nearly 100% of people in developed countries have reliable power, this figure drops to approximately 40% in low-income countries ([Isotani et al., 2023](#)). Internet speeds follow a similar gradient, with Liechtenstein averaging 211 Mbps, while Ethiopia and Guinea-Bissau average just 1 Mbps ([Khowaja et al., 2024](#)). A 2023 UNESCO global survey of over 450 schools and universities found that fewer than 10% had established formal policies or guidance for AI use, despite widespread and rapidly growing adoption of generative AI tools in classrooms ([UNESCO, 2023](#)).

Prior work has demonstrated that offline educational AI is achievable at scale. The AIED Unplugged initiative in Brazil developed a mobile app that enables teachers to scan handwritten essays using QR-coded paper. Operating offline with periodic synchronization, the system reduced the grading time from four months to 72 hours and reached over 500,000 students across 7,000 schools ([Isotani et al., 2023](#)). AI-assisted English instruction platforms in China and SMS-based AI tutoring in Kenya further demonstrate that scalable educational AI in resource-constrained environments is feasible ([Pedro et al., 2019](#)). These implementations directly support the United Nations Sustainable Development Goal 4 (SDG-4), which calls for inclusive and equitable quality education and the promotion of lifelong learning opportunities for all.

Beyond connectivity, cloud-dependent architectures present another challenge in terms of student data privacy. Frameworks such as FERPA in the United States and GDPR in Europe constrain how student work may be transmitted to and processed by third-party vendors, creating compliance risk for cloud-based grading services ([United States Congress, 1974](#); [European Parliament and Council of the European Union, 2016](#)). Comply-

Cloud-Based vs. Offline On-Device Grading

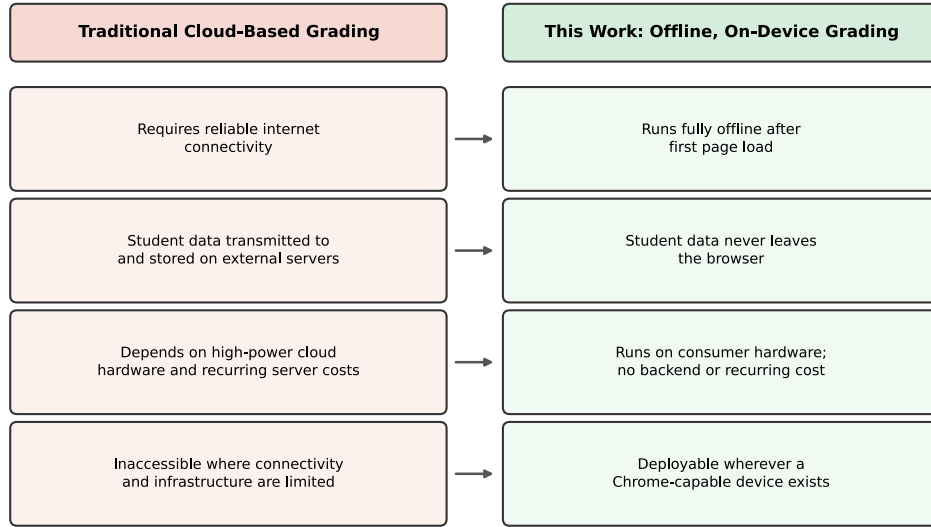


Figure 1: Traditional cloud-based grading systems versus the offline, on-device approach presented in this paper.

ing with these frameworks requires institutional capacity, such as legal and IT resources, that is often unevenly available, particularly outside high-income countries. Data minimization by architecture (never transmitting student data off-device at all) offers a more robust privacy guarantee than relying on institutions to correctly implement policy-based controls. Recent work on federated learning for personalized adaptive tutoring has explored how models can improve across institutions without centralizing raw student data (Saisanthiya et al., 2026).

Motivated by these findings, this research proposes to explore low-power, browser-embedded Edge AI as a viable solution for educational resources in resource-limited educational environments, as represented in Fig.1. This paper presents a fully offline, browser native grading prototype that processes handwritten student submissions using an on-device language model that executes locally on consumer hardware. Positioning this approach within the broader landscape of low-power, on-device AI requires distinguishing between two related but distinct categories, namely Edge AI and TinyML.

Edge AI is the broader category that includes running machine learning models on local devices rather than cloud servers, encompassing everything from powerful edge servers and gateways to smart

cameras, industrial IoT devices, and consumer hardware such as laptops and smartphones (Ray, 2022). These systems typically operate with significant computational resources, consuming watts to tens of watts of power, and can handle complex models with substantial memory requirements. For example, NVIDIA Jetson devices (\$249-\$1,999), which are prominent Edge AI platforms, take 7-60 watts of power and deliver up to 275 trillion operations per second (TOPS) (NVIDIA Corporation, 2024).

In contrast, TinyML is a specialized subset of Edge AI focused specifically on ultra-low-power microcontrollers and sensors (Sanchez-Iborra and Skarmeta, 2020; Immonen and Hämäläinen, 2022). TinyML focuses specifically on running machine learning models, particularly inference, directly on devices with severely limited processing power and memory. TinyML systems operate in the milliwatt range (often less than 1 watt), can operate completely offline without internet connectivity, and are designed for environments with severely limited resources (Ray, 2022). TinyML platforms (Arduino Nano 33 BLE Sense, ESP32 variants) cost \$10-50 and can operate on battery power for extended periods, making them viable for deployment in low-resource settings.

The system presented in this paper falls within

the Edge AI category rather than TinyML. Gemini Nano executes within the Chrome browser on consumer-class hardware (laptops, Chromebooks, and modern smartphones) rather than on microcontroller-class devices. This Edge AI approach shares TinyML’s central motivation of offline, on-device operation and directly addresses the connectivity challenges documented in the literature, where internet speeds vary from 211 Mbps in developed countries to just 1 Mbps in underserved regions. A longer-term goal of this line of work is to move toward TinyML-based capability, extending the offline grading approach demonstrated here to the ultra-low-power, microcontroller-class hardware described above.

The remainder of this paper is organized as follows. Section 2 reviews previous works on TinyML and edge AI in education. Section 3 details the system architecture and implementation. Section 4 describes the system’s implemented functionality. Section 5 discusses the implementation’s design tradeoffs and limitations. Section 6 discusses ethical considerations. Section 7 discusses directions for future work. Section 8 concludes the paper.

2 A Review of TinyML and Edge AI in Education

Recent work has explored the use of TinyML and Edge AI for a wide range of educational applications, spanning classroom monitoring, personalized learning, and student engagement.

Yahyati et al. (2025) surveyed TinyML deployments in education, including smart attendance systems, emotion-aware learning analytics, and inquiry-based science instruction, and found that most studies remain limited to small pilot deployments, rarely evaluate learning outcomes beyond system accuracy, and often do not consider the linguistic or cultural characteristics of the target classrooms. Andoniou (2026) proposed a design thinking framework addressing the same infrastructure gap from a hardware perspective, developing low cost offline AI and IoT solutions for communities with limited electricity and technical infrastructure.

Amo-Filva et al. (2024) reviewed privacy-preserving learning analytics and found that processing data closer to where it is collected reduces many of the privacy and security concerns associated with cloud based learning analytics. Several studies have put this principle into practice, though with varying degrees of success. Mukhit-

dinov et al. (2025) built a lightweight face recognition system that performs attendance tracking entirely on a low cost single board computer, keeping student identity data on device and achieving over 96% recognition accuracy. Valdes-Ramirez et al. (2026) developed a system for estimating classroom engagement from video using embedded GPU hardware, reducing bandwidth requirements from approximately 0.85 Mbps to about 0.1 Kbps of summarized indicators, though classroom logs were still uploaded to a central server for storage. Souhir et al. (2025) reported a similar partial pattern with an edge based facial emotion recognition system, where on device inference achieves 82-83% accuracy but recognized emotions are synchronized to a cloud database for teacher access. Jiang (2025), by contrast, kept processing fully local in a TinyML based English classroom quality assessment scheme, reporting improved accuracy, responsiveness, and resource efficiency over cloud based alternatives.

Labba et al. (2022) applied federated learning to predict student failure risk while keeping enrollment data distributed across clients rather than centralized. Using data from 671 students, their model achieved over 85% prediction accuracy by the eighth week of the academic term, performing within a few percentage points of an equivalent centralized model. Singha et al. (2025) built a system for offline lecture transcription and summarization on a low-cost single-board computer, using a fine-tuned speech-to-text model and an abstractive summarizer. Fine-tuning improved transcription accuracy for accented speech, though summary quality remained below that of a commercial cloud-based tool. Debbarma and Sinha (2022) explored a more exploratory use of TinyML, letting students photograph everyday objects and automatically retrieve related educational content using a compressed object detection model running on embedded hardware.

Singh and Singh (2025) built an edge-based gamification framework for engineering education that adjusted task difficulty using local processing, reporting improvements of approximately 51% in student engagement, 31% in performance, and a 57% reduction in cognitive load compared with conventional instruction. Amzoug et al. (2026) targeted accessibility directly, combining lightweight sign recognition with offline speech recognition and animation to support Deaf and Hard of Hearing students within a Moodle learning management

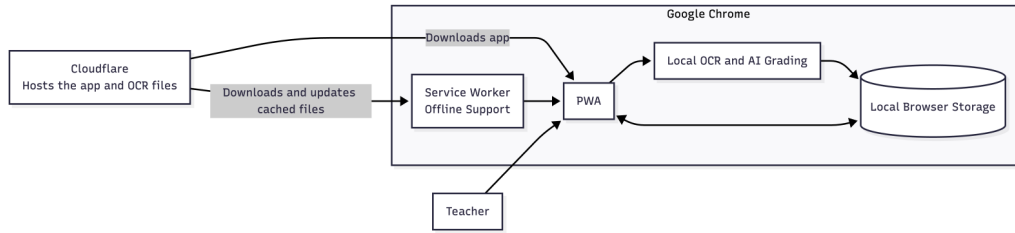


Figure 2: System architecture of the offline, browser-native grading pipeline. All processing occurs within the browser sandbox. The only network dependency is the one-time static asset fetch on first load.

system. Their system achieved 97% classification accuracy with 185 ms end-to-end latency on CPU hardware, and a pilot study reported positive user feedback alongside requests for additional regional sign vocabulary.

These studies show that on-device processing can support a wide range of educational tasks without cloud dependence, though several retain a residual cloud or connectivity dependency even while performing inference locally. Automated grading, however, has not been explored in the papers we identified. This study applies the same underlying principle, keeping inference local rather than in the cloud, to student assessment specifically, using an on-device LLM (Gemini Nano) rather than the smaller task-specific models used in the systems above. The use of a general-purpose language model allows the same system to interpret different natural-language questions and grading instructions without requiring a separately trained task-specific model, although this flexibility comes with greater computational and storage requirements.

3 System Architecture and Implementation

The system architecture is as represented in Fig. 2.

3.1 Technology Stack

The application is implemented as a Progressive Web Application (PWA) using React, enabling installation on the host operating system while remaining fully browser-resident. A service worker, generated by the Vite PWA plugin, caches the application shell and critical assets on the first load, enabling full offline operation thereafter. Assessment data, submissions, and extracted text persist locally in the browser’s IndexedDB store, requiring no backend database or server-side computation after initial deployment.

3.2 On-Device Language Model

Grading and feedback generation were performed using Google Gemini Nano, accessed through Chrome’s built-in Prompt API. Gemini Nano is a quantized language model embedded in Chrome’s inference engine. It executes entirely within the browser process with no network requests required after the model is downloaded on first use. The model is invoked with a structured prompt that includes the assessment rubric defined by the teacher, the OCR-extracted student response, and an instruction to return a numerical score with brief justification in JSON format. Because the Prompt API imposes context length constraints, responses exceeding the token limit are truncated before submission.

3.3 Optical Character Recognition

Text extraction from images of handwritten student work is performed by OCRS, an OCR engine compiled into WebAssembly. Running the OCRS in-browser avoids any network round-trip for image data and operates within the browser’s sandboxed execution environment, reinforcing the system’s privacy guarantees. The OCR pipeline accepts JPEG or PNG uploads, processes them synchronously, and returns plain text stored in IndexedDB along with the original image reference.

3.4 System Workflow

3.4.1 Assessment Creation

The teacher creates an assessment by entering a title, question prompt, and scoring rubric through the PWA interface. The rubric specifies the maximum point value and criteria for each score level. Assessment metadata are serialized and written to IndexedDB, generating a unique local assessment ID.

3.4.2 Submission Ingestion

The teacher photographs or scans the student work and uploads the image through the submission view, associating it with a student identifier and the target assessment. Upon uploading, the image is passed to the OCRS WebAssembly module. The extracted text is stored alongside the image reference in IndexedDB. The teacher may review and manually correct the OCR output before proceeding to grading, thereby providing a human checkpoint for recognition errors.

3.4.3 AI Grading

Grading is initiated per submission or in batch across all submissions for an assessment. For each submission, the system constructs a prompt that concatenates the rubric, the extracted student text, and a structured instruction requesting a JSON-formatted response containing a score and a one-to-two sentence justification. The prompt was submitted to Gemini Nano via the Prompt API. The returned JSON is parsed, and the score and feedback are written to IndexedDB and rendered in the grading dashboard.

4 Results

This section outlines the functionalities implemented in the current browser-native grading prototype. As no user studies or quantitative evaluations of OCR or grading accuracy have been performed, the results focus on describing the implemented workflow rather than measured performance.

4.1 Input: Assessment Definition and Student Submission

Teachers create assessments by specifying a name, optional context, and one or more questions (Fig. 3). Each question includes the question text, a reference answer, grading instructions, and a maximum point value. Questions can be added, removed, or reordered. The current prompt sent to the model incorporates the question, grading instructions, and maximum points but does not utilize the stored reference answer.

Figure 3: Assessment definition interface with per-question answers, grading instructions, and point values.

For each submission, the teacher inputs a student identifier and selects one or more captured response images. These images are processed locally in the browser by a bundled WebAssembly OCR engine with its detection and recognition models. Prior to grading, the teacher can review each image alongside its extracted text and make corrections as needed (Fig. 4).

Figure 4: Student submission images and locally extracted text.

4.2 Output: On-Device Grading and Feedback

Grading is conducted via the browser's built-in Language Model API, using English text for both input and output. The application verifies model

availability and displays download progress when necessary.

A distinct prompt is issued for each assessment question. This prompt includes the question text, its maximum point value, grading instructions, and the entirety of the corrected OCR text from all submitted pages. The model is instructed to respond with a JSON object containing the answer identified in the submission, points awarded (bounded between zero and the question maximum), and feedback. The overall grade percentage is computed from the total points awarded relative to the total points possible.

Figure 5: Per-question answers, points, feedback, and calculated overall grade.

Teachers may subsequently edit the extracted answer, feedback, and awarded points, and can mark questions as requiring review (Fig. 5). The dashboard flags a submission for review if any question is marked.

5 Discussion

All OCR processing, image handling, and grading operations occur entirely within the browser. Assessments, submission images, extracted texts, and grading results are persisted locally using RxDB built on Dexie/IndexedDB. The PWA service worker caches the application shell, OCR WebAssembly binary, and OCR model files, enabling offline use once these assets and the browser language model have been downloaded and cached.

OCR errors can affect grading accuracy, but teachers have the opportunity to correct OCR text prior to grading and to revise grading results afterward. Each question prompt receives the full extracted text from all pages, leaving the language

model responsible for identifying the relevant answer content.

The implementation applies `JSON.parse` directly to each constrained model response. In cases of prompting or parsing failure, errors are logged but no heuristic score extraction, retry logic, or automatic fallback to manual review is implemented. Consequently, a failed grading request may cause the grading process to stall without flagging the submission for review.

On the system used for this prototype, the Gemini Nano model weights occupied approximately 4 GB of local storage. At the 1 Mbps connection speeds reported for some target regions (Khowaja et al., 2024), downloading a model of this size could take several hours, representing a substantial one-time bandwidth barrier that the offline architecture does not eliminate.

While the prototype provides a complete end-to-end local workflow, claims regarding OCR accuracy, grading accuracy, reliability, context window limitations, or consistent operation across browsers and offline conditions require empirical validation and cannot be inferred from the implementation alone.

6 Ethical Considerations

While the proposed offline on-device architecture addresses challenges related to internet connectivity and data privacy, it also introduces several ethical considerations that should be acknowledged.

One important concern is the reliability of automated grading. Gemini Nano is a compact, quantized language model and is expected to produce less accurate scoring than larger cloud-hosted language models or human evaluators. Incorrect grading can directly affect student performance and academic outcomes. For this reason, the generated scores should be considered as recommendations rather than final decisions, with teacher reviews serving as an important safeguard.

Another consideration is the possibility of automation complacency. The proposed workflow assumes that teachers will review both the OCR output and AI-generated scores before finalizing grades. However, in practice, teachers may rely too heavily on automated results, particularly when grading a large number of submissions. This risk is compounded by a current implementation limitation. A failed grading request can stall silently without flagging the affected submission for review

(Section 5), so a teacher relying on the review-flag mechanism may not realize that a submission was never actually graded. Although the confidence-based review mechanism discussed in Section 7 is intended to reduce over-reliance on AI-generated outputs, it cannot completely prevent it and does not currently address this silent-failure case.

Potential bias and inconsistency in grading should also be considered. Compared to larger cloud-hosted language models, smaller on-device models have undergone more limited evaluations for fairness and robustness. As a result, grading performance may vary across different writing styles, dialects, non-native English responses, and unconventional but correct reasoning. This aspect has not been systematically evaluated in the current work and remains an important direction for future research.

Finally, although the proposed system removes the dependence on internet connectivity, it does not eliminate hardware accessibility challenges. Running Gemini Nano requires a compatible chrome-supported device with sufficient computational capability. Consequently, students or institutions with limited access to such devices may still be unable to benefit from the proposed approach. Therefore, offline execution improves accessibility but does not completely address the broader issue of equitable access to AI-assisted educational tools.

7 Future Work

Although the proposed system demonstrates the feasibility of offline AI-assisted grading, several opportunities remain to improve its performance, robustness, and applicability.

One of the most important areas of improvement is handwriting recognition. The current OCRS model was not trained on student handwriting datasets, and general-purpose OCR systems are well known to perform worse on handwriting than on printed text. Given this, OCR errors are likely to be a primary source of grading inaccuracies in practice, though this has not yet been empirically measured (Section 5). Fine-tuning the existing model or replacing it with a handwriting-specific OCR model, particularly one supporting low-resource scripts commonly used in target deployment regions, could significantly improve the overall grading accuracy.

Another useful extension is the addition of confidence-based review mechanisms. Rather than

treating every submission equally, the system could automatically flag low-confidence predictions for manual review while allowing teachers to quickly verify high-confidence cases. OCR confidence scores, response length, and the amount of extracted text relative to the Prompt API context window provide simple signals that could be used to implement such a confidence estimator while maintaining teacher oversight.

The current implementation also assumes English rubrics and student responses. Although Gemini Nano already supports multilingual inference, extending both the prompt templates and the user interface to support additional languages would make the system applicable to a broader range of educational settings, particularly in regions such as sub-Saharan Africa and South and Southeast Asia.

Another promising direction is the use of teacher feedback to continuously improve grading performance. Whenever teachers modify AI-generated scores, these corrections provide valuable information about grading behavior. A federated learning approach can aggregate these corrections across multiple deployments without sharing raw student responses, enabling rubric prompts and grading performance to improve over time while preserving data privacy.

The current evaluation focused primarily on short-answer factual responses. Extending the rubric design and prompt engineering framework to support more complex assessment types, including multi-step mathematical reasoning, diagram labeling, and analytical essays, would broaden the applicability of the proposed system to STEM education and higher-grade classroom settings.

Another important direction is migrating parts of this pipeline toward TinyML class hardware. The current system depends on Gemini Nano and browser-based execution, which require consumer devices such as laptops, Chromebooks, or modern smartphones. Achieving true TinyML-level deployment, on ultra-low-power microcontrollers in the milliwatt range, would require replacing the on-device LLM with substantially smaller, task-specific models for OCR and scoring, since current browser embedded LLMs are far too large to run on microcontroller-class hardware. This tradeoff between model generality and hardware constraints is a central open question for extending this work to the most severely resource-constrained environments described in Section 2.

8 Conclusion

We presented an offline, browser-native automated grading system that processes handwritten student submissions entirely on-device using WebAssembly OCR and Google’s Gemini Nano language model. The system requires no cloud connectivity and no transmission of student data beyond the local device, thus directly addressing the privacy, equity, and cost barriers that limit AI-assisted education in resource-constrained environments.

This work contributes a proof of concept that on-device browser AI has reached a sufficient capability to support educational automation in settings where cloud-dependent approaches remain inaccessible. While this work focuses on education, the underlying principles of offline, low-power, on-device inference generalize well beyond the classroom, from portable ECG/EEG-based diagnostics in resource-constrained healthcare settings (Hizem et al., 2025) to real-time agricultural monitoring (Samanta and Saha, 2026) and wildlife tracking in remote regions (Sabbella et al., 2022).

As browser-embedded model quality continues to improve and the range of supported hardware expands, the architecture described here offers a pathway toward equitable AI-assisted education aligned with SDG-4’s mandate for inclusive quality learning opportunities worldwide, echoing the broader potential of TinyML to advance the UN Sustainable Development Goals (Prakash et al., 2023).

Limitations

This work presents a proof-of-concept prototype without empirical validation of OCR accuracy, grading accuracy, or reliability across browsers and offline conditions. No user studies, comparison against human graders, or benchmarking against cloud-based grading systems have been conducted. The system currently supports English rubrics and responses only, and has been tested informally rather than through systematic evaluation. These limitations are discussed in more detail throughout Sections 5 and 6.

Acknowledgments

The authors thank Dante Ciolfi for his guidance on this work, which was conducted as part of the Computing Research Opportunities for Conservation and Sustainability coursework at the Georgia Institute of Technology. The authors used large lan-

guage models for image generation and language editing.

References

- Daniel Amo-Filva, David Fonseca, Francisco José García-Peñalvo, Marc Alier Forment, Maria José Casany Guerrero, and Guillem Godoy. 2024. [Exploring the landscape of learning analytics privacy in fog and edge computing: A systematic literature review](#). *Computers in Human Behavior*, 158:108303.
- Youssef Amzoug, Chaimae Elfakir, Khadija Hicham, Soufiane Hamida, Soumia Ziti, and Bouchaib Cheradi. 2026. [Lightweight bidirectional sign language translation for inclusive education on edge devices](#). In *2026 6th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, pages 1–7, FEZ, Morocco. IEEE.
- Constantine Andoniou. 2026. [Micro-solutions to mega-problems: A design-thinking framework for affordable offline AI-IoT access in low-resource contexts](#). In *2026 International Conference on Artificial Intelligence, Systems, and Emerging Technologies (ICAISSET)*, pages 1–8, Cairo, Egypt. IEEE.
- Anusree Debbarma and Sharad Sinha. 2022. [TinyML based physical computing for education](#). In *2022 IEEE 19th India Council International Conference (INDICON)*, pages 1–4, Kochi, India. IEEE.
- European Parliament and Council of the European Union. 2016. Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (general data protection regulation). Official Journal of the European Union, L 119, 4.5.2016, pp. 1–88. Available at <https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng>.
- GSMA Intelligence. 2024. The state of mobile internet connectivity 2024. Technical report, GSMA, London, UK.
- Moez Hizem, Mohamed Ould-Elhassen Aoueileyine, Samir Brahim Belhaouari, Abdelfatteh EL Omri, and Ridha Bouallegue. 2025. [Sustainable e-health: Energy-efficient tiny AI for epileptic seizure detection via EEG](#). *Biomedical Engineering and Computational Biology*, 16:11795972241283101.
- Riku Immonen and Timo Hämäläinen. 2022. [Tiny machine learning for resource-constrained microcontrollers](#). *Journal of Sensors*, 2022:7437023.
- International Telecommunication Union. 2023. Measuring digital development: Facts and figures 2023. Technical report, ITU, Geneva, Switzerland.
- Seiji Isotani, Ig Ibert Bittencourt, Geiser C. Chalco, Diego Dermeval, and Rafael F. Mello. 2023. [AIED unplugged: Leapfrogging the digital divide to reach](#)

- the underserved. In *International Conference on Artificial Intelligence in Education (AIED 2023)*, volume 1831 of *CCIS*, pages 772–779. Springer, Cham.
- Shuang Jiang. 2025. TinyML based edge intelligent English classroom quality assessment scheme. *Internet Technology Letters*, 8(4):e70072.
- Sunder Ali Khowaja, Parus Khuwaja, Kapal Dev, Weizheng Wang, and Lewis Nkenyereye. 2024. ChatGPT needs SPADE (sustainability, PrivAcy, digital divide, and ethics) evaluation: A review. *Cognitive Computation*, 16:2528–2550.
- Chahrazed Labba, Rabie Ben Atitallah, and Anne Boyer. 2022. Combining artificial intelligence and edge computing to reshape distance education (case study: K-12 learners). In Maria Mercedes Rodrigo, Noburu Matsuda, Alexandra I. Cristea, and Vania Dimitrova, editors, *Artificial Intelligence in Education*, volume 13355, pages 218–230. Springer International Publishing, Cham.
- Otabek Mukhitdinov, Poyan Bakirov, Nilufar Atashikova, Saida Makhkamova, I. B. Sapaev, Zulfizar Karimova, Mastura Jo'rayeva, and Olima Kholmurodova. 2025. A lightweight AI-based access control system for IoT-integrated smart classrooms. *Journal of Internet Services and Information Security*, 15(2):30–42.
- NVIDIA Corporation. 2024. NVIDIA Jetson Orin for next-gen robotics. NVIDIA. Available at <https://www.nvidia.com/en-us/autonomous-machine-s/embedded-systems/jetson-orin/>.
- Francesc Pedro, Miguel Subosa, Axel Rivas, and Paula Valverde. 2019. Artificial intelligence in education: Challenges and opportunities for sustainable development. Technical report, UNESCO Working Papers on Education Policy, No. 7, Paris, France.
- Shvetank Prakash, Matthew Stewart, Colby Banbury, Mark Mazumder, Pete Warden, Brian Plancher, and Vijay Janapa Reddi. 2023. Is TinyML sustainable? assessing the environmental impacts of machine learning on microcontrollers. *Communications of the ACM*.
- Partha Pratim Ray. 2022. A review on TinyML: State-of-the-art and prospects. *Journal of King Saud University - Computer and Information Sciences*, 34(4):1595–1623.
- Harshavardhan Reddy Sabbella, Abhishek Ramdas Nair, Vishnu Gumme, Sathvik Sridhar Yadav, Shantanu Chakrabarty, and Chetan Singh Thakur. 2022. An always-on tinyML acoustic classifier for ecological applications. In *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2393–2396. IEEE.
- D. Saisanthiya, G. Malarselvi, G. Jessy Sujana, V. Saidulu, and K. Rajesh Kumar. 2026. Federated and privacy-preserving adaptive learning framework for AI-driven mobile education platforms. *International Journal of Interactive Mobile Technologies (iJIM)*, 20(12):124–139.
- Riya Samanta and Bidyut Saha. 2026. Affordable precision agriculture: A deployment-oriented review of low-cost, low-power edge AI and TinyML for resource-constrained farming systems. *arXiv preprint arXiv:2603.15085*.
- Ramon Sanchez-Iborra and Antonio F. Skarmeta. 2020. TinyML-enabled frugal smart objects: Challenges and opportunities. *IEEE Circuits and Systems Magazine*, 20(3):4–18.
- Kiran Deep Singh and Prabh Deep Singh. 2025. Edge-AI driven gamification in engineering education for improving student engagement. *Computer Applications in Engineering Education*, 33(5):e70076.
- Omsubhra Singha, Wanbanker Khongbuh, Preetisudha Meher, Priyam Gupta, and Roshan Sahani. 2025. Self-contained edge AI for offline lecture summarization: A Raspberry Pi-based system with integrated secure FTP. In *2025 IEEE International Conference on Electrical, Electronics, Communication and Computers (ELEXCOM)*, pages 1–6, Dhanbad, India. IEEE.
- Boufrikha Souhir, Bouaafia Soulef, Olfa Ben Ahmed, Hajjaji Mohamed Ali, Mtibaa Abdelatif, and Al Jaam Jihad. 2025. Edge AI framework for real-time facial emotion recognition toward student engagement tracking. In *2025 International Conference on Computer and Applications (ICCA)*, pages 1–6, Bahrain, Bahrain. IEEE.
- UNESCO. 2023. UNESCO survey: Less than 10% of schools and universities have formal guidance on AI. UNESCO. Prepared for the Ministerial Roundtable on Generative AI and Education, 25 May 2023. Available at <https://www.unesco.org/en/articles/unesco-survey-less-10-schools-and-universities-have-formal-guidance-ai>.
- United States Congress. 1974. Family educational rights and privacy act of 1974. 20 U.S.C. §1232g; 34 CFR Part 99. U.S. Department of Education. Available at <https://studentprivacy.ed.gov/faq/what-at-ferpa>.
- Danilo Valdes-Ramirez, Viktor Sobrino Caamal, José Ignacio Huertas Cardozo, and Genaro Zavala. 2026. Edge computing model for real-time classroom behavior analysis in face-to-face lessons. In Goreti Marreiros, Guillermo Hernández, Anne-Cecile Caron, Tânia Rocha, Jesús Ángel Román, Aurora González-Vidal, Diogo Martinho, Rodrigo Gil-Merino, and Javier Parra, editors, *Distributed Computing and Artificial Intelligence, Special Sessions II, 22nd International Conference*, volume 1630, pages 495–505. Springer Nature Switzerland, Cham.
- Chaymae Yahyati, Ismail Lamaakal, Khalid El Makkaoui, Yassine Maleh, and Ibrahim Ouahbi.

2025. [A survey on TinyML applications in education](#). In *2025 International Conference on Electrical Systems & Automation (ICESA)*, pages 1–5, Troyes, France. IEEE.