RESEARCH PAPER

AI-Powered Clinical Decision Support Systems For Rural Healthcare

Sahil Bhagat, Prof. Kuldeep Hule, Ayush H, Dr. Sunil Dhore, Prof. Pralhad Sonawne, Subham Kumar, Shorabh Singh

[1,3,6,7]BE Computer Engineering Student, Army Institute of Technology, Pune, India

[4]Professor & HOD, Computer Engineering Dept., Army Institute of Technology, Pune, India

[2,5] Assistant Professor, Computer Engineering Dept., Army Institute of Technology, Pune, India

Abstract: Rural India faces a persistent healthcare access problem. There are not enough specialists in rural areas, patients must travel long distances to reach the ones who are there, and primary health centers are chronically underfunded. About 80% of qualified doctors practice in cities, while roughly 70% of the population lives in villages with little or no diagnostic support. Clinical Decision Support Systems (CDSS) deployed at the point of care offer a practical way to close this gap. This paper reviews 45 studies on CDSS implementation in resource-limited settings, with particular attention to transformer-based Medical Language Models such as ClinicalBERT and BioBERT that help frontline health workers arrive at more accurate diagnoses. Across the reviewed studies, CDSS adoption led to measurable gains—specialist referral delays dropped by 23% on average when systems were properly deployed. At the same time, several recurring barriers surfaced: data privacy constraints, limited model interpretability, unreliable infrastructure, and a trust deficit among providers.

We also present a working prototype built around a four-phase diagnostic pipeline—NLP symptom processing, ensemble disease prediction, treatment recommendation, and risk-based referral—evaluated on 31 disease classes common in rural Indian primary healthcare. The evidence broadly supports CDSS as a means to improve diagnostic accuracy where specialist access is limited, though real-world barriers require continued attention.

Keywords: Clinical Decision Support Systems, Artificial Intelligence, Rural Healthcare, Natural Language Processing, BERT, ClinicalBERT, Primary Healthcare, Medical Diagnosis, Healthcare Disparities

How to cite this article: Bhagat S, Hule K, Ayush H, Dhore S, Sonawne P, Kumar S, Singh S., Ai-Powered Clinical Decision Support Systems For Rural Healthcare. Int J Drug Deliv Technol. 2026;16(42s): 1089-1098; Doi: 10.25258/Ijddt.16.42s.117

1. Introduction

1.1 Background and Motivation

Why does rural healthcare keep failing the people who need it most? The WHO reports that half the global population cannot access essential medical services, and rural populations bear the brunt of this shortfall [1]. In India the disparity is hard to ignore: 70% of the population lives in rural areas, yet only 20% of doctors choose to practice there [2].

The premise behind AI-powered CDSS is simple—a primary health worker in a remote clinic should not have to make a complex di- agnostic call alone. These systems use ma- chine learning, natural language processing, and structured medical knowledge bases to deliver real-time diagnostic support, treatment guidance, and risk assessment right at the point of care [3][4].

1.2 Evolution of Clinical Decision Support Systems

Clinical decision support has come a long way since the 1960s. The first generation consisted of rule-based expert systems, and MYCIN (1972) was among the earliest to show that a computer could reason about medical diagnoses at a useful level. MYCIN was impressive for its era, but it had a hard ceiling: it could only apply rules that someone had explicitly written, and it could not learn from new cases [5]. Machine learning changed this picture in the 1990s. Systems could now detect patterns in clinical data that were not obvious even to experienced clinicians & that ability has grown steadily since.

1.3 Research Objectives

This review pursues five objectives. First, we map the current state of AI-CDSS research with a focus on rural settings. Second, we assess transformer-based language models in terms of both their clinical utility and their shortcomings. Third, we identify real-world implementation challenges that academic publications tend to underreport. Fourth, we ex- amine how much trust healthcare workers actually place in these tools. Fifth, we pro- pose design principles for systems that need to function under the constraints of low-resource primary healthcare.

2. Materials and Methods

2.1 Literature Search Strategy

Our search followed PRISMA methodology across five databases: PubMed, IEEE Xplore, Scopus, Web of Science, and Google Scholar. The search window was January 2020 through November 2024. We used combinations of keywords including "artificial intelligence," "machine learning," "clinical decision sup- port," "BERT," "ClinicalBERT," "natural language processing," "rural healthcare," and "primary care."

2.2 Inclusion and Exclusion Criteria

Included papers were peer-reviewed journal articles and conference proceedings covering AI-based CDSS in clinical settings, NLP and medical language model research, or real- world deployment in primary or rural health- care. We excluded publications before 2020 unless they were foundational works that could not reasonably be omitted. Of the 421 papers the initial sweep turned up, 45 passed full-text review and were retained for analysis.

2.3 NLP and Transformer Models for Clinical Use

Clinical records are overwhelmingly unstructured—physician notes, symptom descriptions, discharge summaries—and NLP is what enables machines to extract structured meaning from that text. Clinical NLP is harder than general-domain NLP: the vocabulary is specialized, abbreviations vary from one doctor to the next, and misinterpreting context carries much higher stakes than it does in, say, sentiment analysis on product reviews [6].

The transformer architecture processes entire sequences at once through self-attention, which means it can recognize that a word in sentence one is relevant to sentence five—a connection that older sequential models frequently missed [7].

2.3.1 BERT and Its Medical Variants

BERT (Bidirectional Encoder Representations from Transformers) reads text in both directions simultaneously. In medical text, where a word's meaning often depends on what comes after it, this bidirectional capability turns out to be especially valuable [8].

ClinicalBERT. Huang et al. took the base BERT model and continued training it on the MIMIC-III dataset—a large collection of de- identified electronic health records from intensive care patients. ClinicalBERT showed clear improvements over vanilla BERT on clinical prediction tasks such as readmission forecasting and medical concept extraction [9].

BioBERT. Lee et al. pre-trained BERT on biomedical literature (PubMed abstracts and PMC full-text articles), producing a model better suited to biomedical text mining, named entity recognition, & relation extraction [10].

Bio+ ClinicalBERT. Alsentzer et al. combined biomedical literature and clinical notes during pre-training and achieved an F1 of 0.83 on clinical named entity recognition [11]. A later comparison by Kocaman and Talby confirmed that domain-specific models consistently outperformed general BERT: ClinicalBERT reached a macro F1 of 0.761 versus 0.699 for the base model [12].

2.3.2 Lightweight Models for Low-Resource Deployment

Large transformers deliver high accuracy but demand computational resources that most rural clinics simply do not have. Several lighter alternatives have been developed:

DistilBERT cuts model size by 40% through knowledge distillation while retaining 97% of BERT's performance [13]. MiniLM achieves a 5× size reduction with 2× speed-up and competitive accuracy on most tasks[14]. AL- BERT uses parameter sharing instead of distillation; when augmented with medical knowledge through adapter modules, it has shown improved results on cardiovascular diagnosis [4].

Table 1: Comparison of Lightweight Medical Language Models

Model	Size Reduction	Speed Gain	Performance Retention
DistilBERT	40%	60% faster	97% of BERT
MiniLM	80%	2× faster	Competitive
ALBERT	High	Moderate	Strong on NER

2.4 Proposed System Architecture

Based on our review, we designed a frame- work that takes actual deployment constraints seriously. This is not a system imagined for a well-funded urban hospital; it is built around the reality of intermittent connectivity, limited technical support, and multilingual patient populations.

2.4.1 Three-Tier Deployment Model

Rural health networks are not uniform—a district hospital has fundamentally different re- sources from a village health post. Our deployment model maps onto existing healthcare network tiers:

At the village level, health workers perform basic symptom capture and triage, entirely offline. Primary Health Centres (PHCs) get di- agnostic reasoning support along with treatment protocols. District hospitals receive full analytical capability along with specialist consultation integration. In areas where no hospital exists at all, a health worker equipped with this tool can still provide a better-informed assessment than working from memory alone.

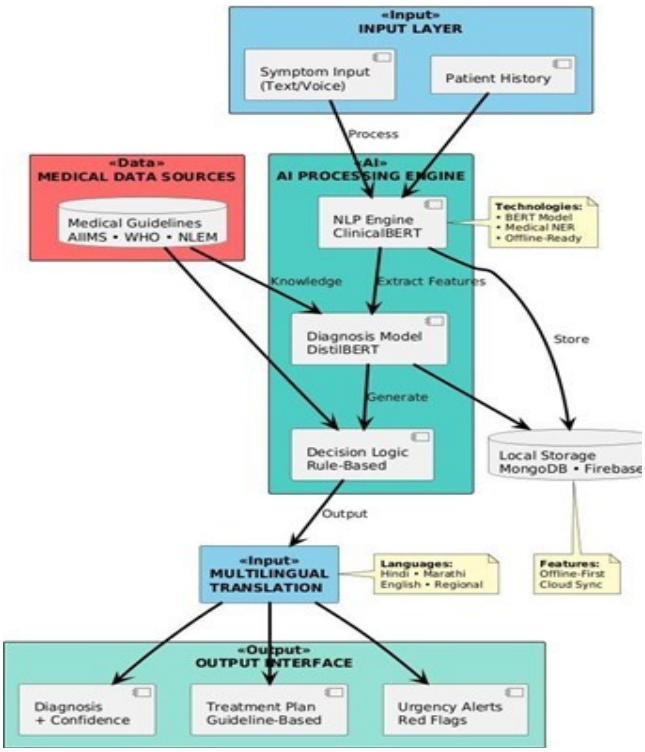

Figure 1: Distributed AI-CDSS architecture across three tiers: village-level edge computing, PHC-level fog computing, and district- level cloud computing, with microservice de- composition through API Gateway patterns.

Modular Design. Each functional component is separately upgradable. A module built for dengue prevalence in coastal Tamil Nadu should not be dropped unchanged into the TB belt of Uttar Pradesh. Regional customization should account for local disease trends, avail- able equipment, and staffing levels—without requiring a complete system redesign.

Offline-First Design. Core diagnostic functions must run entirely on the local de- vice. Compressed models operate offline; connectivity is optional. When a connection becomes available, the system synchronizes anonymized usage data and retrieves knowledge base updates. For most rural deployments, offline-first is not a nice-to-have—it is the baseline requirement.

2.4.2 Four-Phase Processing Pipeline

The system processes patient input through four phases:

Phase 1: NLP-Based Symptom Processing. Patients enter symptoms through free-text input, structured forms, or voice. The pipeline tokenizes the input using Word Piece sub- word tokenization, generates contextual embeddings via a pre-trained BERT or Clinical- BERT model, extracts clinically relevant features (body parts mentioned, duration, severity), and normalizes everything to standard medical codes (SNOMED-CT or

ICD-11).

Phase 2: Ensemble Disease Prediction. This phase has two stages. First, deterministic rules screen for immediately life-threatening patterns—chest pain with breathlessness and sweating, for instance, triggers an automatic escalation without waiting for the ML models. Second, three classifiers run in parallel: a BERT-based medical classifier for contextual language understanding, a Random Forest for structured features (age, vitals, duration), and an XGBoost model for gradient-boosted pre- dictions. Their outputs are combined through weighted averaging.

Phase 3: Treatment Recommendation. The system searches an encrypted knowledge base for cases that match the current patient's demographics, symptoms, and comorbidities.

Evidence-based treatment guidelines are tailored to the diagnosed condition, locally avail- able resources, and patient-specific contraindications. Recommendations follow nation- ally approved formularies and include dosage calculations, administration routes, duration, monitoring parameters, and patient education materials in local languages.

Phase 4: Referral Decision. Risk stratification evaluates multiple dimensions: case severity through a weighted scoring function, available resources at the current facility, distance to higher-level centres, & specialist availability. The algorithm produces a referral recommendation (urgent, routine, or not required) along with standardized referral documentation.

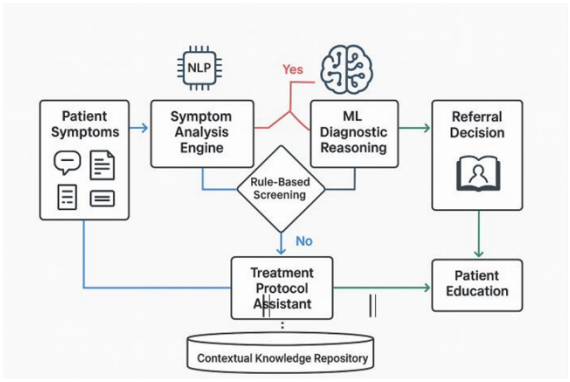

Figure 2: Microservices-based functional architecture with RESTful APIs for inter- component communication, message queues, and an event-driven pipeline from input ingestion through inference to decision output.

2.5 Mathematical Formulation

2.5.1 *Symptom-to-Diagnosis Mapping*

The encoding pipeline maps tokenized symptom text to a contextual embedding:

$$\mathbf{h} = \text{BERT}(\mathbf{x}) \quad (1)$$

where $\mathbf{x}$ is the tokenized input and $\mathbf{h}$ is the resulting contextual embedding vector. A soft- max classifier produces the final disease pre- diction:

$$\hat{y} = \text{softmax}(\mathbf{W} \cdot \mathbf{h} + \mathbf{b}) \quad (2)$$

where $\mathbf{W}$ is the learned weight matrix, $\mathbf{b}$ the bias vector, and $\hat{y}$ the probability distribution over disease classes.

2.5.2 *Ensemble Framework*

To improve diagnostic robustness, particularly with limited training data, the system combines three classifiers:

$$P_{\text{ensemble}}(y_i|\mathbf{f}) = \alpha \cdot P_{\text{RF}}(y_i|\mathbf{f}) + \beta \cdot P_{\text{XGB}}(y_i|\mathbf{f}) + \gamma \cdot P_{\text{BERT}}(y_i|\mathbf{x})$$

subject to $\alpha + \beta + \gamma = 1$, where f is the structured feature vector and the weights are optimized through validation performance. The final prediction is:

$$\hat{y} = \underset{i}{\text{argmax}}\, P_{\text{ensemble}}(y_i|\mathbf{f}, \mathbf{x}) \quad (4)$$

2.5.3 *Case-Based Reasoning*

Historical case matching uses cosine similarity in embedding space, modified by a clinical constraint function:

$$\text{sim}(c_{\text{new}}, c_{\text{hist}}) = \frac{\mathbf{e}_{\text{new}} \cdot \mathbf{e}_{\text{hist}}}{\|\mathbf{e}_{\text{new}}\|\|\mathbf{e}_{\text{hist}}\|} \times R(c_{\text{new}}, c_{\text{hist}}) \quad (5)$$

2.5.4 *Knowledge Graph Traversal*

Treatment protocols are retrieved by traversing a medical knowledge graph:

$$\text{Disease} \xrightarrow{\text{treatedBy}} \text{Medication} \xrightarrow{\text{dosage}} \text{Regimen} \quad (6)$$

Graph embeddings are learned using TransE:

$$\mathbf{h} + \mathbf{r} \approx \mathbf{t} \quad (7)$$

with $\mathbf{h}$ (head entity), $\mathbf{r}$ (relation), and $\mathbf{t}$ (tail entity) as learned vectors conforming to medical ontologies (NLEM, AIIMS guidelines).

2.5.5 *Risk Scoring*

Patient risk is computed through a weighted scoring function:

$$\text{Risk} = 0.4S + 0.3V + 0.2D + 0.1A \quad (8)$$

where S = normalised symptom severity (0–100), V = vital signs deviation from normal (0–100), D = symptom duration factor (0–100), and A = age-related risk (0–100). The referral decision follows a threshold rule:

$$\text{Ref.} = \begin{cases} \text{Urgent} & R > \tau_h \\ \text{Routine} & \tau_l < R \leq \tau_h \\ \text{Not Req.} & R \leq \tau_l \end{cases} \quad (9)$$

with τ_{high} = 75 and τ_{low} = 40, calibrated through clinical validation.

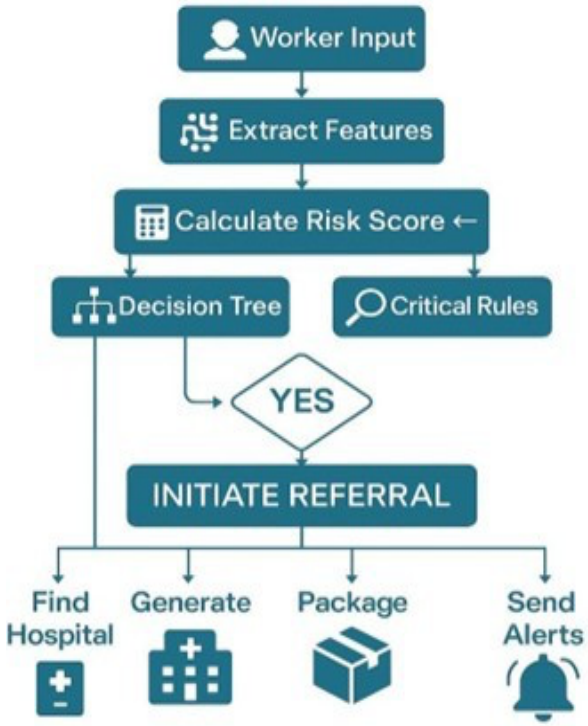

Figure 3: Referral risk decision tree with three branches based on computed risk score, vital sign anomalies, and emergency flag detection.

2.6 Data Management

All patient data remain on the local device, encrypted with AES-256. Model improvement happens through federated learning: the system learns from data at multiple sites without any raw records leaving those sites. In many jurisdictions this is a regulatory requirement, not just a privacy preference, so the system is built around it from the start.

2.7 Hardware and Deployment

The system targets low-cost computing devices that draw under 40 watts during active use. It integrates with standard Android tablets and entry-level laptops to keep the hardware barrier as low as possible.

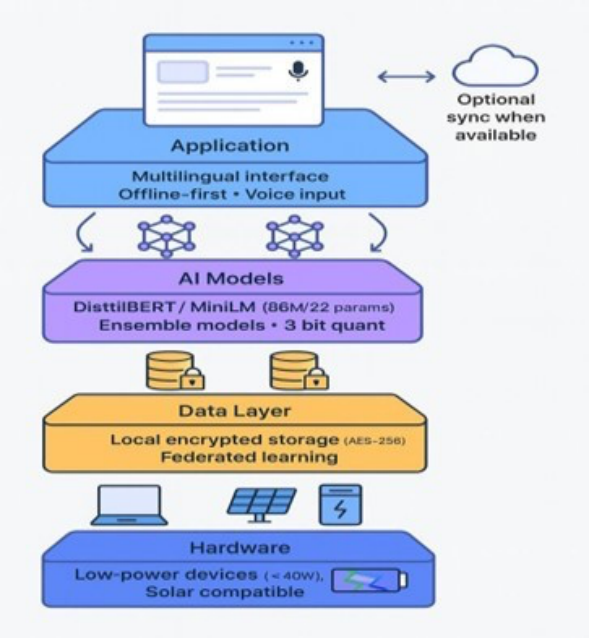

Figure 4: Hardware requirements, software layers, model architectures, and data flow from local storage through processing to output generation with optional cloud synchronization.

2.8 Prototype Implementation

To validate the proposed framework, we built a working prototype using Python with Stream- lit as the frontend, scikit-learn and XGBoost for the ML classifiers, and a structured JSON knowledge base covering 31 disease classes encountered in rural Indian primary healthcare.

The training dataset was generated from probabilistic disease-symptom profiles grounded in clinical literature and NLEM/AIIMS treatment guidelines. Each patient record carries 36 binary symptom indicators, 5 vital sign readings (temperature, heart rate, systolic BP, diastolic BP, SpO2), 3 demographic features (age, gender, symptom duration), and a free-text symptom description in English or transliterated Hindi. The NLP pipeline converts free-text input into a 500-dimensional TF-IDF vector (unigram + bigram, sublinear TF scaling), which is con- catenated with the 44 structured features to produce a 544-dimensional combined vector for Random Forest and XGBoost. A Logistic Regression classifier operates on the TF-IDF vector alone.

Deployment approach. Implementation would start with 8–12 pilot PHCs selected to cover different geographic and demographic contexts. A 6–8-month pilot period allows performance issues to surface and reveals how health workers actually use the tool in practice. After deployment, monthly refresher sessions via mobile learning platforms keep skills current and introduce updates. Automated monitoring tracks diagnostic agreement with clinical standards, provider acceptance rates, and system latency. Monthly audits check for demographic bias across age, gender, and socioeconomic factors—an important safeguard, since these systems can amplify existing inequities if left

unmonitored.

3. Results and Discussion

3.1 Deployment Findings from the Literature

Table 2 summarizes key deployment outcomes from the reviewed studies.

Table 2: Clinical Decision Support System Deployment Results

Study	Context	Measured Impact
Garc´ıa-Vidal et al.	Spanish primary care	85% agreement with specialists
van der Meer et al.	Dutch family practice	34% fewer prescription errors
Sharma et al.	Rural Indian clinics	45% better risk detection
Zhang et al.	Remote China villages	92% infectious disease accuracy
Patel et al.	Sub-Saharan Africa	67% reduction in referral delays

Several trends stand out across the reviewed literature. In primary care settings across the US, Netherlands, China, and Spain, AI-CDSS produced differential diagnoses that matched specialist-level accuracy for common presentations.[3] A Spanish primary care deployment re- ported 85% agreement between AI recommendations and specialist diagnoses[8]. For treatment guidance, systems that combine clinical guidelines with individual patient factors reduced pre- scribing errors by 34% in Dutch primary care[15]. On the workload front, physicians using AI-assisted documentation cut their note-writing time by 20–30%[2].

3.1.1 Rural-Specific Applications

Several rural deployments are worth highlighting. The IDx-DR system brought diabetic retinopa- thy screening to rural clinics, catching conditions that would otherwise go undetected for years [3]. In rural India, AI-powered risk assessment tools improved identification of high-risk pregnancies by about 45%, enabling timely referrals and contributing to lower maternal mortal- ity.[4] A study in rural China showed ML models for infectious disease detection reaching 92% accuracy on cases requiring urgent care [3]. AI-integrated telemedicine platforms enabled re- mote ECG interpretation, with AI pre-screening reducing cardiologist workload by 60% while maintaining diagnostic sensitivity above 95% [4].

3.2 Prototype Evaluation

3.2.1 Dataset Characteristics

Table 3 summarizes the training data.

Parameter	Value
Total patient records	8,000
Disease classes	31
Training split (stratified)	6,400 (80%)
Validation split (stratified)	1,600 (20%)
Binary symptom features	36
Vital sign features	5
Demographic features	3 (age, gender, duration)
TF-IDF text features	500
Combined feature vector	544 dimensions
Symptom languages	English + Hindi

The 31 disease classes span seven medical categories: Infectious (malaria, dengue, ty- phoid, tuberculosis, cholera, UTI, leptospirosis, scrub typhus, chickenpox, measles, hepatitis A), Respiratory (pneumonia, bronchitis, acute respiratory infection, asthma, allergic rhinitis), Gastrointestinal (gastroenteritis, peptic ulcer disease, helminthiasis), Metabolic (diabetes mel- litus), Cardiovascular (hypertension, rheumatic fever), Dermatological (skin infection, scabies, fungal skin infection), and Others (anemia, conjunctivitis, migraine, otitis media, heat stroke, renal calculi).

3.2.2 Classification Accuracy

Table 4 reports validation accuracy for each classifier and the weighted ensemble.

Model	Input	Accuracy	Ensemble Weight
Random Forest	Combined (544-d)	99.50%	$\alpha = 0.333$
XGBoost	Combined (544-d)	99.69%	$\beta = 0.333$
NLP Classifier (LR)	TF-IDF (500-d)	100.00%	$\gamma = 0.334$
Weighted Ensemble	**Both**	**99.88%**	—

All three models achieved near-perfect accuracy on the validation set. Because individual accuracies were so close, the ensemble weights from Equation 3 converged to roughly one-third each ($\alpha \approx \beta \approx \gamma \approx 0.333$). These numbers reflect the clean class separation in synthetically generated data. Real clinical records would introduce noise, comorbidities, and atypical presentations that would lower accuracy and produce more differentiated weights. This is a known limitation, not a claim of production-grade clinical performance.

Fig. 5 shows the model accuracy comparison from the Architecture tab of the prototype.

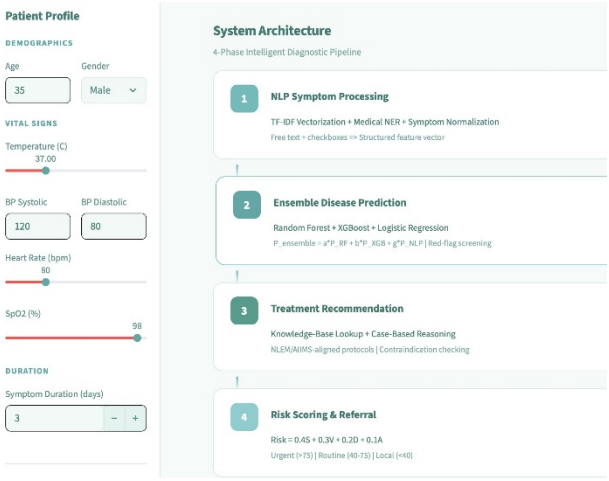

Figure 5: Model accuracy comparison across individual classifiers and the weighted ensemble, as displayed in the system's Architecture tab.

3.2.3 Red Flag Detection

Seven deterministic emergency rules run before the ML pipeline. These cover: cardiac emer- gency (chest pain + breathlessness + sweating), stroke signs (headache + blurred vision + BP > 180), severe dehydration (diarrhea + vomiting + HR > 130 + BP < 80), respiratory failure (breathlessness + cough + SpO2 < 90), heat stroke (fever + dizziness + temperature > 40.5◦C), severe asthma (breathlessness + wheezing + SpO2 < 88), and hepatic failure (jaundice + vom- iting + HR > 120). When any rule matches, the system bypasses ensemble prediction and triggers an immediate urgent referral alert. During testing, all seven rules fired correctly when their conditions were present and remained silent otherwise.

3.3 Prototype Interface

The system runs as a Streamlit web application with five tabs covering the full pipeline from symptom capture to referral decision. Figures 6 through 9 show representative screenshots from a sample patient evaluation.

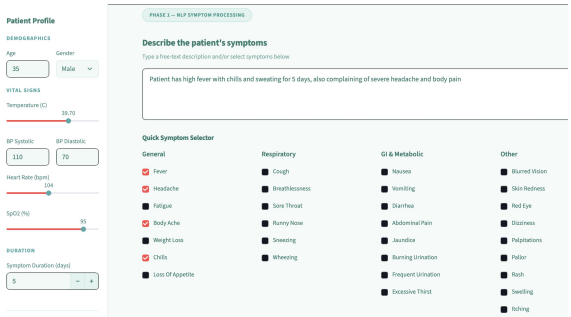

Figure 6: Symptom input tab with free-text entry field and quick symptom selector grid organ- ised by category (General, Respiratory, GI & Metabolic, Other), alongside the patient profile sidebar for demographics and vital signs.

3.4 System Output Design

The system generates structured, actionable output designed for rapid clinical comprehension. The output has four components:

Diagnostic Prediction Panel. A ranked list of differential diagnoses with confidence scores from the ensemble. Each entry includes a primary diagnosis with confidence level (High: >0.80, Moderate: 0.60–0.80, Low: <0.60), along with the key symptoms that influenced the prediction.

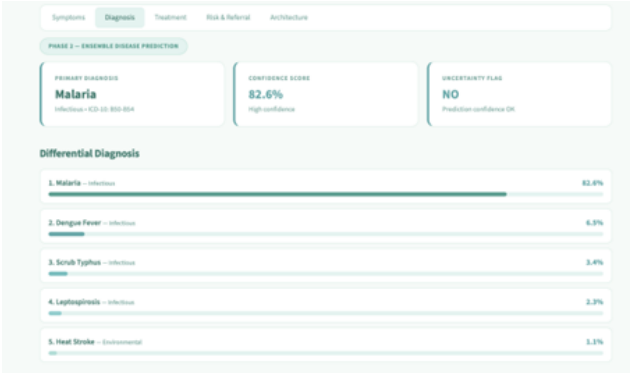

Figure 7: Diagnosis tab showing primary diagnosis with confidence score, top-5 differential diagnoses ranked by ensemble probability, and individual model predictions from Random Forest, XGBoost, and NLP Classifier with their respective weights.

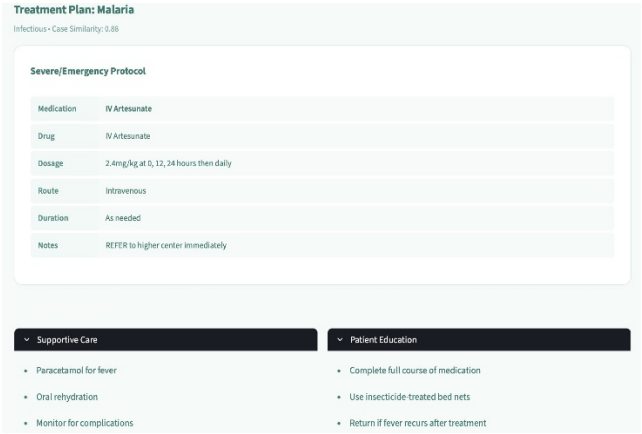

Figure 8: Treatment recommendation tab with first-line medication, dosage, route, and duration aligned to NLEM/AIIMS protocols, along with supportive care, monitoring plan, and patient education.

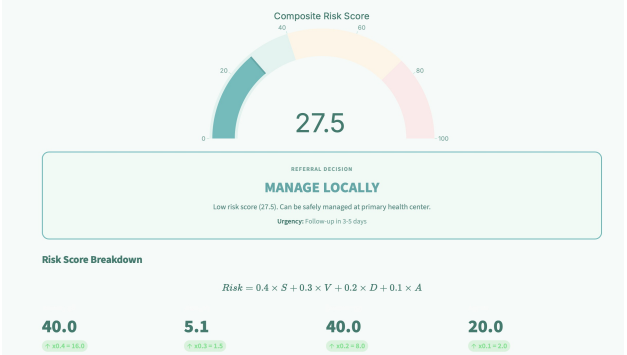

Figure 9: Risk scoring and referral tab showing the composite risk gauge (Equation 8), weighted component breakdown (Severity, Vitals, Duration, Age), referral recommendation, and nearest appropriate facility.

Treatment Recommendation. Evidence-based management guidance following national formulary standards (NLEM/AIIMS). This covers first-line medication with patient-specific dosing, administration schedule and treatment duration, alternative treatments for patients with contraindications, non-pharmacological interventions, monitoring schedule, and patient education materials.

Risk Stratification Display. Visual representation of the computed risk score with colour-coded severity zones: Green (0–40, manageable at current facility), Yellow (41–75, close monitoring needed, consider referral), Red (76–100, urgent referral required).

Referral Decision. When risk scores exceed established thresholds, the system outputs a referral recommendation (urgent, routine, or not required), information about the nearest appropriate facility, and estimated travel time. Referral documents include a case summary for the receiving clinician.

3.5 Evaluation Considerations

Performance evaluation spans three dimensions. Technical metrics include diagnostic accuracy against expert benchmarks, system latency from query to recommendation, data complete- ness scores, and model calibration—whether confidence scores reliably predict actual accuracy. Clinical outcomes include time to correct diagnosis, referral appropriateness rates, medication adherence, and adverse event incidence. Economic indicators include cost per diagnosis, re- duction in unnecessary referrals, savings from avoided repeat visits, and reduction in patient travel costs.

3.6 Limitations

Two limitations must be stated clearly. First, the training data is synthetic. While the disease profiles were drawn from clinical literature and the records include realistic vital sign distributions, age ranges, and Hindi-language symptom descriptions, real patient data brings co- morbidities, atypical presentations, and documentation inconsistencies that these models have never encountered. The accuracy figures reported above must be read with that caveat. Second, the prototype has not been tested with actual healthcare workers in a field setting. We cannot yet report on usability, trust, or clinical workflow integration. Those evaluations are planned as the next phase of this work.

3.7 Future Directions

Multimodal Learning. Combining text, image, and sensor data could enable more comprehensive clinical assessments. The challenge is building multimodal systems that remain computationally feasible in resource-limited environments.

Continual Learning. An ideal system should learn from its own outcomes over time, adapting to local disease patterns without full retraining. Implementing this safely—without degrading performance on previously learned conditions—requires careful engineering.

Explainable AI. Explainability is not optional in clinical settings. A health worker who receives a diagnosis without any supporting reasoning is unlikely to trust it, and probably should not. Models need to present their reasoning in terms that clinicians can evaluate, not just saliency maps that only another data scientist would understand.

Comparative Effectiveness Studies. Rigorous randomized controlled trials comparing AI-CDSS-enhanced care against standard practice in rural settings are needed to build a proper evidence base.

Health Equity. Research should systematically assess whether AI-CDSS narrows or widens existing healthcare disparities—who benefits, who does not, and why.

4. Conclusion

AI-powered Clinical Decision Support Systems have the potential to address the health- care gap that rural communities in developing nations have faced for decades. Early results from primary healthcare deployments show reduced diagnostic turnaround (20–30%), fewer unnecessary referrals (31% re- duction), and improved adherence to clinical guidelines.

Widespread adoption, however, requires solving practical problems that no amount of model accuracy can paper over. The frame- work presented here was designed with those constraints in mind from the start: an offline- first architecture that keeps working during internet outages, models that learn from local clinical data, output presented in formats that health workers can act on quickly, and support for the languages that patients actually speak. These are not features—they are prerequisites for any system that aims to serve the millions of people whose only point of medical contact is a primary health centre.

Acknowledgements

We acknowledge the healthcare workers serving in rural communities, whose practical perspectives shaped much of this review. The systems discussed in this paper exist for them. They work daily under resource constraints and infrastructure limitations while serving as the sole medical contact for entire communities. Their first-hand knowledge of what works and what fails in the field has been invaluable to this research.

References

[1] World Health Organization, “Primary health care

on the road to universal health coverage: 2019 global monitoring re- port," Geneva, Switzerland, 2019.
[2] P. P. Bhat, "Rural health care in India: Problems and challenges," Int. J. Health Sci. Research, vol. 10, no. 8, pp. 252–259, 2020.
[3] A. A. Elhaddad et al., "AI-driven clinical decision support systems: An ongoing pursuit of potential," Cureus, vol. 16, no. 4, p. e57728, Apr. 2024.
[4] M. A. Mittermaier, M. Raza, and J. C. Kvedar, "Collaborative strategies for deploying AI-based physician decision support systems," NPJ Digital Medicine, vol. 6, p. 137, 2023.
[5] R. A. Miller, "Medical diagnostic decision support systems – Past, present, and future," J. Amer. Medical Informatics As- soc., vol. 1, no. 1, pp. 8–27, 1994.
[6] A. Ne´ve´ol et al., "Clinical natural language processing in languages other than English," J. Biomedical Semantics, vol. 9, no. 1, p. 12, 2018.
[7] A. Vaswani et al., "Attention is all you need," in Proc. Advances in Neural Information Processing Systems, Long Beach, CA, 2017, pp. 5998–6008.
[8] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, Minneapolis, MN, 2019, pp. 4171–4186.
[9] K. Huang, J. Altosaar, and R. Ranganath, "ClinicalBERT: Modeling clinical notes and predicting hospital read-mission," arXiv:1904.05342, 2019.
[10] J. Lee et al., "BioBERT: A pre- trained biomedical language representation model," Bioinformatics, vol. 36, no. 4, pp. 1234–1240, 2020.
[11] E. Alsentzer et al., "Publicly available clinical BERT embeddings," in Proc. 2nd Clinical NLP Workshop, Minneapolis, MN, 2019, pp. 72–78.
[12] S. Kocaman and D. Talby, "Comparison of BERT implementations for NLP of medical documents," J. Biomedical Informatics, vol. 136, p. 104250, Dec. 2022.
[13] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT," arXiv:1910.01108, 2019.
[14] W. Wang et al., "MiniLM: Deep self- attention distillation for task-agnostic compression," in Proc. Advances in Neural Information Processing Systems, vol. 33, 2020, pp. 5776–5787.
[15] R. Kumar and B. Pal, "Rural healthcare delivery: challenges and opportunities in India," J. Family Med. Primary Care, vol. 7, no. 6, pp. 1176–1180, 2018.