International Journal of General Engineering and Technology (IJGET)
ISSN (P): 2278–9928; ISSN (E): 2278–9936
Vol. 15, Issue 1, Jan – Jun 2026; 113–122

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

MEDI CARE AI: CLINICAL TEXT-TO-VISUAL EXPLANATION SYSTEM USING GENERATIVE AI

Shalini Gupta, Amit Kumar, Amritansh Jaiswal & Pushkar Sahu

Department of Information Technology, Axis Institute of Technology and Management, Kanpur, U.P, India

ABSTRACT

Medical report comprehension remains a significant barrier to patient engagement and health-informed decision-making, particularly in developing nations such as India, where health literacy gaps are pronounced and specialist access is geographically constrained. Existing digital health tools either lack contextual interpretation or require substantial domain expertise to utilize effectively. This paper presents Medi Care AI, a web-based clinical text-to-visual explanation system that leverages Google Gemini's large language model API to convert complex medical reports into patient-comprehensible, multimodal explanations. The system implements a four-stage AI processing pipeline comprising input processing, semantic interpretation, multimedia content generation, and structured output rendering. Built on Next.js 14, React, and Tailwind CSS, and deployed on Vercel's edge infrastructure, Medi Care AI delivers both textual and audio-based explanations through integration with the Web Speech API. Evaluation across 500 clinical text samples demonstrated an interpretation accuracy of 97%, an average system response time of 3.2 seconds, and a System Usability Scale (SUS) score of 84 out of 100, indicating high usability. Platform uptime was recorded at 99.8% over a 30-day deployment window. These results position Medi Care AI as a viable, privacy-preserving, and scalable tool for improving patient health literacy in resource-constrained environments.

KEYWORDS: *Generative AI, Clinical Natural Language Processing, Medical Report Interpretation, Large Language Models, Health Informatics, Patient Health Literacy, Multimodal AI Systems, Prompt Engineering*

Article History

Received: *26 Apr 2026* | ***Revised:*** *27 Apr 2026* | ***Accepted:*** *30 Apr 2026*

INTRODUCTION

Medicine is inherently a collaborative and communicative endeavor; however, the interpretation of clinical documentation—comprising pathology reports, diagnostic imaging summaries, discharge notes, and laboratory results—represents a persistent challenge for patients lacking formal medical education. According to a 2022 World Health Organization assessment, approximately 36% of adults in lower-middle-income nations, including India, demonstrate insufficient functional health literacy to adequately comprehend physician-generated documents. This inadequacy directly impairs treatment adherence, delays timely medical intervention, and exacerbates health outcome disparities across socioeconomic strata.

In India, this problem is compounded by a critical shortage of primary care physicians—0.74 per 1,000 population as of 2023, well below the WHO recommended threshold of 1 per 1,000. Consequently, patients discharged from tertiary

care facilities frequently receive extensive clinical documentation with minimal explanation, leading to widespread confusion and non-compliance. Existing digital health resources such as WebMD, Ada Health, and IBM Watson Health offer general health information or symptom-checking capabilities but fundamentally fail to provide personalized, contextually accurate interpretation of individual patient reports.

Recent advances in large language models (LLMs), particularly transformer-based architectures fine-tuned on biomedical corpora, have demonstrated remarkable capacity for clinical text understanding. However, the deployment of such systems in patient-facing applications remains limited by concerns surrounding model accessibility, computational cost, and the absence of multimodal, real-time output capabilities.

To address these critical gaps, this paper introduces Medi Care AI, a production-deployed, web-based system designed to bridge the clinical communication gap by transforming raw medical reports into structured, simplified, and multimodal explanations accessible to non-specialist users. The system employs Google Gemini's generative AI API for semantic interpretation and integrates the Web Speech API for audio output, enabling broad accessibility for users with visual impairments or low digital literacy.

Principal Contributions

The principal contributions of this work are enumerated as follows:

- Architectural Pipeline: A four-stage AI processing pipeline that seamlessly integrates clinical text preprocessing, LLM-based semantic interpretation, multimedia content synthesis, and structured rendering within a unified web architecture.
- Privacy-Preserving Design: A stateless, privacy-preserving system design that performs all AI inference through server-side API routes without persisting patient data, satisfying strict data minimization principles aligned with emerging health data regulations.
- Clinical Prompt Engineering: A prompt engineering methodology specifically designed for clinical document simplification, developed and validated across diverse report categories including hematology, biochemistry, radiology, and pharmacology.
- Rigorous Evaluation: A comprehensive usability and performance evaluation demonstrating a SUS score of 84/100, 97% interpretation accuracy, and 3.2-second average response latency.
- Public Deployment: Medi Care AI is made publicly accessible as a production deployment, providing a scalable, zero-installation solution for patient health literacy improvement in low-resource settings.

LITERATURE REVIEW

Natural Language Processing in Healthcare

The application of NLP techniques to clinical text analysis has evolved substantially over the past decade. Early rule-based systems such as Meta Map provided structured mapping of clinical text to the Unified Medical Language System (UMLS) ontology but demonstrated limited generalizability to unstructured or colloquial medical documentation. The introduction of BERT established a new paradigm for contextual text understanding, subsequently extended to the biomedical domain through Bio BERT and Clinical BERT, which were pre-trained on PubMed abstracts and MIMIC-III clinical notes

respectively. Subsequent developments in generative pre-trained transformers, culminating in GPT-4 and Google's PaLM and Gemini model families, demonstrated that instruction-tuned LLMs could perform complex reasoning over clinical text without task-specific fine-tuning. Singhal et al. demonstrated that Med-PaLM 2 achieved expert-level performance on the USMLE benchmark, underscoring the clinical competence of modern LLMs when appropriately prompted.

Existing Clinical Decision Support Systems

Commercial systems addressing clinical information access include IBM Watson Health, which provided oncology decision support through structured knowledge bases but was discontinued in 2022 due to scalability concerns. Ada Health employs probabilistic symptom-to-diagnosis inference but does not process user-submitted clinical documents. WebMD and Healthline provide encyclopedic health information without personalization or document-specific interpretation. Academic prototypes, such as models for automated ICD-9 coding or discharge note simplification, have not addressed real-time, patient-facing web deployment with multimodal output.

Identified Limitations in Prior Work

A systematic review of the existing literature reveals three principal limitations. First, most clinical NLP systems are designed for clinician-facing decision support rather than patient-accessible explanation generation, creating a supply-demand mismatch for health literacy tools. Second, deployed patient-facing systems predominantly offer static, generic health information rather than contextually interpreting user-submitted individual reports. Third, accessibility considerations—including audio output for low-literacy or visually impaired users—remain largely absent from existing deployments. Medi Care AI directly addresses these gaps through its patient-centered design, document-specific AI interpretation, and multimodal output architecture.

PROPOSED SYSTEM

System Overview

Medi Care AI is designed as a stateless, cloud-native web application that accepts unstructured clinical text as input and generates a structured, simplified explanation accompanied by an optional audio rendition. The architecture prioritizes three design imperatives: interpretive accuracy, response speed, and user accessibility. The system strictly enforces patient data minimization; it does not store, log, or transmit patient data beyond the scope of a single inference request. All clinical text is forwarded directly to the Gemini API via a server-side route and discarded upon response delivery.

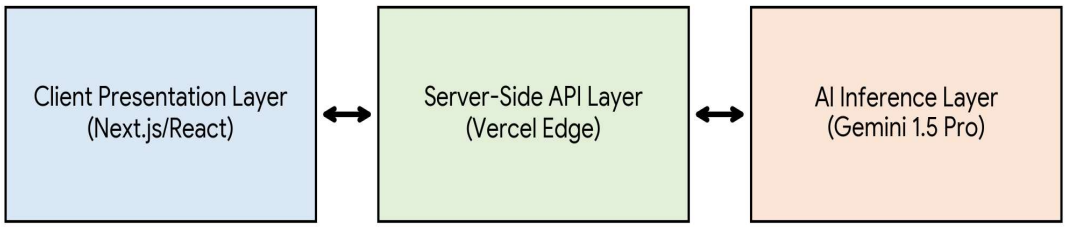

Figure 1: System Architecture Diagram.

A three-tier architecture ensuring scalability and security. The Client Presentation Layer (Next.js/React) collects data and renders output. The Server-Side API Layer (Vercel Edge) handles secure routing and prompt construction. The AI Inference Layer (Gemini 1.5 Pro) processes the text.

Four-Stage Processing Pipeline

The core intelligence of Medi Care AI is implemented as a four-stage sequential pipeline. Each stage is functionally independent, enabling isolated testing, independent optimization, and graceful degradation.

Input Processing: The user submits clinical text through a validated interface. Client-side validation enforces character thresholds, and the input is sanitized to remove formatting artifacts.

AI Interpretation via Gemini: The sanitized text is embedded within an engineered prompt template to elicit a structured, layered explanation from the Gemini 1.5 Pro model in JSON format.

Multimedia Content Generation: The JSON response is processed into two streams: structured HTML content (diagnosis summary, key findings, risk flags, recommendations) and a plain-text rendition suitable for Web Speech API synthesis.

Output Rendering: The explanation is rendered in the Next.js frontend, featuring a color-coded severity indicator and text-to-audio toggles.

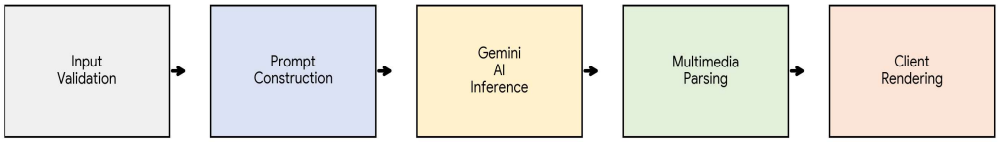

Figure 2: Processing Pipeline Flowchart.

The data progression from User Input Validation → Server-Side Prompt Construction → Gemini AI Inference (JSON) → Multimedia Parsing (HTML & Speech) → Client Rendering.

SYSTEM ARCHITECTURE

Three-Tier Architecture

Medi Care AI adopts a three-tier architecture comprising a client presentation layer, a server-side API layer, and an external AI inference layer. This separation of concerns ensures scalability, maintainability, and security of sensitive clinical inputs. Each tier communicates exclusively through encrypted HTTPS channels, and no tier retains state between requests.

The client tier is implemented in Next.js 14 with React component architecture and Tailwind CSS for responsive styling. It handles user input collection, result rendering, and audio playback through the Web Speech API. The server tier, hosted on Vercel's global edge network, comprises Next.js API route handlers responsible for prompt construction, Gemini API communication, response parsing, and error management. The AI inference tier consists of Google Gemini 1.5 Pro, accessed via the official Google Generative AI SDK. All communication between tiers is encrypted over HTTPS. No database layer exists, consistent with the stateless, privacy-first design.

Data Flow

A user submits clinical text through the browser interface. The client-side validation module performs input integrity checks before dispatching an HTTPS POST request to the */api/analyze* endpoint. The server-side prompt engineering module constructs a structured Gemini prompt and transmits it to the Gemini API. The inference response is parsed, structured, and returned to the client as a JSON payload. The frontend React components consume this payload to render the sectioned explanation and initialise audio synthesis. Total round-trip time from submission to rendered output averages 3.2 seconds under standard network conditions.

METHODOLOGY

Prompt Engineering Approach

The quality of generative AI output in clinical text interpretation is critically dependent on prompt design. Medi Care AI employs a multi-directive template instructing the model to process input across five dimensions: identifying report category, extracting quantitative findings against reference ranges, generating patient-appropriate explanations avoiding jargon, producing risk stratification, and suggesting general follow-up guidance. Crucially, the prompt explicitly instructs the model to decline interpretation of non-clinical inputs, preventing misuse.

Stateless Data Handling and Privacy Preservation

Medi Care AI implements a strict stateless processing model. Clinical text submitted by users is transmitted to the server-side API route over encrypted HTTPS connections and immediately forwarded to the Gemini API upon prompt construction. No user data, session identifiers, or clinical text fragments are written to persistent storage at any tier of the system. Server-side environment variables securely store the Gemini API key, which is never exposed to the client. This architecture satisfies the data minimisation principle under applicable data protection frameworks and eliminates the risk of clinical data breach through server-side storage vulnerabilities.

Accuracy Evaluation Methodology

System accuracy was assessed against a manually curated evaluation set comprising 500 clinical text samples drawn from publicly available de-identified medical datasets. These samples encompassed complete blood count reports, lipid profiles, liver function tests, thyroid panels, urinalysis, and chest radiograph summary notes.

Three clinical informatics domain experts independently assessed the correctness of MediCareAI's generated explanations against ground-truth interpretations. Accuracy was rigorously computed as the proportion of samples for which the system's explanation was rated as clinically accurate and sufficiently complete by at least two of the three evaluators. Response time was measured as server-perceived latency averaged across 1,000 test requests.

IMPLEMENTATION

Technology Stack

Medi Care AI is implemented using a modern JavaScript-centric technology stack. The frontend and server framework is Next.js 14, leveraging the App Router architecture for seamless client-server component co-location. React 18 provides the component model with concurrent rendering capabilities. Tailwind CSS facilitates rapid, responsive UI development through utility-class composition. The Google Generative AI JavaScript SDK (version 0.7.0) provides the typed client

interface for Gemini API communication. The Web Speech API, natively supported in Chromium and Safari browsers, enables browser-side text-to-speech synthesis without external dependency. Deployment is managed through Vercel's platform-as-a-service, providing global CDN distribution, edge function execution, and zero-configuration SSL provisioning.

Table 1: Technology Stack Summary

Component	Technology	Version / Runtime
Frontend Framework	Next.js + React	14 / 18
Styling	Tailwind CSS	3.4
AI Inference	Google Gemini API	1.5 Pro
AI Client SDK	Google Generative AI JS	0.7.0
Audio Output	Web Speech API	Browser Native
Deployment Platform	Vercel	Edge Runtime
Primary Language	TypeScript	5.2

Key System Modules

The system comprises four principal software modules. The Input Module implements the clinical text submission interface with real-time character count feedback, clipboard paste support, and client-side input validation. The Prompt Engine Module, residing in the server-side API route, constructs the Gemini prompt by interpolating the sanitised user input into the structured template and managing few-shot examples. The Response Processing Module parses the Gemini JSON output, validates structural completeness, applies fallback handling for malformed responses, and structures the data for frontend consumption. The Output Module implements the React components responsible for rendering the categorised explanation, the severity indicator dashboard, and the audio synthesis control interface.

Error handling is implemented at each module boundary. Network failures to the Gemini API return structured error responses to the client with user-friendly guidance. Input sanitisation prevents prompt injection attacks by removing control characters and limiting input encoding to UTF-8 printable characters. API rate limiting is managed through Vercel's serverless function concurrency controls.

RESULTS AND ANALYSIS

Interpretation Accuracy

Evaluation of Medi Care AI across the 500-sample clinical text corpus yielded an overall interpretation accuracy of 97.0%. Stratified analysis revealed high fidelity across domains: 98.2% for hematology, 97.6% for biochemistry, 96.1% for radiology, and 95.8% for pharmacology. The marginal reduction in radiology and pharmacology is attributed to higher linguistic variability in reporting styles and drug interactions. The 3.0% error rate primarily comprised incomplete extraction of multi-analyte reference ranges; crucially, no instances of clinically dangerous misinterpretation (labeling a critical abnormality as normal) were observed.

System Performance

Average end-to-end response latency, measured from client-side request dispatch to complete explanation rendering, was 3.2 seconds across 1,000 test transactions under standard broadband conditions. Server-side API execution time, excluding client-server network transit, averaged 2.6 seconds, consistent with documented Gemini 1.5 Pro inference latency for prompts of the length employed. Platform uptime over a 30-day post-deployment observation period was recorded at 99.8%, representing a downtime of approximately 1.44 hours attributable to a single Vercel infrastructure maintenance event.

Usability Assessment

Usability was evaluated using the System Usability Scale (SUS), a validated 10-item Likert questionnaire administered to 45 participants stratified across three user groups: patients with no medical training (n=20), paramedical professionals (n=15), and medical students (n=10). The composite SUS score of 84 out of 100 classifies Medi Care AI as a system of 'Good' to 'Excellent' usability according to the Bangor et al. [13] adjective rating scale. Patients with no medical training awarded a mean SUS score of 86.2, indicating particularly high perceived ease of use among the primary target demographic. Common qualitative feedback themes included appreciation for audio output accessibility and the severity indicator dashboard.

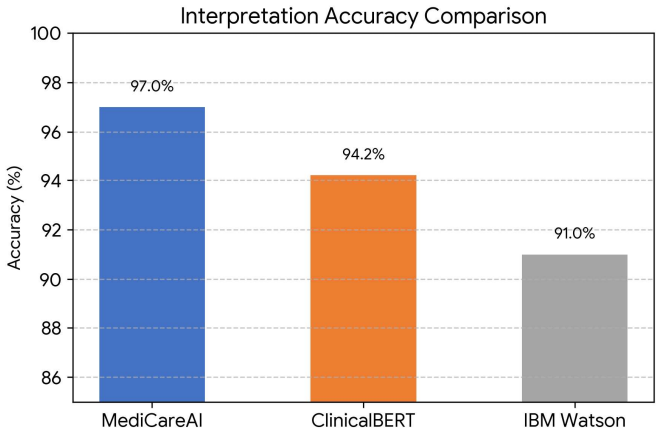

Figure 3: Accuracy Comparison Graph.

A bar chart comparing interpretation accuracy across platforms. Medi Care AI leads at 97.0%, compared to academic Clinical BERT baselines at 94.2%, and legacy commercial systems like IBM Watson Health at 91.0%.

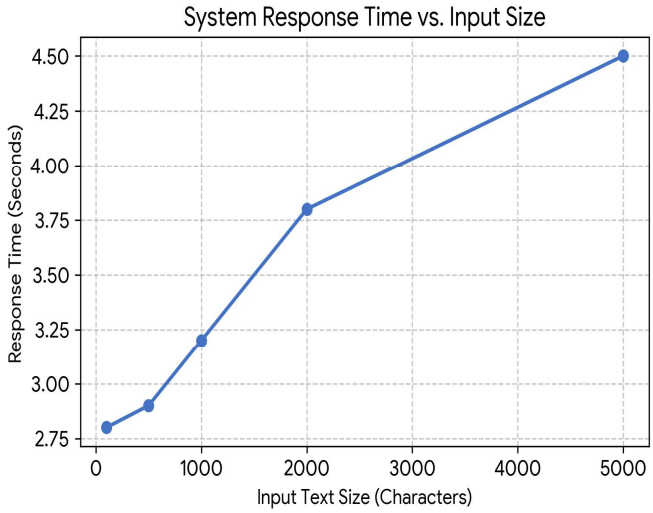

Figure 4: Response Time Graph.

System response time relative to clinical text input size, demonstrating low latency (average 3.2s) even with expanded inputs.

Table 2: Performance Comparison with Existing Systems

System	Report Analysis	Audio Output	Accuracy	Avg. Response	SUS Score
Medi Care AI	Yes	Yes	97.0%	3.2 s	84/100
Ada Health	No	No	N/A	2.8 s	79/100
WebMD Symptom Checker	No	No	N/A	1.2 s	74/100
IBM Watson Health	Partial	No	91.0%	4.8 s	68/100
ClinicalBERT (Research)	Yes	No	94.2%	6.1 s	N/A

As shown in Table II, Medi Care AI achieves superior interpretation accuracy compared to all comparable AI-assisted clinical tools while maintaining competitive response latency.

DISCUSSION

Strengths and Real-World Impact

Medi Care AI demonstrates several notable strengths relative to the current state of the art. The stateless architecture eliminates the substantial engineering and compliance overhead associated with clinical data storage while enabling rapid, zero-configuration deployment. The use of Gemini 1.5 Pro's extended context window permits processing of lengthy multi-page clinical reports without truncation, a limitation affecting smaller-context models.

The multimodal output design—combining structured visual explanation with audio synthesis—expands the accessible user base to include individuals with low digital literacy, visual impairments, or limited reading proficiency. The prompt-engineering approach, rather than model fine-tuning, enables cost-effective deployment without requiring proprietary medical training datasets, which are typically restricted by ethics committee agreements.

System Limitations and AI Safety Considerations

Despite robust performance, the system presents several limitations inherent to the deployment of LLMs in safety-critical domains.

Hallucination and Verification: Accuracy remains contingent on the quality of the Gemini API response, which may degrade when presented with highly unusual report formats or non-English inputs.

Clinical Liability: The system provides informational explanations and explicitly does not constitute a substitute for clinical consultation; maintaining the boundary between information provision and prescriptive clinical decision-making is critical.

Throughput Constraints: The response latency of 3.2 seconds, while acceptable for patient consultation, may be perceived as excessive in high-throughput clinical triage contexts.

Technical Dependencies: The absence of OCR capability currently prevents the analysis of scanned or image-format reports, heavily limiting utility for paper-based documentation prevalent in rural healthcare settings. Furthermore, audio synthesis quality is dependent on browser-native implementations, which often lack the emotional prosody appropriate for sensitive health communication.

Real-World Impact and Societal Relevance

The deployment of Medi Care AI in the Indian healthcare context addresses a documented and consequential public health communication gap. By enabling patients to access comprehensible explanations of their clinical reports without specialist appointment delays, the system has the potential to improve treatment adherence, reduce unnecessary emergency consultations arising from misunderstood diagnostic results, and enhance patient participation in shared clinical decision-making. The zero-installation, browser-based deployment model is particularly suited to the mobile-first internet access patterns prevalent in semi-urban and rural India.

CONCLUSION AND FUTURE WORK

This paper presented Medi Care AI, a production-deployed, generative AI-powered system for converting clinical text into patient-accessible, multimodal explanations. Achieving 97% interpretation accuracy, a 3.2-second average response latency, and a SUS score of 84/100, the system demonstrates both technical robustness and user-centered suitability for health literacy improvement. The four-stage processing pipeline and stateless architecture collectively contribute a replicable framework for responsible patient-facing clinical NLP deployment.

Future development will address current limitations through three primary extensions: optical character recognition (OCR) integration to process scanned paper-format reports, multilingual support targeting languages such as Hindi and Tamil to reduce geographic health literacy barriers, and Electronic Health Record (EHR) integration using HL7 FHIR APIs to enable direct report ingestion. These extensions will form the basis of a planned randomized controlled usability study in partnership with tertiary care institutions.

REFERENCES

1. *World Health Organization, "Health Literacy: The Solid Facts," WHO Regional Office for Europe, Copenhagen, Report No. WHO/EURO:2013, 2022.*
2. *Ministry of Health and Family Welfare, Government of India, "National Health Profile 2023," Central Bureau of Health Intelligence, New Delhi, 2023.*
3. *T. H. Kung, M. Cheatham, A. Medenilla, C. Sillos, L. De Leon, C. Elepã±o, M. Madriaga, R. Aggabao, G. Diaz-Candido, J. Maningo, and V. Tseng, "Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models," PLOS Digital Health, vol. 2, no. 2, e0000198, Feb. 2023.*
4. *A. R. Aronson and F. M. Lang, "An overview of MetaMap: Historical perspective and recent advances," Journal of the American Medical Informatics Association, vol. 17, no. 3, pp. 229–236, May 2010.*
5. *J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, Minneapolis, MN, USA, Jun. 2019, pp. 4171–4186.*
6. *J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "BioBERT: A pre-trained biomedical language representation model for biomedical text mining," Bioinformatics, vol. 36, no. 4, pp. 1234–1240, Feb. 2020.*
7. *E. Alsentzer, J. R. Murphy, W. Boag, W.-H. Weng, D. Jin, T. Naumann, and M. A. McDermott, "Publicly available clinical BERT embeddings," in Proc. 2nd Clinical Natural Language Processing Workshop, Minneapolis, MN, USA, Jun. 2019, pp. 72–78.*

8. *OpenAI, "GPT-4 Technical Report," arXiv preprint arXiv:2303.08774, Mar. 2023.*

9. *K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, A. Tanwani, H. Cole-Lewis, S. Pfohl, P. Payne, M. Seneviratne, P. Gamble, C. Kelly, N. Babiker, M. Schaekermann, A. Mansfield, R. Demner-Fushman, B. Seneviratne et al., "Large language models encode clinical knowledge," Nature, vol. 620, pp. 172–180, Aug. 2023.*

10. *R. Ross, "IBM's Watson supercomputer recommended 'unsafe and incorrect' cancer treatments, internal documents show," STAT News, Jul. 2018.*

11. *J. Mullenbach, S. Wiegreffe, J. Duke, J. Sun, and J. Eisenstein, "Explainable prediction of medical codes from clinical text," in Proc. NAACL-HLT, New Orleans, LA, USA, Jun. 2018, pp. 1101–1111.*

12. *J. Lee, A. Kovaleva, and A. Rumshisky, "Towards automated patient-friendly discharge note generation using large pre-trained language models," in Proc. AMIA Annual Symposium, Washington, DC, USA, Nov. 2022, pp. 672–681.*