SIIM CAIMI26
Conference on Artificial Intelligence in Medical Imaging
OCT 26-27 | University of Pennsylvania, Philadelphia, PA

Radaptive: From Case Exposure to Actionable Feedback in Radiology Training

Ebubechukwu D. Enwerem, University of Pennsylvania; Shawn Lyo, MD; Tessa Cook, MD, PhD, CIIP, FSIIM; Satvik Tripathi

Briefly describe what attendees will see during your demo.

Attendees will see a trainee review a de-identified chest CT in OHIF, dictate or type a report, and receive structured AI-generated feedback. We will then show the Explorer visualizing discrepancy patterns and generating targeted teaching insights from those patterns.

Problem & Motivation: What clinical or operational problem does your MVP address?

Radiology residents do not encounter the same mix of cases during training. This can leave gaps in exposure to important diagnoses and limited opportunities to practice interpreting and reporting high-yield “must-see” cases. Feedback is also inconsistent: it may arrive well after a case, focus on wording rather than the underlying interpretation, and remain scattered across individual readouts instead of becoming a longitudinal learning record. Radaptive was built to connect three activities that are usually separated: reviewing the images, creating a radiology report, and receiving structured feedback grounded in an expert reference report. The goal is to supplement faculty teaching with repeatable case-based practice and make recurring learning needs visible to both trainees and educators. It is an educational tool, not an autonomous diagnostic system or replacement for attending review.

What You Built: Describe your MVP, tool, or workflow in plain language

Radaptive is a working web-based radiology training prototype with two workflows. In the image-based workflow, an educator imports a de-identified DICOM study and its expert reference report. A trainee reviews the complete study in an embedded OHIF viewer and types or dictates separate Findings and Impression sections. The reference report stays hidden until the trainee submits the draft. The original text-only workflow can also compare an existing trainee draft with an attending report. After submission, Radaptive compares the two reports through a five-stage language-model pipeline. It produces structured discrepancies, severity classifications, implicated pathology and anatomy, error types, and an inventory of findings in the expert report. Results can be reviewed by case and aggregated into trainee profiles, trends, focus areas, and interactive views of discrepancy patterns. The system records analysis jobs and intermediate stage results so failures can be inspected and retried rather than silently discarded.

Tech Stack & Development Approach: What tools, frameworks, or methods did you use to build it?

Radaptive is written primarily in Python 3.11 using FastAPI, Pydantic, SQLAlchemy, Alembic, Jinja2, HTMX, and Uvicorn. SQLite and local disk storage support the MVP; PostgreSQL is an available deployment option. DICOM ingestion uses pydicom, and a pinned local build of OHIF provides image viewing without requiring Orthanc or a PACS. The default language model is medgemma1.5:4b-it-q8_0, stored and executed locally through Ollama. Each case runs through a five-stage ensemble and adjudication pipeline with JSON validation, database checkpoints, retry handling, and audit metadata. Dictation uses a local whisper.cpp large-v3-turbo model with FFmpeg normalization; audio is reviewed before insertion and is not retained. The architecture uses small service, adapter, and protocol modules so the DICOM backend, viewer, speech engine, model provider, and database can be replaced independently. Development has been iterative and AI-assisted using Codex, with manual architecture decisions, focused automated tests, and repeated end-to-end demo runs.

Validation & Evidence: Have you done any informal testing or validation? If so, what did you find?
Current validation is technical and formative, not yet educational or clinical. We have completed repeated end-to-end walkthroughs using de-identified chest CT cases. These confirmed that a user can import a multi-slice DICOM study and reference report, review the examination in OHIF, dictate or type a trainee report, submit it, and receive structured discrepancy results and trainee-level analytics. Informal testing also exposed important limitations. Model-generated feedback can inconsistently distinguish clinically meaningful discrepancies from acceptable reporting variation, reinforcing the need for faculty adjudication. Automated tests verify workflow integrity but do not establish feedback accuracy or educational value. The next validation phase will compare Radaptive's discrepancies with independent faculty assessments, followed by resident usability testing. A prospective pilot would evaluate report completeness, discrepancy patterns over time, user confidence, engagement, and faculty review burden

Feedback You're Seeking: What specific feedback, help, or collaboration are you hoping to get from the CAIMI26 community?
We want help determining whether Radaptive's discrepancy categories and explanations are educationally useful and how to distinguish clinically meaningful errors from acceptable differences in reporting style. We are particularly interested in faculty approaches to adjudicating model-generated feedback and building a high-quality, curriculum-aligned case library. We also seek collaborators or clinical champions who can advise on a prospective resident pilot: appropriate outcomes, number and diversity of cases, baseline comparisons, faculty review burden, and measures of report quality, usability, confidence, and engagement. Imaging-informatics feedback on institutional deployment, authentication, data governance, and eventual PACS/reporting-system integration would also be valuable.

Known Limitations & Honest Failures: What isn't working yet, or what have you already tried that failed?
Radaptive's primary limitation is that its feedback has not yet been systematically adjudicated by faculty radiologists or evaluated with residents. The system may label acceptable reporting variation as an error, miss a clinically meaningful discrepancy, or assign an inappropriate severity. It also treats the attending report as the reference standard, although an individual report is not infallible ground truth. Faculty oversight therefore remains necessary. The current case library is small and designed for demonstration rather than balanced curricular coverage. We have not shown that using Radaptive improves report quality, knowledge retention, confidence, or clinical performance. Local analysis may also take several minutes, which could affect engagement during repeated practice. Dictation can introduce clinically important errors involving negation, laterality, anatomy, or measurements, so trainees must review all transcribed text. Finally, the MVP uses manual case import and has not undergone institutional security or HIPAA review. Integration with PACS, enterprise authentication, and clinical reporting systems would require institutional governance and access to vendor-supported interfaces.

Potential for Impact: If this MVP were fully developed and deployed, who would benefit and how?
If fully developed and validated, Radaptive could give trainees more consistent exposure to essential diagnoses, realistic reporting practice, immediate case-level feedback, and a longitudinal view of recurring weaknesses. Faculty could use aggregated discrepancy patterns to target teaching time, select remediation cases, and identify curriculum gaps that are difficult to see during individual readouts. Residency programs could provide a more standardized educational experience while still keeping expert faculty in control of case selection and feedback adjudication. Imaging-informatics teams could use the same structured data to study reporting patterns and evaluate educational interventions. Patients may benefit indirectly if better targeted practice improves report completeness and recognition of important findings, but that outcome must be demonstrated through prospective evaluation rather than assumed from the MVP.

GitHub Repo:
https://github.com/shawnktl/radgnosis/tree/feat/RadAdptiveForDemo