A Unified Agentic Framework for Medical VLMs: Case Retrieval, Reporting, and Visual Question Answering

Andy Xu Wang *

Tsinghua International School, Beijing, China

* Corresponding Author Email: andy.wang.2026@this.edu.cn

Abstract. Clinical decision-making often requires physicians to iteratively generate hypotheses, gather multimodal evidence, and refine diagnoses under conditions of time pressure and information overload. While medical vision–language models (VLMs) have shown promise in visual question answering, report generation, and image-based diagnosis, most remain standalone tools without integration into broader diagnostic workflows. In this work, we introduce Casidence, an agentic medical VLM framework that unifies evidence retrieval, similar case retrieval, and conversational AI into a single decision-support platform. At its core, Casidence incorporates a fine-tuned 3D medical VLM trained on the CT-RATE dataset to achieve state-of-the-art performance on volumetric imaging tasks. To enable robust similar case retrieval, we propose a novel Query Auto Encoder (QAE) that disentangles semantic medical content from surface linguistic variation, producing compact embeddings aligned across paraphrased reports. Together, these components allow Casidence to operationalize evidence-based reasoning: planning and executing tool-augmented workflows, curating structured evidences, and generating auditable diagnostic outputs. Quantitative and qualitative evaluations demonstrate that Casidence improves planning transparency, retrieval fidelity, and report quality over strong baselines. By grounding model outputs in clinical evidence and supporting iterative human–AI collaboration, Casidence represents a step toward trustworthy, workflow integrated medical agentic systems.

Keywords: Medical Agentic System, Human-ai collaboration. evidence-based reasoning, Medical Visual Question Answering (VQA), Similar Case Retrieval, Contrastive Learning.

1. Introduction

Clinical diagnosis is a high-stakes process in which physicians integrate a patient's signs, symptoms, and contextual information to determine the most plausible underlying condition and an appropriate course of action. In daily practice, this process unfolds as an iterative cycle of hypothesis generation, targeted evidence gathering, and hypothesis refinement. Physicians formulate differential diagnoses [1], acquire complementary evidence through history, physical examination, laboratory tests, and instrumental studies such as medical imaging, and revise their working hypotheses accordingly. Among these tools, medical imaging plays a central role across specialties: radiography and computed tomography (CT) for structural assessment, magnetic resonance imaging (MRI) for soft-tissue and functional characterization, ultrasound for real-time bedside evaluation, and nuclear imaging for perfusion and metabolic insights. These modalities provide rich, multi-scale features that clinicians interrogate to localize disease, stage severity, and anticipate complications.

Recent advances in medical vision–language models (VLMs) have demonstrated the potential to bridge visual and textual modalities at scale. These models support applications such as medical visual question answering, radiology report generation, and automated diagnostic reasoning. Typically, general-purpose VLMs (e.g., LLaVA [2]) are fine-tuned on domain-specific datasets of paired radiology images and reports, enabling visual encoders to align with large language models and capture clinical semantics. Such systems offer clear strengths: they reduce reporting burden, improve accessibility of medical knowledge, and unify multimodal inputs.

Yet, several important limitations remain. First, most VLMs are developed and assessed as standalone models, lacking integration with other essential components for the comprehensive decision-making support, such as similar case retrieval, evidence retrieval, report generation, and interactive dialogue. Throughout this medical reasoning process, clinicians often consult prior cases

via personal memory, institutional archives, or curated literature to calibrate expectations, recognize atypical presentations, and contextualize ambiguous findings. However, as data volume grows and clinical workflows become increasingly time-pressured, human memory and manual retrieval are insufficient to meet demand, which highlights the unmet need for intelligent case-retrieval systems that can adapt searches to a physician's specific context and information needs, thereby providing timely, personalized reference cases to support clinical decision-making.

Second, existing retrieval methods struggle with robustness and clinical fidelity. Challenges such as negation, temporality, abbreviations, and the "semantic gap" between visual similarity and clinical equivalence continue to limit their reliability across diverse settings. These gaps underscore that building clinically useful systems requires not only strong perception models, but also workflow integration, retrieval precision, and support for real-time reasoning.

In this work, we present Casidence: an agentic medical vision–language system that moves beyond isolated model performance to provide integrated clinical decision support. Our system combines three complementary capabilities: (1) Evidence retrieval from guidelines, pathways, and literature tailored to a patient's presentation, ensuring best practices are available at the point of care; (2) Similar-case retrieval (SCR) that surfaces prior patients or imaging studies with clinically aligned presentations and outcomes, enabled by a medical semantic encoder designed to improve fidelity beyond purely visual or embedding-based similarity match; (3) Conversational AI assistance that functions as a retrieval-augmented copilot, engaging in free-text dialogue to answer clinical questions, generate draft notes, and propose next steps with clinician oversight. At the core of Casidence is a newly developed state-of-the-art foundation model for radiology, which surpasses prior models through extensive supervised fine-tuning and instruction-tuning on diverse, clinically curated datasets. This foundation model provides robust multimodal understanding of medical imaging and serves as the backbone for Casidence, enabling accurate interpretation of patient scans while seamlessly supporting the system's retrieval and conversational components.

Together, these components form an end-to-end decision-support platform designed to help physicians manage diagnostic complexity under conditions of time pressure and information overload. By grounding technical advances in real clinical workflows, our approach emphasizes not only performance gains but also practical utility in enhancing safety, consistency, and efficiency of care.

2. Related Work

2.1. Foundation Visual Language Models

Modern vision–language models (VLMs) are trained on massive datasets of image–text pairs (often hundreds of millions of web-scraped images with captions), enabling them to learn aligned visual and textual representations for robust multimodal understanding. A wide variety of VLM designs have emerged – including (1) contrastively-trained dual-encoder models like OpenAI's *CLIP* [3] and Google's *ALIGN* [4], which jointly embed images and text); (2) cross-modal Transformer [5] architectures that fuse image and text features (e.g. DeepMind's *Flamingo* [6] or Salesforce's *BLIP-2* [7]) via learned visual adapter modules, and multimodal LLM systems such as OpenAI's *GPT-4 Vision* [8], Google's *Gemini* [9], or the open-source *LLaVA* [10] that extend text-only large language models with image inputs. These popular models can caption images and answer visual questions about them, as well as recognize objects and even read text within images, demonstrating remarkable cross-modal reasoning and understanding capabilities.

2.2. Medical Vision-Language Models

Medical vision-language models (VLMs) have been developed to interpret medical images and produce textual outputs for tasks such as visual question answering (VQA) [11], automated report generation [12], and image-based diagnosis [13]. These models combine a visual encoder (processing radiology images like X-rays, CT, or MRI scans) with a large language model, enabling a unified understanding of visual and textual data. Early medical VLMs often built upon general-purpose

vision-language foundations to leverage their pre-trained knowledge. For example, LLaVA-Med [2] adapts the open-source LLaVA model to the biomedical domain, pairing a CLIP-based visual encoder with an LLM to answer medical image questions in a conversational manner. Similarly, Google's Med-PaLM [14] is a multimodal model based on PaLM-E [15] that handles both images and text; it achieved state-ofthe-art performance on diverse medical benchmarks (e.g. radiology report generation and medical QA) by jointly encoding clinical text, radiographs, and even genomic data in one model. These early models largely focused on 2D radiology images (like chest X-rays or dermatology photos) and used visual instruction tuning to align the LLM with medical imaging concepts. Notably, E3D-GPT [16] introduced a 3Daware VLM to tackle volumetric scans: it uses a self-supervised 3D image encoder and masked feature modeling to better represent CT volumes, which enhanced image-text alignment and led to improved results in CT report generation and VQA.

Recent VLM models push further in unifying modalities and specializing tasks in the radiology domain. RadFM [17] exemplifies a generalist radiology foundation model trained on a massive MedMD dataset of 16 million images, covering both 2D X-rays and 3D scans. RadFM's architecture interleaves vision and language, allowing it to accept multi-image inputs (e.g. series of slices) and generate free-text descriptions or answers. It excels across a spectrum of tasks and outperformed prior multimodal models (including other open VLMs and even a GPT-4V baseline) on a dedicated radiology benchmark. Another work, CT-RATE [18], provides the first large-scale pairing of 3D chest CT volumes (25,692 scans; 50,188 reconstructions) with corresponding radiology reports and multi-abnormality labels, enabling robust training and evaluation of volumetric VLMs. Based on CT-RATE, CT-CHAT [18] integrates a CT-specific contrastive vision encoder (CT-CLIP) with a large language model and is finetuned on more than 2.7 million QA pairs from CT-RATE, yielding state-of-theart interactive VQA and report-style responses on 3D chest CT. M3D-LaMed [19] introduced a large-scale 3D radiology dataset (M3D-Data) and a multi-modal 3D VLM that directly processes volumetric scans. By combining a CLIP-pretrained 3D visual encoder with a spatial pooling module, M3D-LaMed can handle whole CT/MRI volumes and was shown to achieve state-of-the-art results on various tasks including image–text conversion, radiology report generation, VQA, segmentation and lesion localization tasks. However, M3D-LaMed's design highlighted challenges like high computational cost for large 3D inputs and suboptimal cross-modal fusion.

The latest research has further improved efficiency and accuracy for 3D vision language reasoning. Med3DVLM [20], for instance, builds on M3D-LaMed's foundation but introduces a more efficient 3D encoder (using decomposed convolutions) and a refined alignment mechanism to better fuse image features with text. This leads to substantial performance gains: Med3DVLM dramatically outperforms M3D-LaMed on unified benchmarks, nearly quadrupling image–text retrieval accuracy (61.0% vs 19.1% R@1) and greatly improving report generation quality.

2.3. Medical Patient Retrieval

Keyword-based (lexical) information retrieval has long been the foundation of chart review and cohort discovery in EHRs. Classic inverted-index methods such as BM25 [21] remain widely deployed, with systems like EMERSE [22] providing enterprise search layers over free-text notes across academic medical centers [23]. Despite their utility, these approaches face persistent challenges [24, 25], including synonym and abbreviation mismatches, limited handling of negation and temporality, section or context insensitivity, heavy dependence on user query literacy, etc.

Vector-based retrieval, on the contrary, encodes cases (notes, images, or multimodal bundles) into dense vectors and retrieves neighbors via Maximum Inner Product Search, typically using approximate-NN backends such as FAISS [26], HNSW [27], or ScaNN [28] for scalability. Within this family, bi-encoder dense retrievers map clinical notes and queries with transformer encoders (e.g., ClinicalBERT [29]). Lateinteraction models such as ColBERT [30] preserve token-level vectors and apply MaxSim scoring to better capture negation and section context in long notes. Beyond text, image-based content retrieval leverages CNN or self-supervised embeddings for query-by-image, exemplified by SISH [31] in pathology whole-slide imaging. Crossand multimodal dual encoders

further align reports and images in shared spaces through CLIP-style contrastive training, enabling cross-modal retrieval [32].

2.4. Medical Agentic System

Recent advances in LLM-based agents have demonstrated impressive capabilities across diverse domains, including complex decision-making [33], opening new avenues for automating tasks and augmenting human expertise. Large language model–driven agents are beginning to play a meaningful role in healthcare [34], with studies reporting benefits in various applications such as diagnostic decision support [35], medical education [36], patient-provider communication [37], EHRbased application [38], etc. By integrating domain-specific resources such as clinical knowledge graphs, guidelines, and electronic health records, LLM-based agentic systems are being developed to use predefined tools [39,40] to navigate nuanced clinical information, integrate heterogeneous data sources, and provide contextually relevant outputs that complement clinician decision making. However, the current generation of medical agentic systems remains constrained in important ways. Most implementations rely on a limited set of external tools and resources, which restricts their ability to handle the full spectrum of clinical tasks. Moreover, clinical validation is still scarce and underrepresented in the literature of medical agentic system. Addressing these limitations will be critical for translating prototype systems into realistic components of routine healthcare practice.

3. Casidence: Case-Evidence Agentic System

Casidence operationalizes evidence–based diagnosis as an iterative, tool-augmented workflow that cycles between *case understanding*, *targeted evidence acquisition*, and *synthesis*. Given a user prompt (task), available inputs (e.g., clinical text, 2D/3D images), and any prior evidence, Casidence plans and executes a sequence of analysis steps using a modular toolbox, organizes results into verifiable evidence statements, and issues a diagnosis with an explicit confidence and rationale. The system is designed for medical search and evidence curation, supports expanding toolchains, and encourages human–AI collaboration via editable plans and user provided evidence. The high-level dataflow is shown in Figure 1. Casidence couples research-aware planning with multimodal tool use, producing auditable evidence trails and enabling test-time scaling law [41]: When the system's confidence is insufficient, it allocates additional compute to deeper investigation (more retrieval, more analysis steps) until criteria are met or failure modes are surfaced for human review. The toolbox is extensible; new tools register capabilities and I/O contracts and become immediately plannable. Human–AI collaboration is built in: users can edit plans, inject evidence, and re-run partial pipelines.

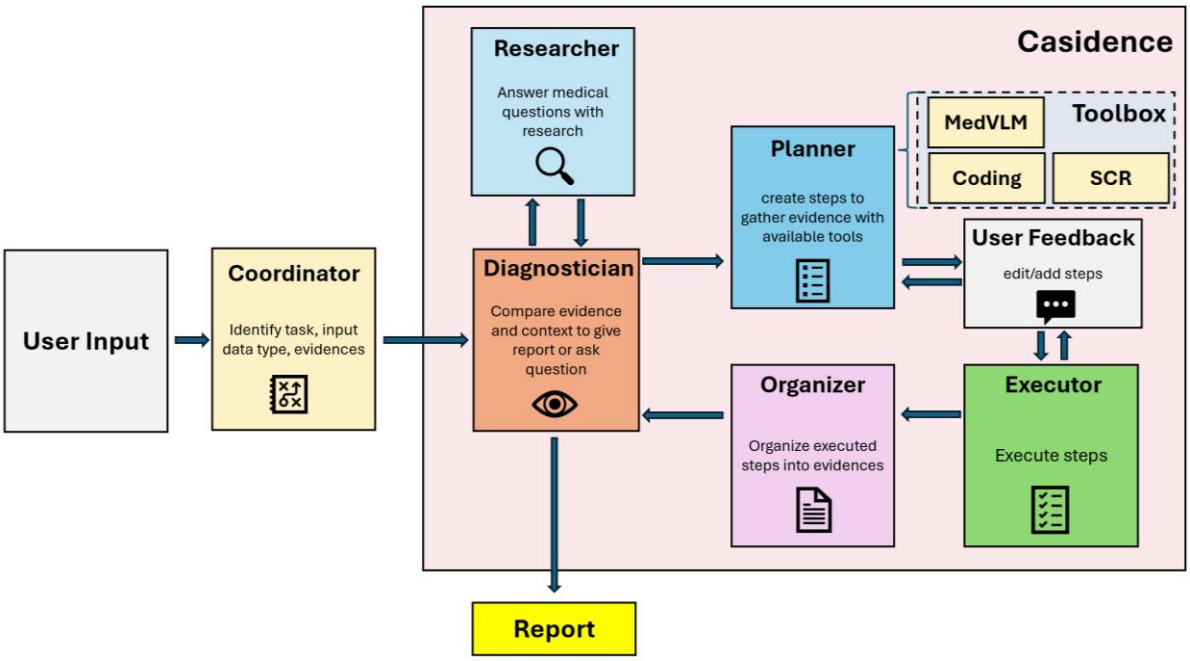

Figure 1. Overview of Casidence agentic architecture and dataflow. User input is parsed by the *coordinator*, triaged by the *Diagnostician*, and—when additional evidence is needed—routed through *Researcher* → *Planner* → *Executor* → *Organizer*. Outputs are converted into atomic evidences that return to the *Diagnostician* to produce the final report. Plans are editable via *User Feedback* and use an extensible *Toolbox* (e.g., MedVLM, Coding, SCR).

3.1. Inputs and outputs

Inputs include structured or free-text case descriptions; 2D images (e.g., PNG/JPEG), 3D volumes (e.g., NIfTI), and optional prior evidence. Outputs include: (i) a diagnosis or focused assessment, (ii) supporting and contradicting evidence lists, (iii) missing-evidence requirements and next-step recommendations, and (iv) a confidence score.

3.2. Agentic Architecture

Overview. Figure 1 depicts the agentic architecture and role interactions. Each role has a single responsibility and a typed interface, so components can be swapped or improved independently. The roles communicate through two canonical artifacts: *Context Data* (research Q&A used as contextual grounding) and *Evidences* (atomic, standalone evidence statements). Provenance is preserved from tool outputs to evidence to diagnosis. All agents are powered by the OpenAI's GPT-5 1model in order to ensure the best performance, and all patients are de-identified to protect privacy.

Roles and responsibilities

- Coordinator parses the user prompt into *task*, *input_data_type*, and *available_evidences*. It normalizes modalities (text, 2D, 3D) and surfaces constraints to downstream components.
- Diagnostician acts like a physician's reasoning hub, deciding when current evidence is enough or when more investigation is needed. In *report mode*, it compares current evidence with research context to determine if a diagnosis can be issued; it always returns a specific assessment, supporting/contradicting evidence, missing evidence, and a scalar confidence. In *research mode*, it formulates targeted research questions to close the gaps.
- Researcher brings in external knowledge (literature, guidelines, prior cases), reducing the risk of blind spots. It learns to answer the Diagnostician's questions using a comprehensive researcher-pipeline over medical sources. The results are stored as Q&A pairs with formatting suitable for reuse (bulleted, numbered, or paragraph) and accumulate over iterations.
- Planner synthesizes the task, evidence and research context into a *structured plan*: an ordered list of tool invocation steps with declared dependencies, expected inputs/outputs and qualitative or

quantitative intent. Plans are user editable to ensure consistency with the user expectation and can be generated without execution for review.

- Executor runs the plan stepwise, enforces dependencies, and captures raw output (e.g., VLM answers, measurements, segmentations) with metadata.
- Organizer converts raw outputs into atomic evidence statements, suitable for direct comparison and aggregation. It supports both qualitative assertions (e.g., "cup-to-disc ratio enlarged") and quantitative metrics, and attaches provenance to each statement.

Toolbox and expandability Casidence maintain a pluggable toolbox, which includes: a *3D Medical VLM* for volumetric image question answering; a *2D general VLM* and a *2D medical VLM* (e.g., LLaVA-Med) for 2D imaging; a *coding agent* for quantitative analysis and ad-hoc computation; and segmentation components (e.g., MedSAM [42]) for producing measurements and ROIs. Tools declare modalities, I/O schemas, and side effects, enabling the Planner to decompose them into executable graphs. New tools can be added by registering their capabilities; no changes to upstream roles are required.

Iteration and test-time scaling Casidence employs a diagnose–plan–execute–organize loop with early exit. If the Diagnostician's confidence is below a threshold or required evidence is missing, the Researcher and Planner produce additional queries and steps; the loop repeats until the confidence is adequate or the system converges to irreducible uncertainty, which is presented as actionable missing-evidence requirements. This realizes test-time compute scaling in a clinically grounded way: More compute buys more targeted evidence, rather than redundant sampling.

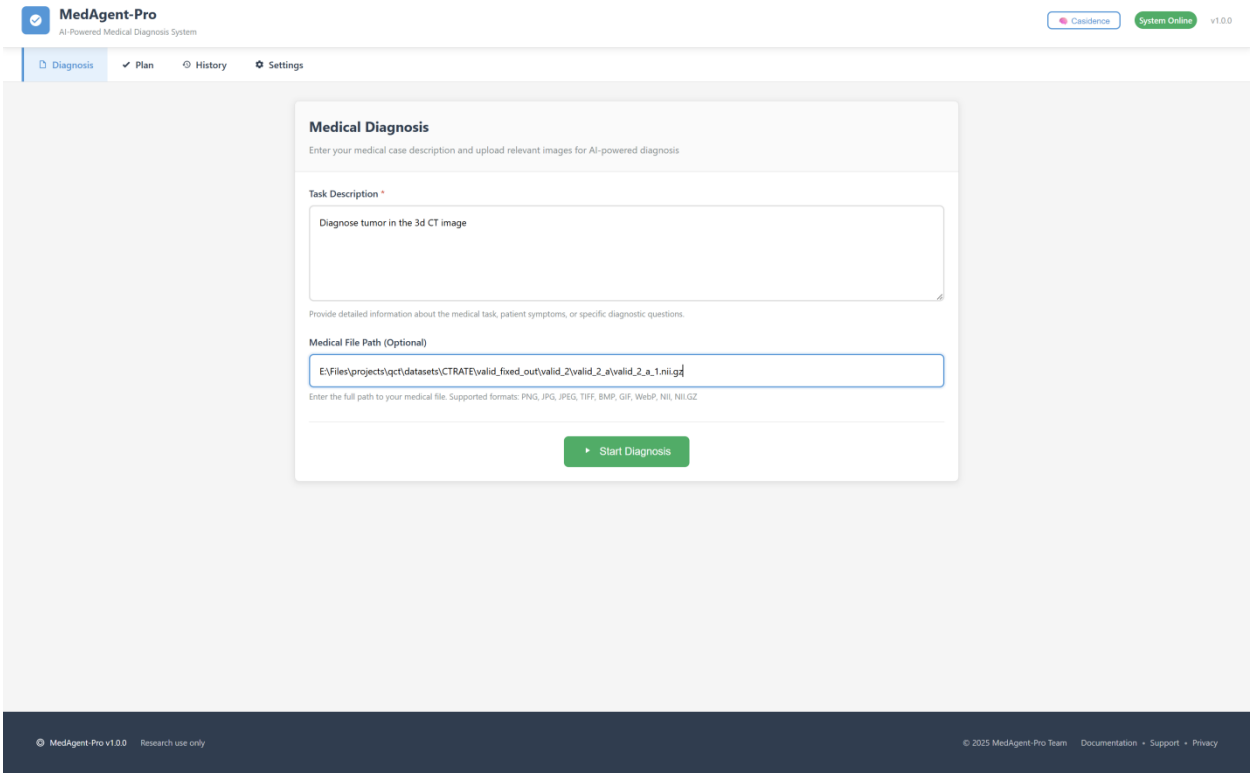

Figure 2. Demo of Casidence webpage UI. User interface for entering task and uploading file.

3.2.1. User Interfaces and Human–AI Collaboration

We provide three user interface surfaces (see Figure 2, 3, 4): *(i) Initial*—where users specify the task, upload data, and supply seed evidence; *(ii) Plan*—which presents the generated workflow for inspection and editing (e.g., adding, removing, or reordering steps, selecting alternative tools, or injecting user-provided evidence); and *(iii) Execution*—which enables step-by-step execution, inspection of raw outputs, review of organized evidence, and targeted follow-up queries. All interactions are tracked for provenance, and the system supports partial re-execution while preserving context.

3.3. Algorithmic Specification

Data structures *ContextData* is a list of ⟨question, answer, render_format⟩ triples. *Evidences* is a set of atomic statements with fields: source, modality, value (text or numeric), and confidence.

Main loop Let Me be the current evidence set and C the context store.

1. Triage: Diagnostician (*task*, E, C) → {can_diagnose, assessment, supporting, contradicting, missing, confidence, recommendation}.
2. If can_diagnose or confidence ≥ τ: emit report and stop.
3. Else Researcher generates questions from {task, E}; update C with answers.
4. Planner (*task*, E, C, toolbox) → plan with typed steps.
5. Executor executes plan; Organizer extracts new evidence Δε; update ε ← ε ∪ Δε.
6. Repeat from Step 1 (bounded by user constraints or saturation).

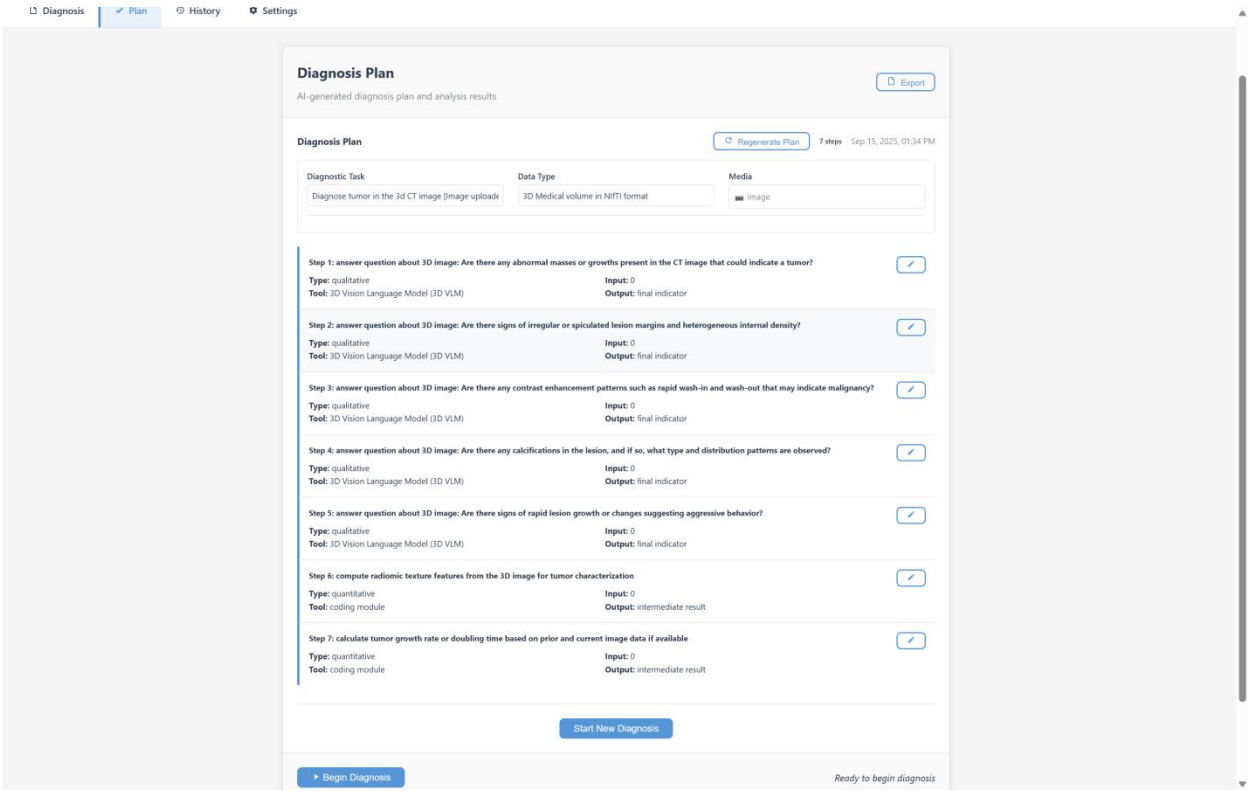

Figure 3. Demo of Casidence webpage UI. Generated plan from user entered task, including every tool used for each step.

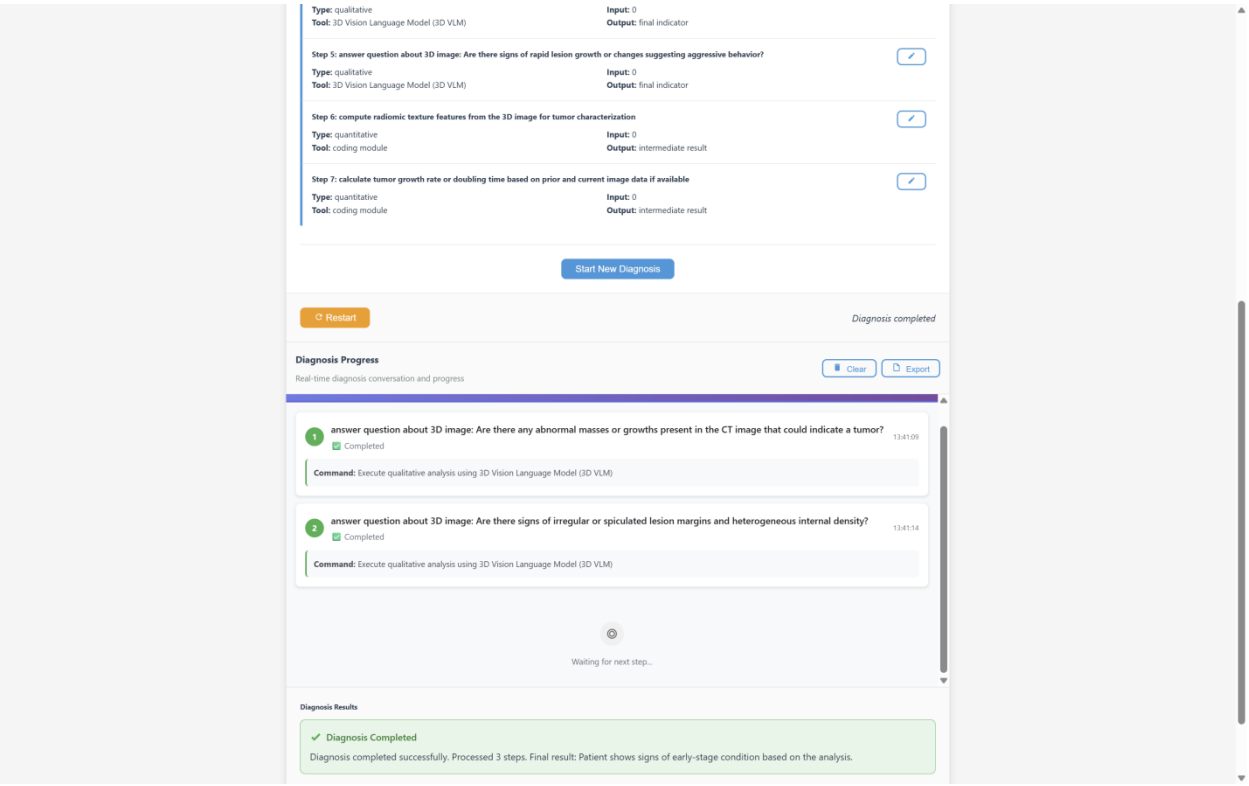

Figure 4. Demo of Casidence webpage UI. Execution steps and diagnosis results of executing plan.

3.4. Built-in Models and Tools

3D Medical VLM. Answers targeted questions on volumetric data (e.g., NIfTI) and supports automated question generation from step descriptions.

2D VLMs. A general 2D VLM and a medical 2D VLM (e.g., LLaVA-Med) for qualitative evaluations on radiographs, fundus photos and pathology slides.

Coding Agent. Generates and runs analysis code to compute measurements and statistics; used for quantitative evidence.

Similar Case Retrieval (SCR). Searches for similar cases in the database and literature through user prompts and interactions.

Segmentation (MedSAM). Produces ROIs and derived metrics that feed quantitative evidence.

3.5. Implementation Notes

All roles communicate through JSON-serializable schemas with strict parsing and fallback strategies to handle partially valid model outputs. Plans support generation only and execution-only modes to enable human review and iterative refinement. Evidence and context stores are append-only with versioned records, enabling full provenance from final conclusions back to raw tool outputs.

3.6. Researcher Architecture

Researcher The *Researcher* serves as the reasoning core of Casidence (Figure 5), following a plan–refine–orchestrate–search–report pipeline. A planner decomposes the input question into targeted steps, which are refined as new evidence is gathered. An orchestrator assigns tasks to specialized executors, while a ReAct+RetrievalAugmented Generation (RAG) module conducts searches across authoritative sources (e.g., PubMed, MedRxiv, web crawlers). Finally, a reporter synthesizes the results into a concise, citation-ready answer. All findings are stored as structured observations in Casidence's context store, forming a reliable basis for the Diagnostician's decisions.

The *Researcher* is designed to transform open-ended clinical questions into trustworthy outputs by combining deliberate planning, explicit orchestration, retrieval augmented search, and focused reporting. Unlike ad-hoc querying, it systematically expands questions into targeted sub-tasks,

searches authoritative sources, consolidates results, and returns both a clinically usable answer and a structured cache of supporting material. This process ensures that downstream agents have access to detailed, verifiable evidence for robust diagnostic reasoning.

Pipeline As illustrated in the Researcher figure, the dataflow is: *Research Question → Planner → Plan Modifier → Orchestrator → ReAct+RAG → Reporter → Answer*.

State is persisted across the graph (dialog messages, current plan, iteration count, observations, and final report), enabling recovery, reproducibility, and iterative refinement when more evidence is required.

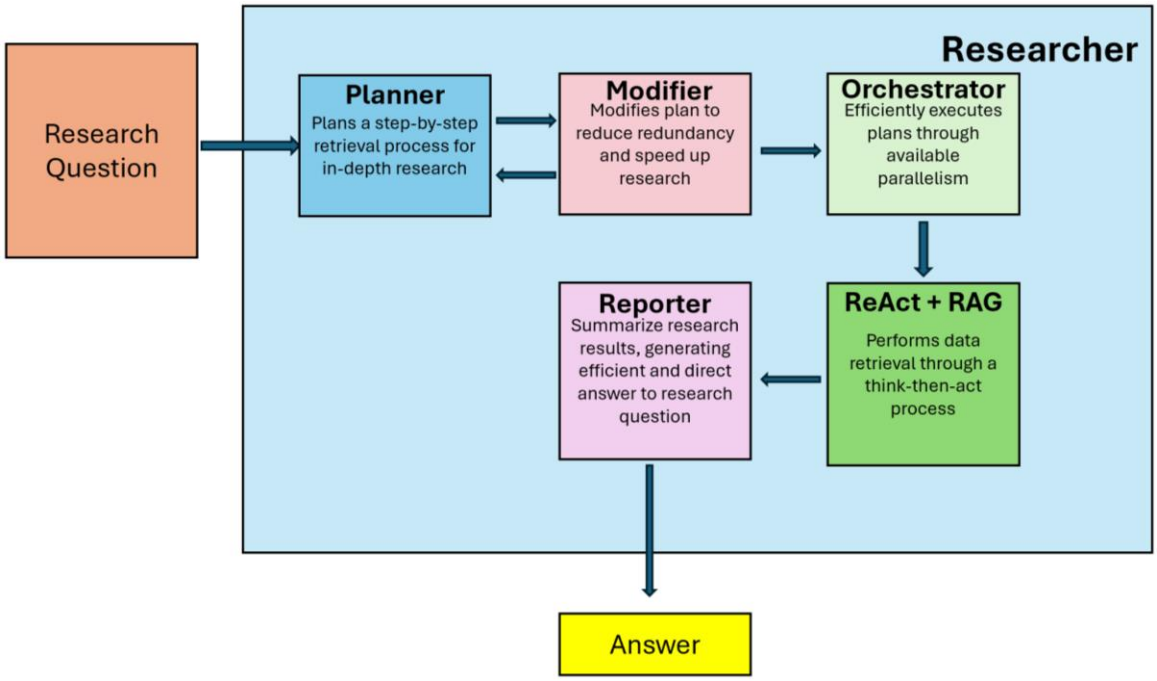

Figure 5. Researcher architecture. A clinical research question is decomposed by the *Planner*, refined by the *Plan Modifier*, orchestrated to execution by the *Orchestrator*, answered via a *ReAct+RAG* search module, and synthesized by the *Reporter* into a concise, sourced answer. Arrows denote control/data flow and support iterative refinement.

Planning The *Planner* inspects the question and any available context and emits a compact plan of at most a few steps (typically ≤ 3) that can be executed independently or in parallel. Each step is typed as either research (external information gathering) or processing (local synthesis, filtering, or light analysis). Steps declare a short title, intent, expected inputs/outputs, and a need_web_search flag to control use of external search. The *Plan Modifier* reviews the initial plan for coverage and redundancy, merges or splits steps when appropriate, and—if execution reveals gaps or contradictions—requests a minimal revision rather than expanding unboundedly. This keeps compute targeted while preserving completeness.

Orchestration and execution The *Orchestrator* is a controller that routes unexecuted steps to the appropriate executor, tracks dependencies, and exploits parallelism when steps are independent. It also enforces budgets (maximum plan iterations, step count, and search results per step) and standardizes error handling and retries, ensuring that partial progress is captured as observations even when a specific tool fails.

ReAct + RAG For steps requiring external evidence, the *ReAct+RAG* executor interleaves reasoning with actions: it generates hypotheses and sub-queries ("*think*"), performs searches and retrieval ("*act*"), inspects results, and decides whether to continue digging or to consolidate. Retrieval spans biomedical databases (e.g., peer-reviewed literature and preprint servers) and high-quality web sources; when domain-specific tools are available, they can be dynamically called through a tool interface without architectural changes. Results are normalized into a common schema capturing key claims, quoted snippets when helpful, basic metadata (source, date, venue), and a stable reference list.

Deduplication and contradiction checks are applied across steps to avoid double counting and to surface disagreements explicitly.

Reporting The *Reporter* converts all observations into a direct, clinically useful answer with compact context. Outputs include: (i) a short answer to the original question; (ii) a bullet-style synthesis summarizing consensus and edge cases; (iii) a list of key supporting findings with provenance; and (iv) a final reference block. The formatting is adjustable (bulleted, numbered, or paragraph) to serve downstream prompts. The same bundle is stored in the system's context store so that subsequent planning and diagnosis are conditioned on the exact evidence the Researcher assembled.

Design principles and guarantees (1) *Completeness with discipline*: plans are intentionally small and auditable, but the system will iterate when confidence is low or contradictions remain. (2) *Provenance*: every claim is traceable to a source and step, enabling the Diagnostician to weigh evidence. (3) *Efficiency*: bounded step counts, result caps, and early-exit conditions prevent unnecessary exploration. (4) *Extensibility*: new search tools or processing modules can register their capabilities and be composed by the Planner without modifying the orchestration logic. (5) *Clinical utility*: the final product is not just a summary—it is a structured, high-signal research artifact that provides the Diagnostician with a defensible foundation for case assessment.

4. Medical Visual-Language Model

Goal In *Casidence*, a 3D multimodal visual-language model is a necessary component in order to incorporate medical imaging data as user input, as GPT-5 does not natively process 3D medical images. To bridge this gap, we fine-tune a 3D Medical Vision–Language Model (VLM) capable of handling volumetric imaging. Specifically, the model is extensively finetuned on the CT-RATE dataset [18], achieving state-of-the-art performance on its evaluation split. Our contribution lies in designing a task-specific training recipe and a tailored data curation pipeline built upon an existing VLM architecture, without introducing modifications to the underlying network components.

4.1. Base Architectures

LLaVA family Our model follows the LLaVA design in which a frozen (or partially trainable) vision encoder is connected to an LLM through a lightweight projection module. Images are converted to visual tokens by the encoder; the projector aligns these tokens to the LLM's embedding space; instruction-tuning teaches the LLM to attend to visual context and produce grounded text outputs.

Med3DVLM, overview We adopt Med3DVLM [20] as the 3Dbackbone. Med3DVLM extends the LLaVA recipe from 2D images to volumetric studies by tokenizing 3D patches/slices and injecting volumetric positional priors, enabling the LLM to reason over anatomy across depth. The design follows the *MetaFormer* [43] idea: the block structure separates a generic token-mixer (e.g., attention) from per-token feedforward updates, making it straightforward to process 3D tokens while preserving the language interface. In our setup we keep the Med3DVLM encoders, projector, and LLM unchanged and only fine-tune parameters during supervised instruction training.

4.2. Training Dataset

CT-RATE CT-RATE [18] is a CT radiology dataset consisting 50,188 unique radiology reports and CT volumes with radiology-style prompts and rating/assessment targets. We use the CT-Rate's split of dataset to create training set for model tuning and the held-out evaluation set for reporting. To fit the VLM interface, each sample is converted into instruction–response pairs covering abnormality presence, localization, and severity ratings. Volumes are resampled to an isotropic spacing, clipped to a fixed Hounsfield range, and normalized. For memory-efficient batching, we sample either (i) fixed-depth sub volumes or (ii) salient slices determined by simple heuristics.

4.3. Training Details

Objective and schedules We perform supervised instruction tuning with a standard next-token cross-entropy objective on concatenated visual tokens and text prompts. Mixed-precision training is used throughout. Unless otherwise noted, optimization uses AdamW [44] with cosine decay and linear warmup. We apply gradient clipping and weight decay to stabilize long-context updates.

Freezing strategy to balance stability and adaptation, we adopt a staged scheme: (1) warm-up with the vision encoder frozen while updating the projector and LLM; (2) optionally unfreeze the upper encoder blocks for modest visual adaptation; (3) conduct a brief final pass with a reduced learning rate to harmonize the projector and language layers. We found this schedule sufficient to specialize to CT-Rate without overfitting.

Batching and context Each batch contains studies padded to a fixed token budget. Prompts follow an instruction style that asks for ratings/justifications; targets are short, normalized strings (e.g., discrete grades or bounded numerics). We interleave questions per study (presence, location, severity) to encourage multifacet grounding.

Regularization and augmentation Unlike volumetric augmentation preprocessing of 3D images in Med3DVLM where volumetric information is corrupted, we unify the voxel spatial scale over the entire dataset to preserve volume information.

Implementation notes. We initialize from publicly available Med3DVLM weights, maintain the architecture intact and train with a global batch size of 64 for 1 epoch at a maximum learning rate of 0.00005. Training runs on 4 A100 GPUs with maximum sequence length 768 tokens.

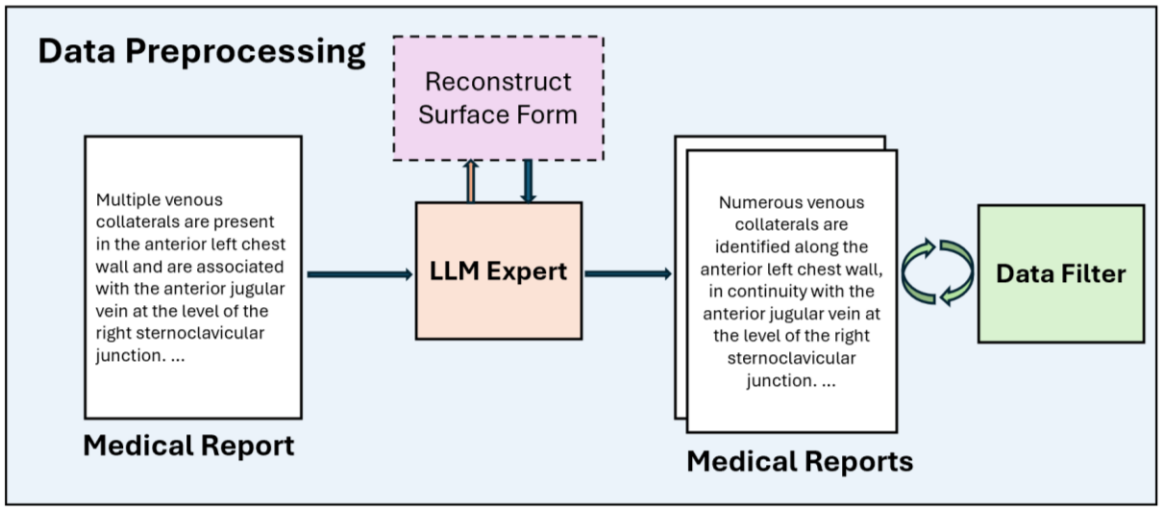

Figure 6. Overview of the Pseudo Labeling process. Each medical report is given to a Large Language Model with engineered prompt to construct two medical reports of the same medical information but different surface forms. We then filter failed cases after generation to ensure quality.

5. Query Auto Encoder: Similar Case Retrieval

For similar case retrieval, we developed a bi-encoder architecture that indexes every case separately into vectors in the same embedding space, and performs Maximum Inner Product Search (MIPS) on the indexed database with the encoded query vector. To train our vector encoder, we propose Query Auto Encoder (QAE): a self-supervised masked denoising auto-encoder structure that effectively leverages the powerful understanding of medical information of medical language models, training a Query Encoder that is able to encode aligned medical semantic vector representations of the input sequence. For better user interactions and compact, enhanced medical semantic data, we encode only the compact sequence data and not redundant visual information that could be compactly represented by more sequence embeddings with more abstract medical semantic information.

5.1. Dataset & Pseudo Labels

For QAE, we again use the publicly available CT-RATE [18] dataset. Our pseudolabeling process is represented in Figure 6. For each radiology report, we use Large Language Model to generate similar reports with the same information but different forms, such as grammar, sentence order, wording, etc. These pseudo-labeled reports allow our model to develop a robust encoder that focuses on medical semantic information only. We then filter the generated reports, removing low-quality data with overtly different information. We keep generated data with small tweaks in information to allow greater robustest of medical semantic space embedding during training, better encoding sequences of similar medical information to similar vectors. We finally create pairs of all similar labels, each pair being a training case, resulting in 86k pairs. We took 80% data as training data, and 20% data as the evaluation dataset to evaluate the robustness of medical semantic encoding of our query encoder. There are no overlaps in CT volumes between the training dataset and evaluation dataset.

5.2. Modeling

Our QAE architecture consists of three main components shown in Figure 7: Sequence Encoder/Language Model (LM), Query Encoder and Query Decoder. We use a Clinical-BERT [45] as our sequence encoder to effectively convert medical information into embeddings consisting of medical information. Our Query Encoder then encodes the medical information consisting of sequence into a fixed-length positional-independent embedding, representing medical information within the sequence, also referred to as the query embedding. Before inputting into the Query Decoder, we will first randomly mask different tokens of LM output sequence embeddings. The masked sequence embeddings and the query embedding will then be passed into the Query-Decoder to recover the masked embedding. During training, we separately encode the pair sequences, and swap query embeddings for the purpose of excluding surface form information in the query embedding, keeping only positional-independent medical information. The Query Decoder is used only during training and will be removed in inference time.

5.3. Notations

During training, a pair of sequences will be entered as a case. We denote our input sequences as Seq_1 and Seq2. We denote sequence embeddings of Seq_1 encoded by sequence encoder as X_1, the query embedding encoded by the query encoder as Q_1, the randomly masked sequence embeddings as X_1^{mask}, its corresponding mask as M_1 and the recovered sequence embedding by the Query Decoder as $\hat{X}_1$. The same applies to Seq_2 , with X_2, Q_2, X_2^{mask}, M_2 and $\hat{X}_2$.

5.4. Contrastive learning & Auto-Encoder

To effectively ensure the encoded query embeddings contain medical information only, our Query Auto Encoder takes in a pair of sequences of same medical information and different surface form as a training case, and use the other sequence's query embedding to recover the current sequence.

We employ both contrastive learning and masked embedding reconstruction to effectively train a medical information only embedding space for our Query Encoder. Contrastive learning is performed between two query embeddings to ensure the alignment in embedding space of the same medical information, and create difference with query embeddings of different medical information within other case pairs. Masked embedding reconstruction ensures that the encoded medical information within query embeddings is correct and based off of the sequence such that they can be used to recover masked medical information within the sequence. To ensure effective contrastive learning, we require a minimum batch size of 16.

During training, we first encode our sequences with the sequence encoder, obtaining X_1 and X_2, and obtain X_1^{mask} and X_2^{mask} by randomly masking 1/3 of the embed-dings. We then obtain Q_1 with X_1 and Q_2 with X_2 using the query encoder. We then use Q_1 and X_2^{mask} to generate $\hat{X}_2$

and calculate a Mean Squared Loss (MSE) between X_2 and $\hat{X}_2$ on the recovered masked embeddings as the following equation, denoted as L^2_{MSE}.

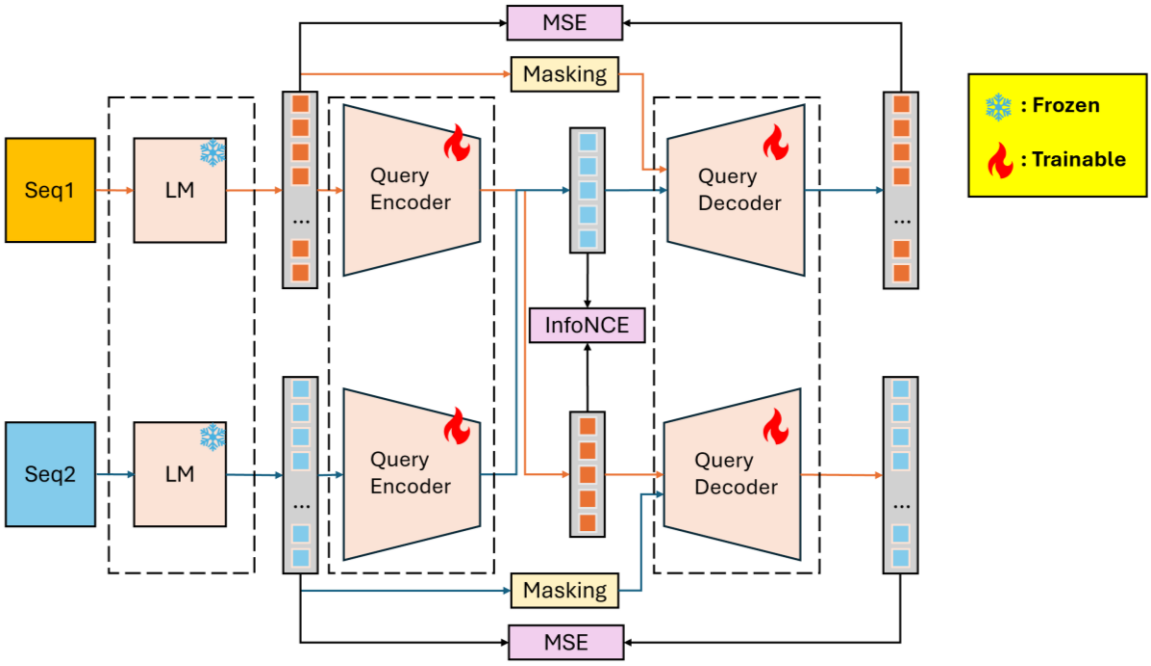

Figure 7. Query Auto Encoder (QAE) Overview. Given a pair of sequences representing the same medical information, frozen language model (LM) encoders produce initial embeddings that are further processed by trainable query encoders. The resulting query embeddings are masked and then reconstructed by query decoders, with objectives including mean squared error (MSE) reconstruction loss and InfoNCE [46] loss for contrastive alignment. Each sequence is used to recover the masked representation of its paired sequence, encouraging cross-sequence consistency. This design ensures that embeddings capture semantically aligned medical content across variations in text, improving robustness to paraphrases, abbreviations, and redundant phrasing often present in clinical narratives.

$$\mathcal{L}^2_{\text{MSE}} = \frac{1}{\sum_{i=1}^{L} M_2(i)} \sum_{i=1}^{L} M_2(i) \left\| X_2[i] - \hat{X}_2[i] \right\|_2^2 \tag{1}$$

We then repeat the process with Q2 and X_1^{mask}, and obtain L^1_{MSE}.

For contrastive learning, we employ the Information Noise-Contrastive Estimation (InfoNCE) loss between Q_1 and Q_2, where the diagonal is positive case, and all others are negative cases. The equation of InfoNCE loss is demonstrated in (2)

$$\begin{aligned} \ell^{-2} &= -\log \frac{\exp(\text{mim}(q_{1,i}, q_{2,i})/\tau)}{\sum_{d=1}^{L} \exp(\text{sim}(q_{1,j}, q_{2,j})/\tau)}, \\ \mathcal{L}_{\text{InfoNCE}} &= \frac{1}{2L} \sum_{i=1}^{L} (\ell_i^{1\to 2} + \ell_i^{2\to 1}). \end{aligned} \tag{2}$$

This ensures alignment of query embedding over sequences of the same information, while pushing it away from other cases in the batch, producing compact, medically aligned representations. We finally average $\mathcal{L}_{\text{InfoNCE}}$, L^1_{MSE}, and L^2_{MSE} for final loss.

$$\mathcal{L}_{\text{final}} = \frac{1}{3}(\mathcal{L}_{\text{InfoNCE}} + \mathcal{L}^1_{MSE} + \mathcal{L}^2_{MSE}) \tag{3}$$

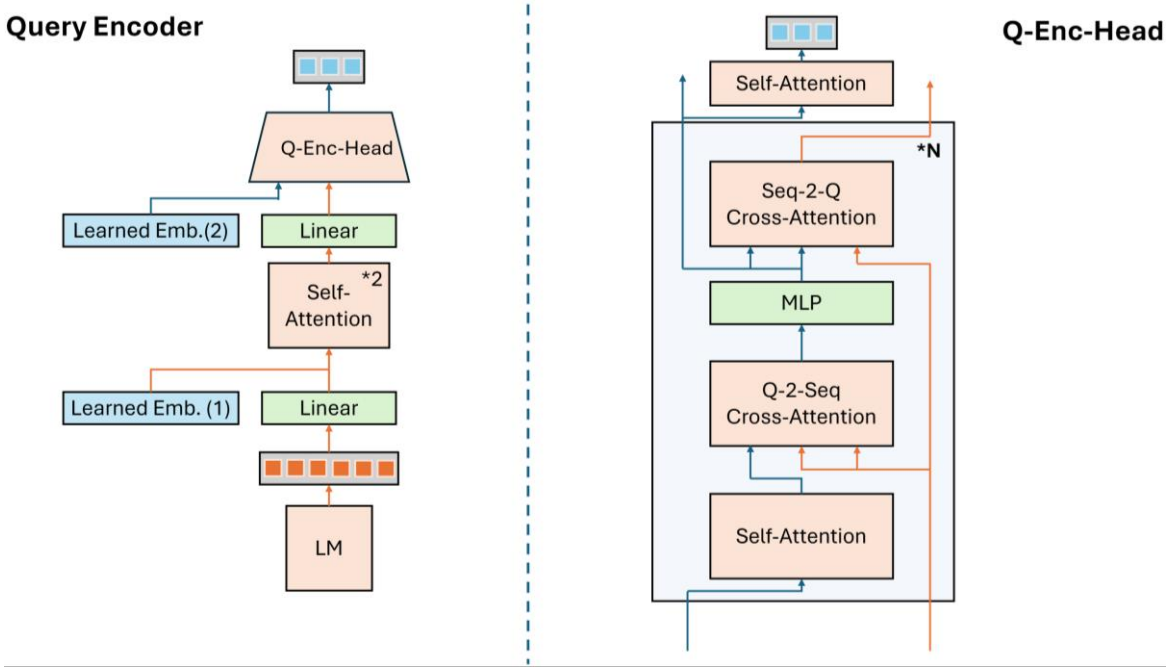

Figure 8. Query Encoder and Q-Enc-Head architectures. Left: The Query Encoder takes frozen LM embeddings and enriches them with two layers of self-attention conditioned on learned embeddings, followed by a linear projection. The resulting representations are passed into the Q-Enc-Head to form query embeddings. Right: The Q-Enc-Head refines these embeddings through stacked layers (two in this work), each consisting of self-attention, query-to-sequence (Q2Seq) cross-attention, sequence-to-query (Seq2Q) cross-attention, and an MLP block. This design enables mutual alignment between sequence embeddings and query embeddings while maintaining positional information, producing compact query vectors that capture semantic content independent of surface-form variation.

5.5. Query Encoder

Our query encoder architecture is represented in Figure 8. Before generating the query embedding, we process the sequence embeddings with two self-attention layers, allowing important information to be highlighted by the nature of the self-attention, strengthening the medical information stored in the sequence embedding. Inspired by Segment Anything Model (SAM) [47] Mask Decoder, we develop QEnc-Head to generate query embeddings of dense information through the extraction of information from the sequence embeddings with Cross-Attention. Additional to the SAM Mask Decoder, we remove positional-encoding in Cross-Attention to ensure no surface form information, such as sentence order, are stored in the query embeddings.

5.6. Query Decoder

Our query decoder architecture is represented by Figure 9. Our query decoder is adapted based on the decoder structure of classic transformer [48] and the masked token prediction pretraining of [49]. Additional to the transformer Decoder, we remove the positional encoding in cross-attention due to the irrelevance of positional information in the query embedding, and use full self-attention instead of masked attention, allowing equal training on information distributed across the entire sequence, instead of unequal information in different positions from the masked-self-attention.

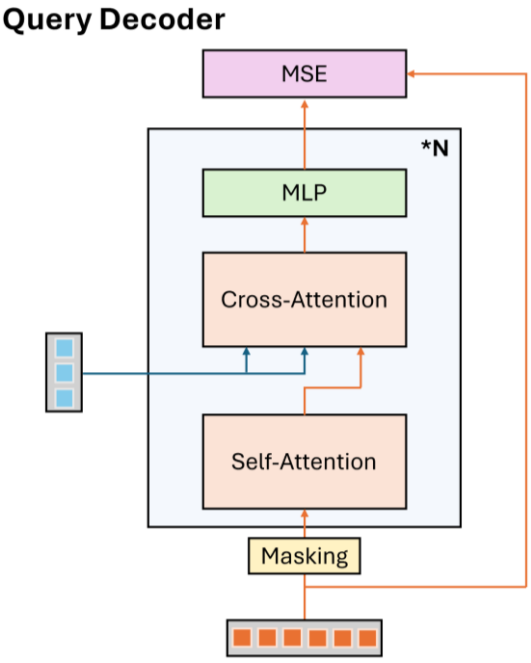

Figure 9. Overview of the Query Decoder. The masked sequence embedding is first run through a self-attention layer, then through a cross-attention without positional encoding.

We limit the scale of the Query Decoder for lower information processing ability of the Decoder to ensure the training of the Query Encoder for higher requirements in a accurate and information-rich query embedding.

5.7. Retrieval & Dataset Indexing

To perform Similar Case Retrieval (SCR) with our bi-encoder architecture, we first index our dataset and then use MIPS with the query embedding of the search sequence during inference. For dataset indexing, we first input the description of each medical case within the database into the sequence encoder and Query Encoder, then stack the generated query embeddings, obtaining a vector representation of the medical description and store each case's vector with FLAT index using FAISS [26]. During search, the search sequence will be encoded into a query embedding by the Query Encoder, and MIPS will be performed within the indexed dataset to find vector representations of the largest cosine similarity in the embedding space.

5.8. Training

During training of the QAE, we used 3 A100 GPUs and a batch size of 25 per GPU, training 36 GPU hours with a maximum learning rate of 0.00002.

6. Casidence Results

6.1. Quantitative Evaluation

We evaluate the following components of Casidence: Plan, Execution, Report. Figure 11 demonstrates quantitative evaluation metrics of Casidence.

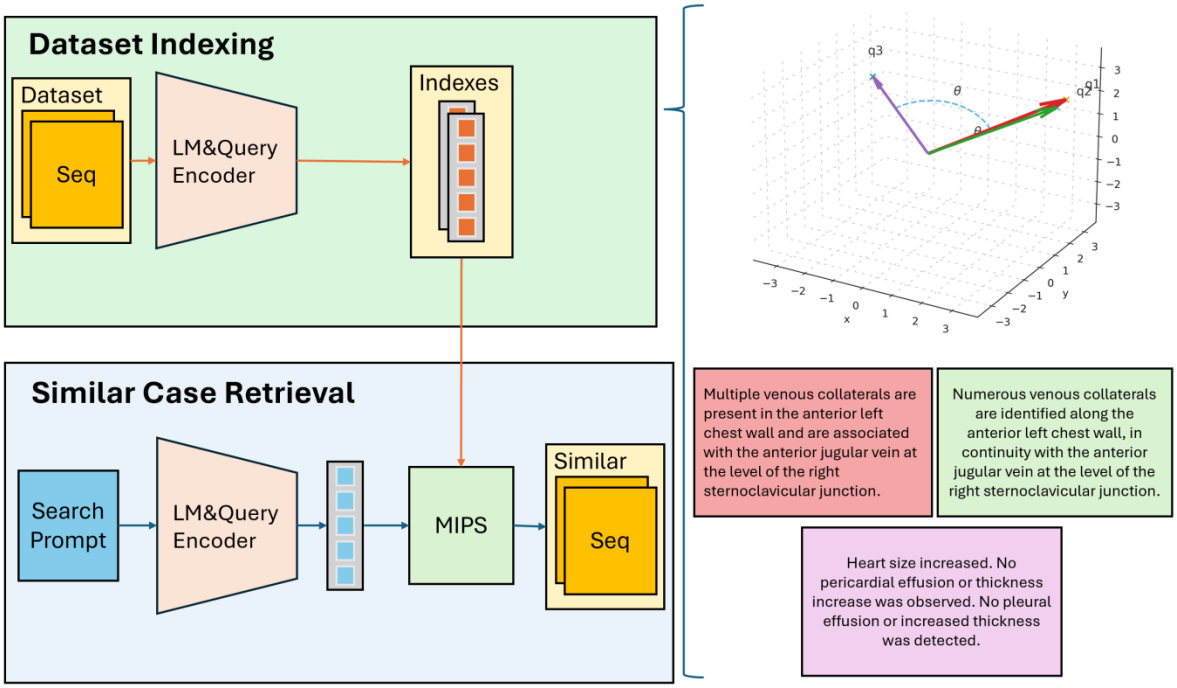

Figure 10. Top Left: Overview of Dataset Indexing. Bottom Left: Overview of Similar Case Retrieval. Right: 3D representation of indexed embedding space of 3 medical descriptions. q1 and q2 are similar and thus have high cosine similarity in the embedding space, q3 is different from the two and thus low cosine similarity in the embedding space.

Experimental Setup We evaluate Casidence on 34 cases selected at random from the CT-RATE [18] validation VQA dataset, enclosing different symptoms and both positive and negative cases. During each run, we record the generated plan, the execution process, and the final report. We then use a rubric shown in Figure 11 to score the quality of the plan, execution and report in different dimensions. For GPT review, we carefully engineered a evaluation prompt template, including the rubric, the plan, the execution steps and the final report, making GPT score each case according to the rubric. For Clinician Review, we approach the Cancer Hospital, Chinese Academy of Medical Sciences where clinicians scored Casidence in mix of real time runs and previously recorded run cases.

Analysis Casidence's research-first, plan–execute–verify architecture yields strong upstream rubric scores: GPT evaluators average ≈ 4.1 across planning (Problem Framing 4.21, Guideline Alignment 4.32, Resource Stewardship 4.24, Transparency 4.15), closely matched by clinicians at ≈ 4.0. Report-level metrics aligned with these architectural choices—Guideline Concordance (4.30) and Communication Quality (4.21)—are likewise high, indicating that our deliberate planning and rationale capture are consistently clear, guideline-grounded, and clinician-friendly.

In contrast, execution lags with a mean of 3.80 (Relevance 4.00, Correctness 3.59), trailing clinician execution performance (Relevance 5.00, Correctness 4.00) and correlating with a lower Diagnostic Accuracy of 3.35. This pattern suggests that while our research and reasoning pipeline is robust, measurement and evidence-selection errors during tool runs propagate to final judgments. Because execution depends on a state-of-the-art 3D medical VLM, we attribute the shortfall primarily to that model's output quality. Closing the ~ 0.3–0.9 execution gap—via a stronger VLM, calibration/ensemble checks, and reliability gates before evidence extraction—should lift end-to-end diagnostic accuracy while preserving Casidence's architectural strengths in research accuracy, reasoning, and usability.

Plan (score [1-5], 1=unsafe/low quality, 5=high quality)		
Item	**GPT**	**Clinician Review**
Problem framing — Clear understanding of the diagnostic question and goals.	4.21	4.00
Coverage & sufficiency — Steps likely to gather all necessary evidence; key positives/negatives considered.	3.61	4.00
Tool selection & sequencing — Right tools in an efficient order; sensible fallback if a tool fails.	4.05	4.00
Guideline alignment — Steps reflect current clinical guidance where relevant.	4.32	4.00
Resource stewardship — Avoids unnecessary steps, radiation, cost, or delay.	4.24	4.00
Transparency — Each step has a clear rationale linked to the diagnostic objective.	4.15	4.00

Execution (score [1-5], 1=unsafe/low quality, 5=high quality)		
Item	**GPT**	**Clinician Review**
Relevance — Collected evidence directly addresses the diagnostic question.	4.00	5.00
Correctness — Measurements/findings are accurate and reproducible.	3.59	4.00

Report (score [1-5], 1=unsafe/low quality, 5=high quality)		
Item	**GPT**	**Clinician Review**
Diagnostic accuracy / justified differential — Correct primary dx OR well-justified differential with probabilities.	3.35	N/A
Guideline concordance — Recommendations reflect standards; deviations justified.	4.30	N/A
Communication quality — Clear, concise, professional clinician-facing tone.	4.21	N/A

Figure 11. Qualitative Evaluation Rubric on Casidence Diagnosis runs, separated into 3 main parts: Plan, Execution, Report. Scores scale from 1.00-5.00, where 1.00 indicates unsafe and low quality, and 5.00 indicates high quality.

6.2. Qualitative Evaluation

We present a qualitative analysis on Casidence with a run case from the CT-RATE validation set shown in Figure 12. We examine how the plan–execute–verify pipeline behaves in routine CT interpretation. Our aim is to characterize the consistency of the research backbone—task decomposition, targeted tool use, and explicit evidence capture—and to identify where evidence extraction errors propagate into narrative reports. The lens is representative case workups that share the same error profile as the example below.

TASK	DESCRIBE THE FINDINGS REGARDING THE PLEURAL EFFUSION AND ATELECTASIS IN THIS CHEST CT IMAGE.
GROUND TRUTH	There is a moderate amount of pleural effusion, more prominent on the right bilaterally, and atelectasis and volume losses in the lower lobes of both lungs, more prominent on the right.
DIAGNOSIS RESULT	The patient has both pleural effusion and atelectasis. Pleural effusion is present in both hemithoraxes as evidenced by low-attenuation areas in the pleural space with Hounsfield unit values in the range of 0–20. Atelectasis is indicated by lung volume loss, mediastinal shift, and increased parenchymal density in the right lung upper lobe anterior segment and lower lobe superior segment.
COLLECTED EVIDENCE	• 3D VLM Analysis 'answer question about 3D image: Are there signs of pleural effusion, such as low-attenuation areas in the pleural space with Hounsfield unit values in the range of 0–20?': Yes, there are pleural effusion areas in both hemithorax. • 3D VLM Analysis 'answer question about 3D image: Is there evidence of atelectasis, such as lung volume loss, mediastinal shift, or increased parenchymal density?': Yes, lung volume is decreased.; Yes, a small amount of fluid is observed in the mediastinum.; Yes, an increase in parenchymal density in the right lung upper lobe anterior segment and lower lobe superior segment adjacent to the effusion may be compatible with atelectasis."
PLAN	Step 1 Tool: Med3DVLM answer question about 3D image: Are there signs of pleural effusion, such as low-attenuation areas in the pleural space with Hounsfield unit values in the range of 0–20?" Step 2 Tool: Med3DVLM answer question about 3D image: Is there evidence of atelectasis, such as lung volume loss, mediastinal shift, or increased parenchymal density?",

Figure 12. Single Casidence Run Case including Task, ground truth, diagnosis result, collected evidence, and plan. Research result is not displayed due to its overtly long length.

In this run, Casidence's research-first, plan–execute–verify architecture functioned as designed: it decomposed the task into high-value questions (presence of pleural effusion and atelectasis), invoked appropriate tools, and recorded explicit evidence prior to drafting the report. This yielded transparent, guideline-concordant reasoning and clinically sensible problem framing, mirroring our quantitative findings that planning and reporting earn high rubric scores (4.1 for planning; strong guideline alignment and communication). The safety profile was acceptable—Minor potential for harm—because the system correctly detected both effusion and atelectasis, supporting an overall clinical utility of 4 for this case.

However, the execution layer—driven by a 3D medical VQA model—proved to be the limiting factor. Feature extraction introduced unsupported specifics (e.g., Hounsfield units 0–20; mediastinal shift) and mislocalization (atelectasis placed in the right upper-lobe anterior segment) while omitting key qualifiers present in the ground truth (moderate volume; right-greater-than-left asymmetry; bilateral lower lobe predominance). These errors are representative of the quantitative gap observed in execution (mean 3.8, with lower correctness) and explain the drift between sound research planning and the less faithful final narrative. Strengthening the vision backbone should reduce propagation of extraction errors, thereby improving diagnostic accuracy without sacrificing Casidence's core strengths in research rigor and evidence-centric workflow.

7. Similar Case Retrieval Results

Our method significantly out performs existing methods in sequence-to-sequence retrieval as shown in Table 1, achieving a 100% retrieval success rate on the pseudo labeled evaluation dataset explained in Section 5.1, largely surpassing current SOTA models. This near-perfect performance reflects the strong alignment between QAE embeddings and the information structure of the dataset, as the QAE is specifically trained to disentangle semantic content from surface-form variation in this domain.

7.1. Quantitative Evaluation

Experimental Setup We evaluate two models on our pseudo-labeled CT-RATE SCR dataset of 17.1k cases: ColBERT [30] (based on ClinicalBERT) and our model. For the evaluation of each model, we independently index the dataset with the corresponding indexing method of the model. We then perform top K similar case retrieval in each case over with the model over the dataset and calculate the results. In experiment, we have three separate versions of medical reports of the same volume, with the same medical information and different surface form. We separate them into three pools. For retrieval, we perform search of vector representation of medical report of each case in the first pool on each of the two other pools, and vice versa for the two other pools, and finally average the results. Due to the different surface forms of different versions of medical report and same medical information, retrieval systems based solely off of medical information while excluding similarity in surface form will result in higher accuracy. The evaluation set and training set are separate, and our model has not seen the reports of any volume used in the evaluation set.

We calculate the following metrics in experiment of each model to evaluate its ability on similar case retrieval: R@1, R@3, R@5, R@10, Mean Reciprocal Rank (MRR). R@K refers to Recall@K: the rate at which the ground truth appears in the top K similar cases identified by the model. MRR refers to the reciprocal of the rank of the ground truth in the retrieved similar cases, where, for a case, if correct case is ranked the 3rd most similar, the MRR for this case is 1/3, and 1/5 if ranked as the 5th most similar, etc.

Indexing Due to the positional-independent and query embedding of our method, we simply index our dataset by encoding every case into query embeddings $Q \in RC\times D$, then stacking the embeddings into one vector representation $V \in RCD$, where C refers to the constant length of our fixed-length query embeddings. We further index each case with FAISS [26] using the default and accurate FLAT indexing.

Table 1. Average retrieval accuracy on the evaluation set (17.1k cases pool). R@K: rate of ground truth retrieved in top K similarity cases from retrieval. MRR: reciprocal of correct case rank in top similarity cases obtained by model.

	R@1	R@3	R@5	R@10	MRR
Ours	1.00000	1.00000	1.00000	1.00000	1.00000
ColBERT (ClinicalBERT)	0.51235	0.76064	0.79585	0.83651	0.64438

For ColBERT based on Clinical BERT, we first encode each case into sequence embeddings. Then, we index each case's unfixed length multi-vector representation using FAISS IVF indexing following the ColBERT method.

We separately encode the three pools, obtaining three datasets.

Top K Retrieval For the retrieval of each case of our method, we encode each medical report into a vector representation and performs MIPS search in the two other datasets retrieving the top 10 most similar cases, then calculating the retrieval metrics. For MRR, we take 0 if the correct case was not retrieved in the top 10 most similar cases. For the retrieval of ColBERT, we encode each medical report into a multi-vector representation, and perform maximum similarity search in the previously indexed dataset. Specifically, for the similarity calculation between two multi-vector representation, ColBERT takes the maximum similarity of each vector to the other vectors in the other multi-vector representation, and sum the max-similarities to obtain a final similarity. We also retrieve the top 10 most similar cases for ColBERT, and then calculate evaluation metrics.

Analysis Table 1 demonstrates the evaluation results of Our method and ColBERT method with ClinicalBERT base model. We proudly discover that our method achieves 100% accuracy over all evaluation metrics, largely surpassing the SOTA ColBERT method. We believe this is due to the enhanced medical information alignment and surface form information removal of our method, which is directly tied to a high cosine similarity in embedding space of similar medical information. On the other hand, the lower accuracy demonstrated by ColBERT's maximum-similarity search that weakly filter off information order further proves the robustness and ability to extract and store only the

medical information in an aligned way, whereas misaligned storage of the same information within the query embedding will lead to low similarity vectors as they will then not be compared with each other in cosine similarity of the vector representations.

7.2. Qualitative Evaluation

Figure 13presents a qualitative comparison between the correct case, most relevant retrieval result of ColBERT based on ClinicalBERT and our method's second relevant retrieval result. We avoid using the most relevant retrieval result of our method due to the 100% retrieval accuracy and to prove the robustness of our retrieval system, demonstrating its ability to retrieve cases of highly similar information, further proving its ability to remove surface forms and encode aligned medical information only.

Similar: all three | reports 1 & 2 | reports 1 & 3

Difference: both 1&2 and 1&3 | 1 & 2 | 1 & 3

Correct Case	Trachea, both main bronchi are open. Mediastinal main vascular structures, heart contour, size are normal. Thoracic aorta diameter is normal. Pericardial effusion-thickening was not observed. Thoracic esophagus calibration was normal and no significant tumoral wall thickening was detected. No enlarged lymph nodes in prevascular, pre-paratracheal, subcarinal or bilateral hilar-axillary pathological dimensions were detected. When examined in the lung parenchyma window; Peripherally located subpleural nodular **ground glass densities are observed in both lungs. The findings were evaluated in terms of early viral pneumonia and Covid-19 pneumonia.** Clinical and laboratory correlation and close follow-up are recommended. Upper abdominal organs included in the sections are normal. No space-occupying lesion was detected in the liver that entered the cross-sectional area. Bilateral adrenal glands were normal and no space-occupying lesion was detected. Bone structures in the study area are natural. Vertebral corpus heights are preserved.
ColBERT (ClinicalBERT)	The trachea and both main bronchi are patent. The mediastinal main vascular structures, heart contour, and size are normal. The thoracic aorta diameter is normal. No pericardial effusion-thickening was observed. The thoracic esophagus is of normal caliber, and no significant tumoral wall thickening was detected. There were no enlarged lymph nodes in the prevascular, pre-paratracheal, subcarinal, or bilateral hilar-axillary regions. Upon examination of the lung parenchyma in the lung parenchyma window, **the aeration of both lung parenchyma is normal, and no nodular or infiltrative lesions were detected in the lung parenchyma. No pleural effusion-thickening was observed.** The upper abdominal organs included in the sections are normal. No space-occupying lesion was detected in the liver that entered the cross-sectional area. The bilateral adrenal glands are normal, and no space-occupying lesion was detected. The bone structures in the study area appear natural, with preserved vertebral corpus heights.
Ours	The trachea and both main bronchi are patent, with normal appearances of the mediastinal main vascular structures, heart contour, and size. The thoracic aorta diameter is normal, and there is no evidence of pericardial effusion or thickening. The thoracic esophagus is of normal caliber, and no significant tumoral wall thickening is identified. No enlarged lymph nodes are observed in the prevascular, pre-paratracheal, subcarinal, or bilateral hilar-axillary regions. In the lung parenchyma window, diffuse and patchy subpleural ground-glass opacities are seen in both lungs, suggestive of a possible Covid-19 pneumonia. The upper abdominal organs are normal, and there are no space-occupying lesions in the liver that intersect the cross-sectional area. The bilateral adrenal glands are normal, and no space-occupying lesions are detected. The bone structures within the study area are unremarkable, with preserved vertebral corpus heights.

Figure 13. Example of one retrieval case. Correct Case shows the ground-truth report that should be retrieved. ColBERT [30] (ClinicalBERT) displays the top-ranked case from ColBERT, which shares surface-level similarities but misses key clinical findings (e.g., absence of ground-glass opacities). Ours shows the second-ranked case from our model, which still accurately captures the critical diagnostic details—including diffuse subpleural ground-glass opacities consistent with Covid-19 pneumonia—while preserving other clinically relevant observations.

Table 2. Evaluation results on the CT-RATE dataset. For consistency with prior work, we report only BLEU-1 and METEOR, as these are the metrics used in earlier studies. Our model achieves state-of-the-art performance on METEOR, surpassing all baselines, while obtaining mid-range results on BLEU-1. Given that BLEU-1 is a relatively coarse metric focused on surface-level n-gram overlap, stronger METEOR score more reliably reflects improved semantic alignment and clinical relevance.

	BLEU-1	METEOR
LLaVA 1.6 (Mistral 7B)	0.0542	0.1050
LLaVA 1.6 (Vicuna 13B)	0.0438	0.0960
CXR-LLaVA	0.2029	0.1396
LLaVA-Med	0.1373	0.1561
CT-CHAT (Mistral 7B)	0.4702	0.2820
CT-CHAT (Vicuna 13B)	0.4747	0.2915
CT-CHAT (Llama 3.1 8B)	0.4801	0.2936
CT-CHAT (Llama 3.1 70B)	0.4824	0.2948
Ours	0.3940	0.3713

From a clinician's perspective, the correct result and our model's second most relevant retrieval align on the central clinical signal—bilateral subpleural ground-glass change compatible with early viral/COVID pneumonia—despite differing descriptors ("peripherally located . . . nodular" vs. "diffuse and patchy"). This indicates the system is robust to surface-form variation and captures pathology-level semantics rather than boilerplate similarities. In contrast, ColBERT (ClinicalBERT)'s top result shares abundant routine negatives and normal structures with correct result, but critically contradicts the pulmonary finding by asserting normal aeration with no nodular or infiltrative lesions, i.e., it matches on generic radiology scaffolding rather than the disease-defining content. Even using our *second-best* hit, the retrieval remains clinically concordant with the target, underscoring generalization beyond lexical overlap and demonstrating that our architecture surpasses the ColBERT+ClinicalBERT baseline in prioritizing truly relevant, pathology-aligned evidence.

8. 3D Medical Vision-Language Model Results

Our finetuned model demonstrates SOTA results in the CT-RATE Visual Question Answering (VQA) long answer benchmark as demonstrated in Table 2. It out performs all baselines in the METEOR metrics, and performs in the middle tier for the BLEU-1 metrics. However, the BLEU-1 metrics evaluates only wording similarity, whereas the METEOR evaluates the similarity of the generated text to the ground truth as a whole, indicating better inference performance, as the BLEU-1 metric can be easily affected by differences in grammar. We evaluate only on BLEU1 and METEOR instead of Bert Score due to the lack of Bert Score evaluations in the baseline.

9. Discussion

9.1. Addressing Limitations of existing implementations

Prior medical agentic systems have typically taken one of two forms: (i) comprehensive research platforms designed to support clinical decision-making, or (ii) orchestrator-based frameworks that integrate a wide array of tools. However, to the best of our knowledge, no existing system has adopted a research-first and evidence gathering–centric architecture explicitly tailored for diagnostic reasoning and clinical application. For instance, MedAgentPro [40] follows research–plan–diagnose paradigm, but its implementation relies on a weak research module, employs a non-iterative process, and lacks mechanisms for effective human–AI collaboration. Similarly, current 3D Medical Vision–Language Models are often fine-tuned on isolated image–question–answer pairs, which restricts them to single-turn interactions. This setup undermines their ability to sustain multi-turn dialogue,

diminishes their language capacity, and limits their usefulness for complex diagnostic reasoning. Casidence addresses this gap by embedding 3D Medical VLMs into a broader agentic framework: the models function as feature-collection tools, queried through a sequence of simple and targeted questions to progressively build an evidence base for downstream reasoning. While this design strengthens reasoning capabilities, the diagnostic accuracy of Casidence remains closely tied to the underlying VQA performance of its foundation models, as further discussed in Section 9.2.

A similar limitation arises in the domain of similar-case retrieval. Prior approaches typically rely on multi-vector representations with maximum similarity search or on single-vector embeddings. These methods do not disentangle semantic meaning from surface-form features, leaving them vulnerable to spurious matches caused by superficial similarities such as sentence order, grammar, or phrasing, rather than underlying clinical content. To overcome this, we propose the Query Auto Encoder (QAE), which extracts semantic information from sequence embeddings and maps it into query embeddings invariant to surface-form variation. This design enables more accurate retrieval driven by true medical content rather than linguistic artifacts.

9.2. Challenges & Future Work

The accuracy of Casidence final diagnosis suffers from the lower accuracy of current 3D medical visual-language models (VLMs). Currently, we employed our finetuned Med3DVLM, a model achieving SOTA results in the field. However, execution correction in our evaluation of Med3DVLM in Figure11 still demonstrates low correctness, leading to incorrect evidence collected and low final report accuracy. With the development of future 3D medical VLMs, the accuracy and report quality of Casidence will largely increase.

In QAE, we address the challenge of surface-form bias in similar-case retrieval. A potential limitation, however, is that QAE may require domain-specific training to adapt the query embedding space for specialized searches, such as medical report retrieval. Because information is compressed into fixed-length embeddings, QAE implicitly assumes that all relevant content—whether explicitly stated or not—can be represented within that vector. This constrains the scalability of information captured. While expanding the embedding dimension could mitigate the issue, doing so significantly increases computational cost due to the quadratic complexity of transformer models. Future work should explore representations that preserve alignment while scaling to broader and more general retrieval tasks.

10. Conclusion

In this paper, we present Casidence (Case-Evidence), an evidence gathering based agentic system developed for diagnosis with comprehensive researching system, various SOTA tools for evidence gathering, and strong human-AI collaboration and iterative approach to greatly assist the process of performing diagnosis and generating medical reports; Query Auto Encoder (QAE), a novel similar case retrieval architecture able to encode sequences into a single vector representation with aligned information and without surface forms, allowing accurate similar case retrieval over only medical information; and a finetuned 3D Medical Vision-Language Model in Medical Visual Question Answering (VQA) based on the Med3DVLM architecture.

References

[1] Harsha Nori, Mayank Daswani, Christopher Kelly, Scott Lundberg, Marco Tulio Ribeiro, Marc Wilson, Xiaoxuan Liu, Viknesh Sounderajah, Jonathan Carlson, Matthew P Lungren, et al. Sequential diagnosis with language models. *arXiv preprint arXiv: 2506.22405*, 2025.

[2] Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36: 28541 – 28564, 2023.

[3] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748 – 8763. PmLR, 2021.

[4] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904 – 4916. PMLR, 2021.

[5] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019.

[6] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35: 23716 – 23736, 2022.

[7] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730 – 19742. PMLR, 2023.

[8] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv: 2410.21276*, 2024.

[9] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv: 2312.11805*, 2023.

[10] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36: 34892 – 34916, 2023.

[11] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425 – 2433, 2015.

[12] Phillip Sloan, Philip Clatworthy, Edwin Simpson, and Majid Mirmehdi. Automated radiology report generation: A review of recent advances. *IEEE Reviews in Biomedical Engineering*, 18: 368 – 387, 2024.

[13] Zihan Li, Diping Song, Zefeng Yang, Deming Wang, Fei Li, Xiulan Zhang, Paul E Kinahan, and Yu Qiao. Visionunite: A vision-language foundation model for ophthalmology enhanced with clinical knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[14] Tao Tu, Shekoofeh Azizi, Danny Driess, Mike Schaekermann, Mohamed Amin, Pi-Chuan Chang, Andrew Carroll, Charles Lau, Ryutaro Tanno, Ira Ktena, et al. Towards generalist biomedical ai. *Nejm Ai*, 1 (3): AIoa2300138, 2024.

[15] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, et al. Palm-e: An embodied multimodal language model. 2023.

[16] Haoran Lai, Zihang Jiang, Qingsong Yao, Rongsheng Wang, Zhiyang He, Xiaodong Tao, Wei Wei, Weifu Lv, and S Kevin Zhou. E3d-gpt: enhanced 3d visual foundation for medical vision-language model. *arXiv preprint arXiv: 2410.14200*, 2024.

[17] Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Hui Hui, Yanfeng Wang, and Weidi Xie. Towards generalist foundation model for radiology by leveraging web-scale 2d&3d medical data. *Nature Communications*, 16 (1): 7866, 2025.

[18] Ibrahim Ethem Hamamci, Sezgin Er, Chenyu Wang, Furkan Almas, Ayse Gulnihan Simsek, Sevval Nil Esirgun, Irem Doga, Omer Faruk Durugol, Weicheng Dai, Murong Xu, et al. Developing generalist foundation models from a multimodal dataset for 3d computed tomography. *arXiv preprint arXiv: 2403.17834*, 2024.

[19] Fan Bai, Yuxin Du, Tiejun Huang, Max Q-H Meng, and Bo Zhao. M3d: Advancing 3d medical image analysis with multi-modal large language models. *arXiv preprint arXiv: 2404.00578*, 2024.

[20] Yu Xin, Gorkem Can Ates, Kuang Gong, and Wei Shao. Med3dvlm: An efficient vision-language model for 3d medical image analysis. *IEEE Journal of Biomedical and Health Informatics*, (99): 1 – 14, 2025.

[21] Stephen Robertson, Hugo Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3 (4): 333 – 389, 2009.

[22] David A Hanauer. Emerse: the electronic medical record search engine. In *AMIA annual symposium proceedings*, volume 2006, page 1189, 2006.

[23] David A Hanauer, Qiaozhu Mei, James Law, Ritu Khanna, and Kai Zheng. Supporting information retrieval from electronic health records: a report of university of michigan's nine-year experience in developing and using the electronic medical record search engine (emerse). *Journal of biomedical informatics*, 55: 290 – 300, 2015.

[24] Wendy W Chapman, Will Bridewell, Paul Hanbury, Gregory F Cooper, and Bruce G Buchanan. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of biomedical informatics*, 34 (5): 301 – 310, 2001.

[25] Chaitanya Shivade, Preethi Raghavan, Eric Fosler-Lussier, Peter J Embi, Noemie Elhadad, Stephen B Johnson, and Albert M Lai. A review of approaches to identifying patient phenotype cohorts using electronic health records. *Journal of the American Medical Informatics Association*, 21 (2): 221 – 230, 2014.

[26] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7 (3): 535 – 547, 2019.

[27] Yu A Malkov and Dmitry A Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence*, 42 (4): 824 – 836, 2018.

[28] Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. Accelerating large-scale inference with anisotropic vector quantization. In *International Conference on Machine Learning*, pages 3887–3896. PMLR, 2020.

[29] Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. Publicly available clinical bert embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72 – 78, 2019.

[30] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39 – 48, 2020.

[31] Chengkuan Chen, Ming Y Lu, Drew FK Williamson, Tiffany Y Chen, Andrew J Schaumberg, and Faisal Mahmood. Fast and scalable search of whole-slide images via self-supervised deep learning. *Nature Biomedical Engineering*, 6 (12): 1420 – 1434, 2022.

[32] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. FILIP: Fine-grained interactive language-image pre-training. In *International Conference on Learning Representations*, 2022.

[33] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824 – 24837, 2022.

[34] Jianing Qiu, Kyle Lam, Guohao Li, Amish Acharya, Tien Yin Wong, Ara Darzi, Wu Yuan, and Eric J Topol. Llm-based agentic systems in medicine and healthcare. *Nature Machine Intelligence*, 6 (12): 1418 – 1420, 2024.

[35] Yubin Kim, Chanwoo Park, Hyewon Jeong, Yik S Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Marzyeh Ghassemi, Cynthia Breazeal, and Hae W Park. Mdagents: An adaptive collaboration of llms for medical decision-making. *Advances in Neural Information Processing Systems*, 37: 79410 – 79452, 2024.

[36] Hao Wei, Jianing Qiu, Haibao Yu, and Wu Yuan. Medco: Medical education copilots based on a multi-agent framework. In *European Conference on Computer Vision*, pages 119 – 135. Springer, 2024.

[37] Xuhai Xu, Bingsheng Yao, Ziqi Yang, Shao Zhang, Ethan Rogers, Stephen Intille, Nawar Shara, Guodong Gao, and Dakuo Wang. Talk2care: Facilitating asynchronous patient-provider communication with largelanguage-model. In *Proceedings of the AAAI Symposium Series*, volume 4, pages 146 – 151, 2024.

[38] Yixing Jiang, Kameron C Black, Gloria Geng, Danny Park, James Zou, Andrew Y Ng, and Jonathan H Chen. MadgeTech: a virtual ehr environment to benchmark medical llm agents. *NEJM AI*, page AIdbp2500144, 2025.

[39] Binxu Li, Tiankai Yan, Yuanting Pan, Jie Luo, Ruiyang Ji, Jiayuan Ding, Zhe Xu, Shilong Liu, Haoyu Dong, Zihao Lin, et al. Mmedagent: Learning to use medical tools with multi-modal agent. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8745 – 8760, 2024.

[40] Ziyue Wang, Junde Wu, Linghan Cai, Chang Han Low, Xihong Yang, Qiaxuan Li, and Yueming Jin. Medagent-pro: Towards evidence-based multi-modal medical diagnosis via reasoning agentic workflow. *arXiv preprint arXiv: 2503.18968*, 2025.

[41] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv: 2501.19393*, 2025.

[42] Jun Ma, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang. Segment anything in medical images. *Nature Communications*, 15 (1):654, 2024.

[43] Weihao Yu, Chenyang Si, Pan Zhou, Mi Luo, Yichen Zhou, Jiashi Feng, Shuicheng Yan, and Xinchao Wang. Metaformer baselines for vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46 (2): 896 – 912, February 2024.

[44] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv: 1412.6980*, 2014.

[45] X. Liu, H. Liu, G. Yang, et al. A generalist medical language model for disease diagnosis assistance. *Nature Medicine*, 31: 932 – 942, 2025.

[46] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[47] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[49] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.