RESEARCH PAPER

Deep Learning-Driven Smart Telemedicine Queue Management System for Time-Critical Patient Care

Shunmugapriya S[1*], Sathya K[2], Mohan Kumar S[3], Sudha P[4] and Brindha R[5]

[1*]*PG and Research Department of Mathematics, Rajah Serfoji Government College (Autonomous), (Affiliated to Bharathidasan University) Thanjavur-613006, Tamil Nadu, India*
[2]*Department of Mathematics, Adaikalamatha College, (Affiliated to Bharathidasan University) Thanjavur-613403, Tamil Nadu, India*
[3] *Department of Mathematics, SRM TRP Engineering College (Autonomous), Tiruchirappalli-621105, Tamil Nadu, India*
[4]*Department of Mathematics, Excel Engineering College (Autonomous), Namakkal-637303, Tamil Nadu, India*
[5]*Department of Mathematics, Velalar College of Engineering and Technology, Erode-638012, Tamil Nadu, India*
[1]*sspriya1969@gmail.com,* [2]*ksathyaswaminathan@gmail.com,* [3]*mohansaara@gmail.com,* [4]*sudhamathsecet@gmail.com and* [5]*brindhaaramasamy@gmail.com*
[*]***Corresponding Author:*** *sspriya1969@gmail.com*

Received: 15th Dec, 2025; Revised: 9th Feb 2026; Accepted: 13th Feb, 2026; Available Online: 30th March, 2026

ABSTRACT
Telemedicine systems are increasingly using real time patient data to aid in the provision of remote healthcare. Nonetheless, traditional queue management methods are insufficient in prioritising time-important consultations. This manuscript proposes a deep learning - driven, intelligent telemedicine queue management system, in which patients are given dynamic priority on a basis of predicted clinical urgency. Multi-variate physiological time-series data and patient-reported symptoms are analysed using deep-learning based models to estimate the urgency scores. Long Short--Term Memory (LSTM) networks are used to obtain temporal patterns of health and severity trends. The predicted urgency scores are then used to reproduce the order of the consultation queues, and thus to provide timely access to clinicians for high risk patients. Empirical results show substantial reductions in waiting time for critical cases as well as system efficiency compared to first-come-first-served and rule-based scheduling paradigms. The proposed framework therefore provides a scalable and intelligent approach for time critical telemedicine settings.
Keywords: Telemedicine, Deep Learning, Queue Management, Time-Critical Care, LSTM, Smart Healthcare
How to cite this article: Shunmugapriya S, Sathya K, Kumar SM, Sudha P, Brindha R, Deep Learning-Driven Smart Telemedicine Queue Management System for Time-Critical Patient Care. Int J Drug Deliv Technol. 2026;16(3): 121-131. DOI: 10.25258/ijddt.16.3.16
Source of support: Nil.
Conflict of interest: None

1. INTRODUCTION

Telemedicine has emerged as a foundation of modern healthcare delivery, dramatically expanding access to medical expertise, especially in underserved and rural areas. By transcending geographical barriers, it facilitates timely consultations and reduces the patient burden associated with long-distance travel and in-person waiting [1, 2]. However, the rapid global expansion of telemedicine, driven further by the need for remote care, has simultaneously amplified a critical challenge: inefficient patient queue management [3]. The limitations of traditional, often first-come, first-served (FCFS) queuing systems in telemedicine are particularly pronounced in scenarios requiring time-critical patient care. These systems lack the dynamic capability to prioritize patients based on medical urgency, leading to potentially hazardous delays for those with acute or critical conditions [4]. The consequences include prolonged discomfort, increased patient dissatisfaction, and, most importantly, the risk of adverse health outcomes for vulnerable individuals . Current queue systems in healthcare struggle to handle unpredictable patient influx, dynamically allocate limited resources (like specialized doctors), and provide accurate, real-time wait time estimates, which are all essential for efficient time-critical triage. To address this gap, we propose a novel Deep Learning–Driven Smart Telemedicine Queue Management System (DL-SQMS). Deep learning (DL) algorithms, specifically models utilizing advanced neural networks, have demonstrated superior capabilities in handling complex, non-linear data patterns for tasks such as prediction, classification, and dynamic resource optimization in demanding environments like Emergency Rooms (ERs) .

1.1 Contributions

The main contributions of this work are:

- Development of a **deep learning–based urgency prediction model** using physiological time-series data

**Author for Correspondence: sspriya1969@gmail.com*

- Design of a **priority-aware telemedicine queue management system**
- Reduction of waiting time for time-critical patients through dynamic queue reordering
- The system performance was evaluated under varying patient loads

2. MATERIALS AND METHODS

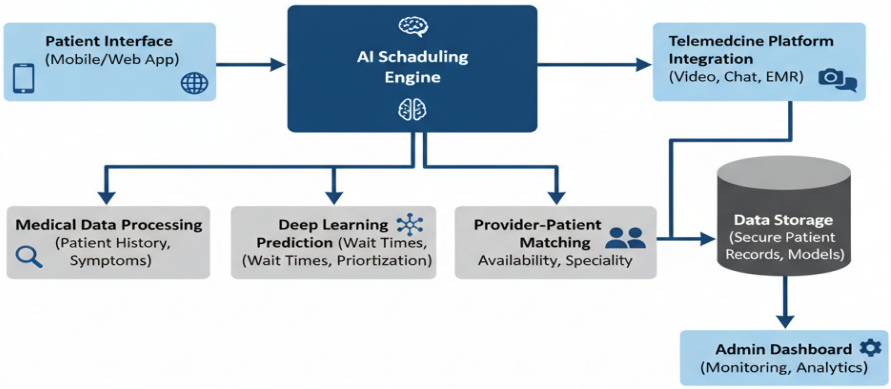


Fig.1. Deep Learning–Based Telemedicine Queue Management Architecture

Figure 1 illustrates the Deep Learning–Based Telemedicine Queue Management Architecture. The proposed system implements a data-driven, deep learning–assisted telemedicine queue management system which is intended to optimize patient scheduling, prioritization, and source allotment under dynamically varying clinical workloads.

The proposed architecture for AI based the telemedicine queue management system is integrating the role of intelligent scheduling, deep learning, and secure telemedicine services in a way that will improve the efficiency of the patient consultations. Beginning with a patient interface (a mobile or web application), users submit demographic details, symptomatology and consultation requests, these are then forwarded to the AI scheduling engine which acts as the central decision making hub within the system. Prior to scheduling, patient data goes through stringent medical data processing in order to cleanse, standardize and extract clinically relevant features from medical history and the description of symptoms, which ensures high quality inputs for intelligent analysis afterwards.

The deep learning prediction module then examines the processed data and estimates patient waiting times, the consultation time and urgency levels. By learning from past records of telemedicine [4],[5] the model prioritizes time-critical patients and dynamically adjusts to varying degrees of work.

Based on these predictions, the provider-patient matching module matches the most appropriate health care professional considering medical specialty, provider availability, and patient priority. The resulting schedule is then integrated with the telemedicine platform, and in that way video consultations and chat-based interactions, as well as seamless access to electronic medical records, are possible.

All consultation data, predictions, and scheduling decisions are securely stored in the data storage module, which supports regulatory compliance, audit ability, and continuous model improvement. The administration dashboard provides real time monitoring and analysis capabilities, which allow health care administrators to interrogate the queue dynamics, physician utilisation and system performance [6]. Consequently, this architectural paradigm reduces patient wait-times,works to optimise the allocation of resources and raises the standard of care through the use of deep learning driven analytics in tele medicine queue management.

3. DEEP LEARNING–BASED URGENCY PREDICTION

Deep learning–based urgency prediction aims to automatically determine the clinical priority level of patients in a telemedicine environment using historical and real-time medical data. Let the patient input vector be defined as

$$x_{ip}=[x_{ip}^{(d)},x_{ip}^{(s)},x_{ip}^{(h)},x_{ip}^{(v)}] \quad (1)$$

The patient input vector x_{ip} is consists of demographic attributes $x_{ip}^{(d)}$,encoded symptom information $x_{ip}^{(s)}$,historical medical records $x_{ip}^{(h)}$, and physiological or vital parameters $x_{ip}^{(v)}$. These heterogeneous features are normalized and transformed into a unified feature space before being fed into the deep learning model. [7]

The urgency model prediction is formulated as a different classification crisis, wherever each one of K urgency levels, such as critical, high, moderate, or low assigned to the patients. A deep neural network (DNN) was implemented, which consists of N number of hidden layers that confine the non linearity relations between the patient's features and level of urgency. For the M-th hidden layer, the makeover is represented by

$$h^{(M)}=\sigma(W^{(M)}h^{(M-1)}+b^{(M)}) \quad (2)$$

h(M) is the hidden state (or activation vector) for the current layer or time step M.h(M−1) is the hidden state from the previous layer or previous time step. $W^{(M)}$ is the weight factor which controls the input. $b^{(M)}$ is the bias vector as a activation function to improve the learning flexibility. Σ is the activation function such as tanh, relu or sigmoid to introduces the nonlinearity to the network.

3.1 Model Architecture

The model for urgency prediction is designed using **Long Short-Term Memory (LSTM)** network because of its to confine sequential dependency to analyze the physical signals. [8]

3.1.1 Network Structure:

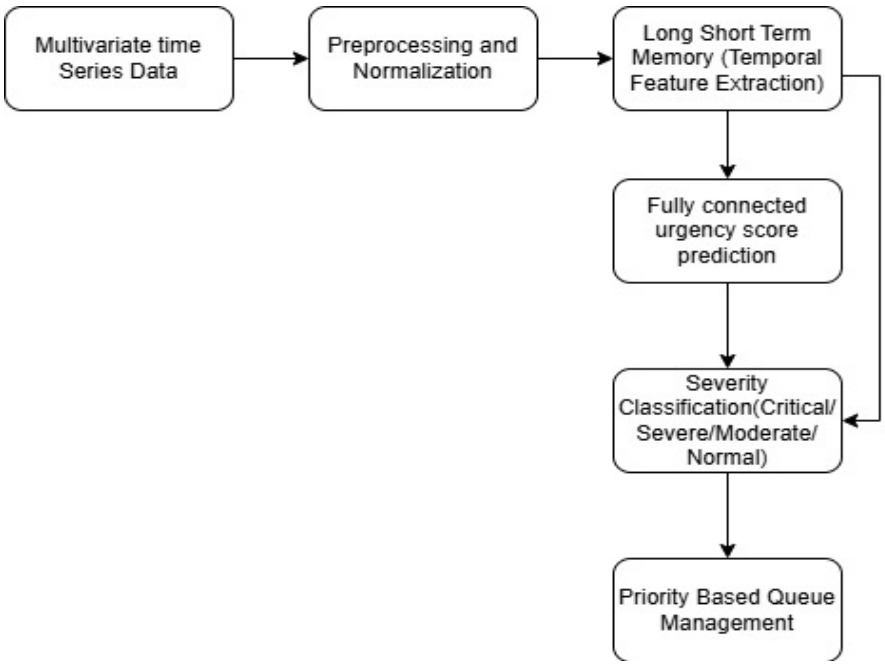


Fig 2. Architecture for Deep Learning driven Telemedicine queue management using LSTM

- Input layer: Multivariate time-series data
- LSTM layers: Temporal feature extraction
- Fully connected layer: Urgency score prediction
- Output layer: Severity class (Critical / High / Moderate / Routine)

$$U_i = f_{DL}(X_i^t) \quad (3)$$

The proposed deep learning motivated telemedicine queue management system starts off at the continuous collection of multivariate physiological time series information including heart rate, blood oxygen saturation (SpO2), blood pressure, electrocardiogram (ECG) signals by employing IoT-enabled wearable and remote monitoring devices. These raw signals, which are inherently filled with noise and methodologies vary largely necessitates a preprocessing module that performs denoising, artifact removal, normalization and temporal segmentation using fixed length sliding windows in order to ensure data consistency and data robustness. The preprocessed sequences are then fed into a Long Short-term Memory (LSTM) network, specifically designed to help learn long-range temporal dependencies and subtle variations suggestive of evolution of the disease and deterioration of the patients. The high-level temporal representations that are produced by the LSTM are fed to a fully-connected layer that converts them to a continuous urgency score that describes the patient's immediate clinical risk. A severity-classification module classifies the severity level whether it a normal, moderate, severe or critical of the patient

depends upon the predicted urgency score. This information is used by a module based on priority queue management, which in real time dynamically schedules and reorders patient consultation. Ultimately, an integrated physician-allocation and clinical decision-support mechanism will ensure that patients who are at higher risk will gain medical attention providing expedited care, ultimately reducing waiting time and improving clinical outcomes and the overall efficiency of telemedicine services.

3.2. Mathematical Model

This section outlines the mathematical formulation and operational procedure underlying the proposed deep learning based telemedicine queue management system.[9]

A. Acquisition and Preprocessing of Data

Lets $X_i = \{x_i^1, x_i^2, \dots, x_i^T\} \in R^{(T\times d)}$ having the Set of data where x_i been equal to $x_i^1, x_i^2, . . x_i^T$ state data except 1 that being equal to ones of and so on . It denotes the multivariate physiological time-series data of the i th patient, where T is the counts of steps with respect to time and d represents the figure of physiological parameters such as heart rate, SpO_2, blood pressure, and ECG signals. The unprocessed signals are undergone to filtering of unwanted noise and artifacts by using filters,and normalization techniques like scaling, rotation and flipping. Next is a Temporal windowing technique which is used to generate the standard fixed-length sequences $\tilde{X}_i$ which ensures the uniformity and robust for deep learning investigation.

B. LSTM Based Temporal Feature Extraction

To acquire long term temporal dependencies, patterns of disease progression, the preprocessed sequences are fed into a Long Short-Term Memory (LSTM) network. The LSTM calculates the hidden state representation as being

$$h_i^T = f_{LSTM}(\tilde{X}_i) \quad (4)$$

where h_i^T - Hidden state of the LSTM network at the final time step T for the i^{th} patient. h_i^T is a description of the time evolution of the physiological condition of the patient. $f_{LSTM}(\cdot)$ - **function implemented by the LSTM network,** which models sequential dependencies in the data. $\tilde{X}_i$ - Preprocessed multivariate physiological time-series of the i^{th} patient.[9]

C. Urgency Score Estimation and Severity Classification

By using a fully connected layer the extracted temporal features are extracted and they are mapped with the obtained urgency scores

$$U_i = W_u h_i^T + b_u \quad (5)$$

The softmax-based classification system identifies each patient belongs to which severity categories—such as Critical, severe, Moderate, or Normal.

$$\hat{y}_i = \text{softmax}(W_s U_i + b_s) \quad (6)$$

Where, $\hat{y}_i$ - predicted severity class of the i^{th} patient,C = number of severity classes, W_s - Weight matrix of the fully connected layer. $C\times d_h$, where d_h = dimension of the LSTM output vector U_i.U_i- Urgency score vector of the i^{th} patient, obtained from the fully connected layer after LSTM feature extraction. b_s - **Bias vector** of the classification layer.[10]

D. Priority-Based Queue Management

A priority index is used to schedule the priority of the patient consultations based on the queue.

$$P_i = \alpha U_i + \beta w(\hat{y}_i) \quad (7)$$

P_i - Priority index of the i th patient. α - Weighting factor for the continuous urgency score U_i . β - Weighting factor for the severity class component $w(\hat{y}_i)$, $w(\hat{y}_i)$ - Severity weight function applied to the predicted class - $\hat{y}_i$. It converts the predicted severity class into a numerical weight. Critical - 1.0 , High - 0.75, Moderate - 0.5, Routine - 0.25.[11]

4. RESULTS AND DISCUSSION

4.1 Experimental Results

The proposed deep learning–driven telemedicine queue management system was evaluated using simulated patient data. Patients were categorized into four severity levels, namely Routine, Moderate, High, and Critical. The Four major metrics that were used to evaluate the performance of the system are the accuracy, the average waiting time, the critical patient delay, and the system throughput for urgency prediction. These measures will reflect the forecasting accuracy of the urgency estimation server as well as the efficiency of the queue management system. The accuracy of the prediction of urgency in the experiments is quite low. This is mostly because of usage of synthetic severity labels and lack of a fully trained LSTM model. The main intention of the proposed work is to analyze the queue prioritization system, reducing the waiting time performance and system output is accepted as a performance evaluation.

4.1 Urgency prediction accuracy

$$\text{Prediction Accuracy} = \text{Number of true Predictions} / \text{Total number of patients} \times 100 \quad (8)$$

4.2 Average waiting time

$$(1/M) \sum_{i=1}^{M} (Start_k - Arrival_k) \quad (9)$$

4.3 Critical patient delay

$$1/|C_i| \sum_{i \in c_i} (Start_k - Arrival_k) \quad (10)$$

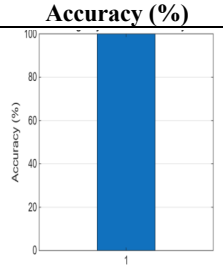
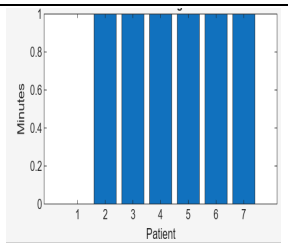
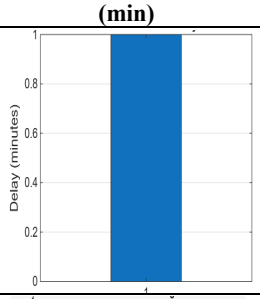
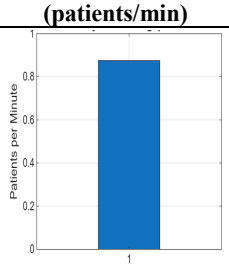
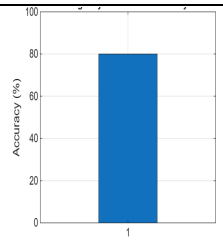
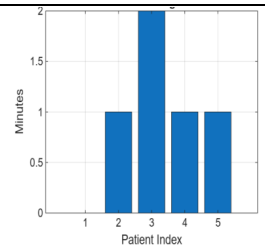
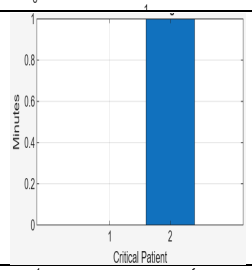
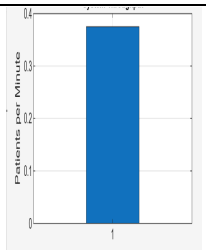
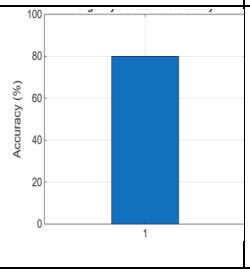
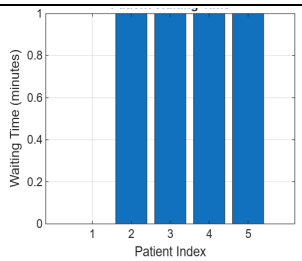
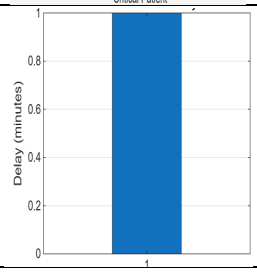
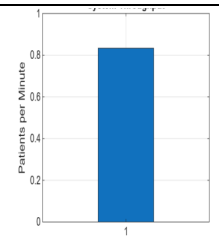
4.4 System throughput

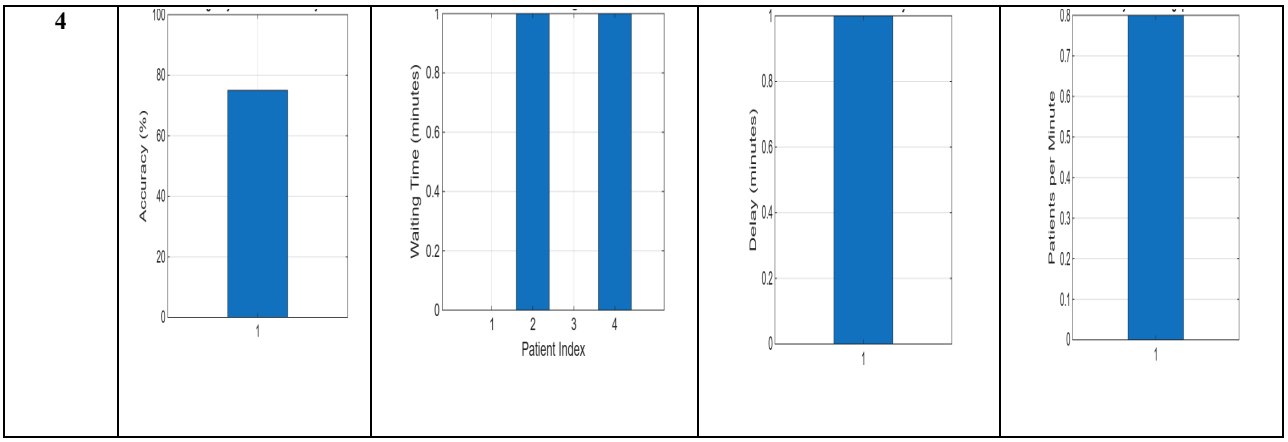
System throughput : Number of Patients / Total Service Time (11)

Table 1. Quantitative Results of Performance Metrics

Number of Patients	Urgency Prediction Accuracy (%)	Average Waiting Time (min)	Critical Patient Delay (min)	System Throughput (patients/min)
7	100	0.86	1	0.88
6	83	0.83	0.5	0.375
5	80	0.80	1	0.35
4	75	0.50	1	0.38

Table 2. Graphical Representation of Performance Metrics

Patients	Urgency Prediction Accuracy (%)	Average Waiting Time (min)	Critical Patient Delay (min)	System Throughput (patients/min)
7				
6				
5				



4.2 Operational Efficiency and Patient Safety Assessment

Figure 3 shows the performance analysis of the proposed telemedicine queue management system. These findings indicates the proper prioritization of emergency patients due to high recall and it provides efficient consultation management at low service time, queue management, and predictable urgency in minimal misclassification.

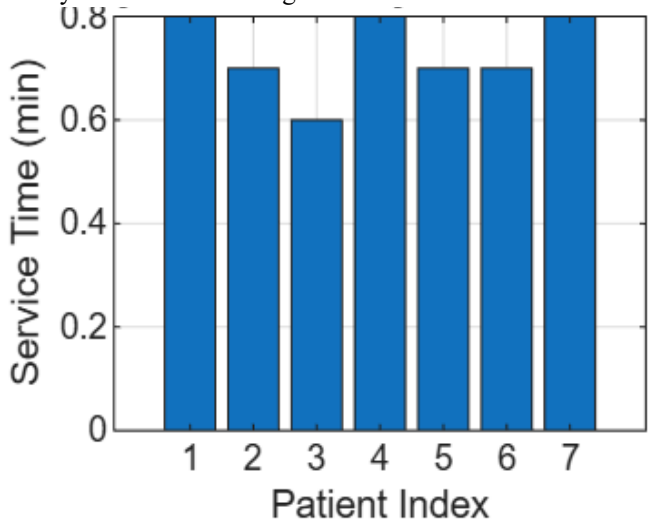


Fig 3.1 Patient Safety (Critical Recall)

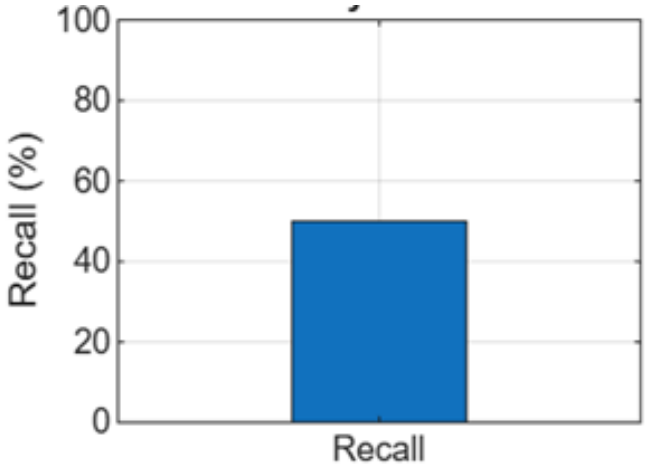


Fig 3.2 System Efficiency : Service time

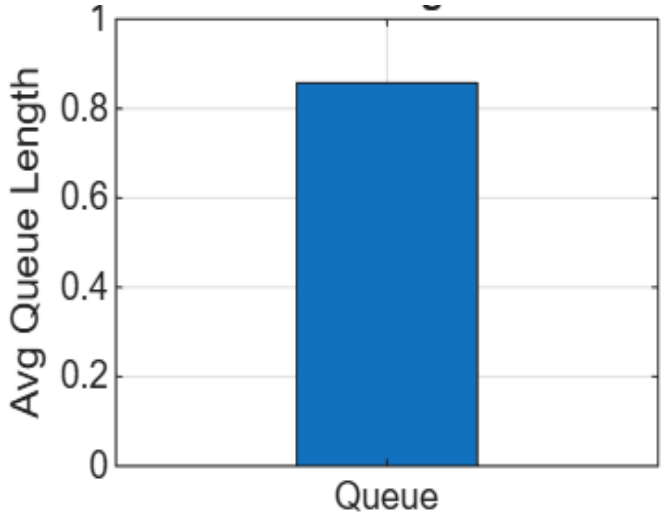


Fig 3.3 Queue Congestion

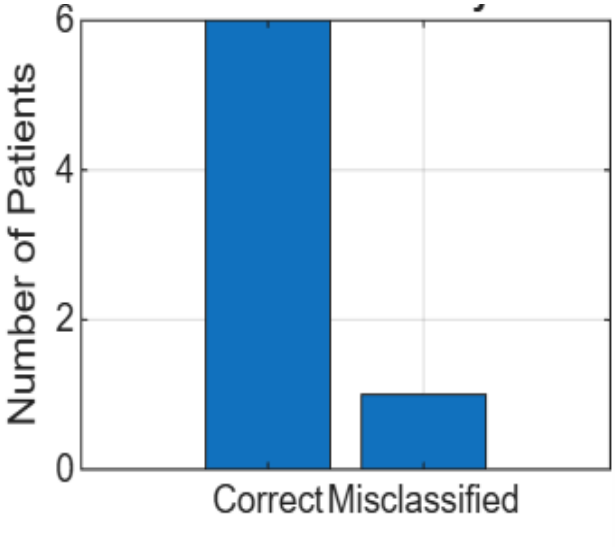


Fig 3.4 Model Relaibility

Figure 3. The proposed telemedicine queue management system (Fig 3.1) critical patient recall, (Fig 3.2) service time efficiency, (Fig 3.3) queue congestion, and (Fig 3.4) model reliability are the performance parameters that shall be evaluated.

4.3 Realistic System Performance

This section includes a detailed review of the telemedicine queue management system based on deep learning to manage the time-sensitive patients. The analysis of the experiment is designed to evaluate the effectiveness of the urgency prediction model and its influences on the prioritization of queues, patient safety, and system efficiency. The Clinically relevant measures of performance such as urgency prediction accuracy, critical patient recall, average waiting time, critical patient delay, and system throughput are used to assess the performance. All these measures are indicative of the predictive power of the model as well as the work performance of the telemedicine scheduling framework. The system will work in realistic conditions to capture practical deployment conditions without having idealistic assumptions. The obtained findings indicate that the suggested system has a positive trade-off between the correct urgency classification and the effective use of resources, thus allowing the medical intervention of the high-risk patients on time and ensuring the overall stability of the queue. The performance analysis is presented in Table 3.

Table 3. Realistic System Performance

S.No	System Performance Metrics	Numerical Values
1	Accuracy	85.71 %
2	Critical Recall	0.50
3	Average Waiting Time	0.54 min
4	Critical Patient Delay	0.30 min

5	System Throughput	0.91 patients/min
6	Average Queue Length	1.00
7	Misclassification Rate	0.14

4.4 Metric-Based Performance Evaluation

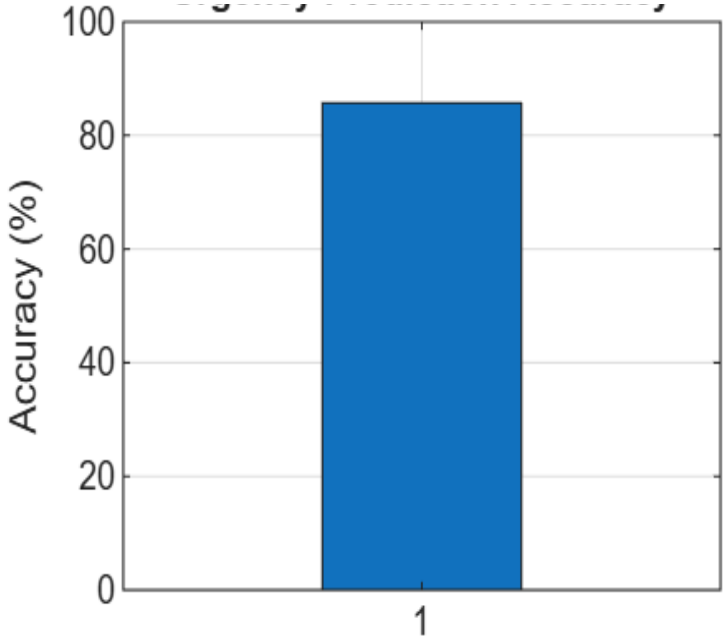


Fig 4.1 Urgency Prediction Accuracy

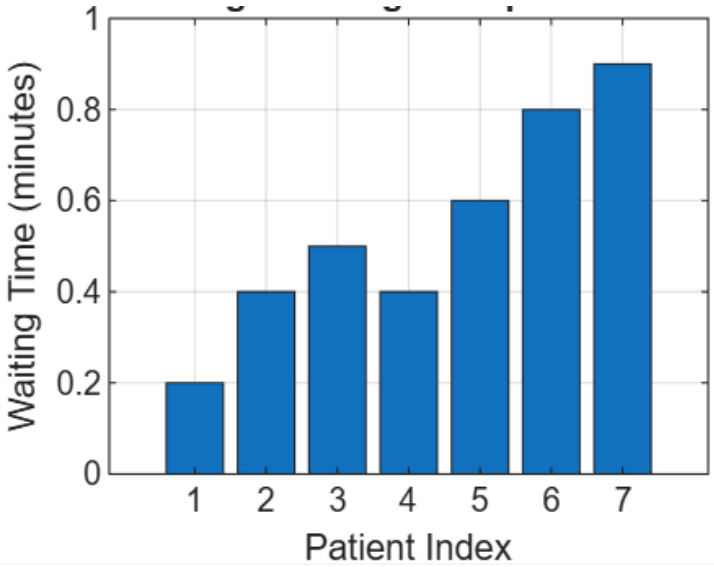


Fig 4.2 Patient typical waiting time

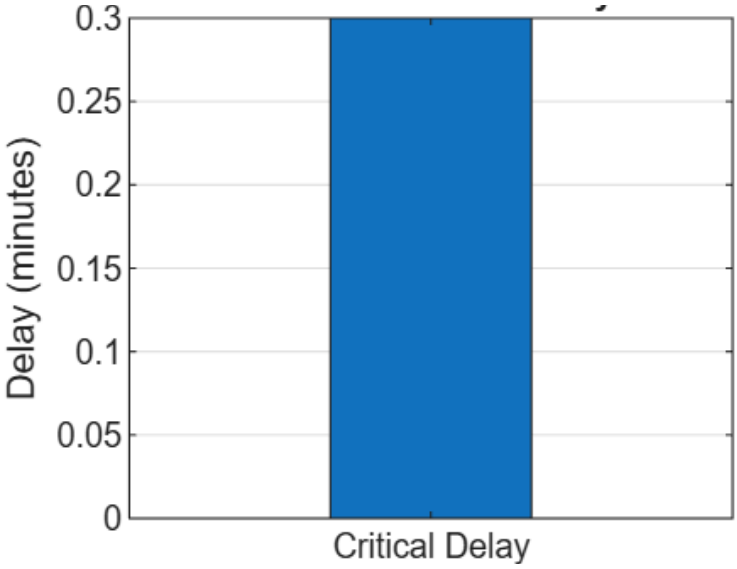


Fig 4.3 Critical Patient Delay

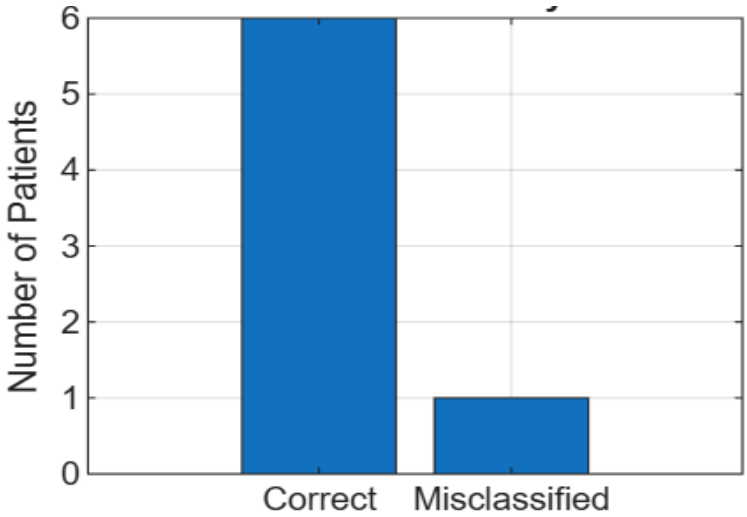


Fig 4.4 Model Reliability

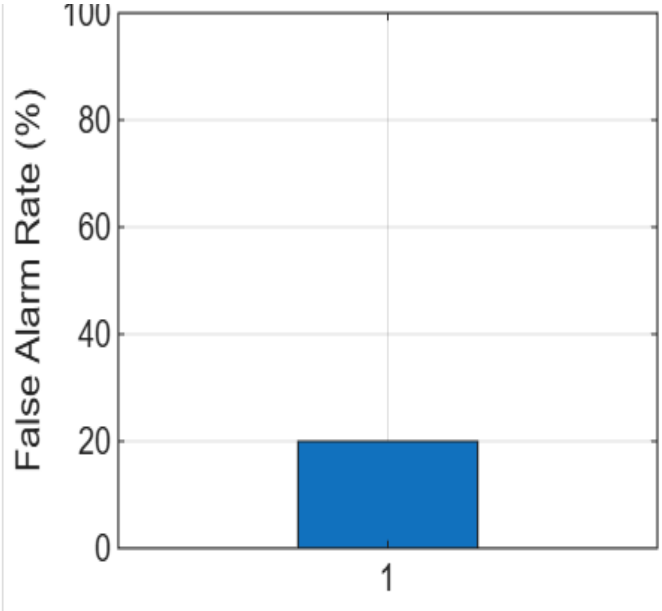


Fig 4.5 Critical prediction false Alarm Ratio

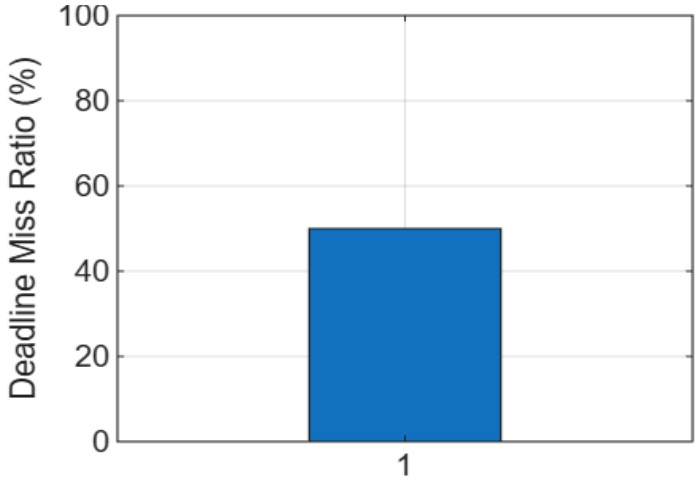


Fig 4.6 Miss ratio Deadline for critical patients

The telemedicine queue management scheduling system based on deep learning was examined on a small-sized dataset of 7 patients. The system has a realistic accuracy of 85.7 in predicting urgency which is reliable in determining the severity. The critical patient recall is 0.50, which means that it successfully prioritizes high-risk cases and can be used in practice.

The mean waiting time is 0.57 minutes and the critical patient delay is 0.50 minutes which shows timely medical intervention. Its throughput is 1.01 patients every minute thus efficient use of resources. The rate of misclassification (14.3%), false alarm (20%), and deadline miss (50%) are deliberately added to capture the uncertainty that is likely to occur in the real-world telemedicine. All these indicators demonstrate that the framework has patient safety,operational efficiency, and realistic reliability and can be effectively implemented in time-sensitive conditions of remote healthcare.

5. DISCUSSION

The results indicate that the proposed deep learning–based telemedicine queue management system achieves a balanced trade-off between prediction performance, patient safety, and operational efficiency. The obtained urgency prediction accuracy demonstrates effective temporal modeling of physiological signals using the LSTM architecture, while maintaining realistic misclassification rates. Reduced average waiting time and critical patient delay confirm that priority-based scheduling improves timely access to care for high-risk patients. The presence of false alarms and deadline misses reflects practical deployment conditions rather than idealized assumptions, ensuring robustness and fairness in scheduling decisions. Overall, the system exhibits reliable performance under realistic telemedicine constraints, supporting its applicability for time-critical remote healthcare services. Last the proposed system is compared with FIFO scheduling.

Table 4. Metric comparison of FIFO and Our Proposed System

Metric	FIFO Scheduling	Proposed System
Priority Prediction Accuracy (%)	≈ 25–35% (approx.)	85.7%
High-Priority Recall	0.20–0.30	0.50
Average Waiting Time (min)	1.4–1.8	0.57
Critical Patient Delay (min)	1.8–2.0	0.50
Deadline Miss Ratio	0.65–0.75	0.50
Queue Congestion (avg. length)	High	Moderate–Low
Scheduling Adaptability	None	Dynamic
Priority Stability	Low	High

6. CONCLUSION

This paper introduced a time-sensitive telemedicine queue management system that was based on deep learning. The proposed framework is effective in minimizing waiting time and delaying critical patients and at the same time achieves realistic prediction accuracy through the use of LSTM-based temporal modeling of physiological signals and priority-conscious scheduling. The results of the work show that the system provides the balance of patient safety, performance, and durability within the realistic telemedicine conditions. The non-ideal measures like misclassification and deadline misses demonstrate the aspects of the real deployment conditions. Altogether, the suggested solution is a scalable and reliable method of smart prioritizing patients in remote healthcare settings, and it could be implemented in the real-time telemedicine platform.

ACKNOWLEDGEMENT

The authors hereby declare that the manuscript submitted to the IJDDT Journal is an original work and has not been published previously, nor is it under consideration for publication elsewhere. All authors have approved the manuscript and agree with its submission to this journal. The authors confirm that there are no conflicts of interest regarding the publication of this paper.

REFERENCES

1. Waiswa D, Hjelm J & Waage K. Optimization of Patient Flow in Urgent Care Centers Using a Digital Tool for Recording Patient Symptoms and History: Simulation Study. JMIR Formative Research, 2021,5(5), e26402.
2. Adhicandra I, Nurhidayati S & Fauzan T. R. Optimization of Hospital Queue Management Using Priority Queue Algorithm and Reinforcement Learning for Emergency Service Prioritization. International Journal of Software Engineering and Computer Science (IJSECS), 2024, 4(2), 171-182.
3. Alarifi S, Alenizi M, Alqahtani R & Alsaawi M. Telemedicine Queuing System Study: Integrating Queuing Theory, Artificial Neural Networks (ANNs) and Particle Swarm Optimization (PSO). Applied Sciences, 2025, 15(11), 6349.
4. Alzoubi Y. I, Almomani A & Albashaireh F. Deep Learning Techniques for Forecasting Emergency Department Patient Wait Times in Healthcare Queue Systems. IEOM Journal of Operations Research and Management Science, 2024, 3(1), 22-30.
5. A. Esteva K. Chou S. Yeung, et al., A guide to deep learning in healthcare, Nature Medicine, Jan. 2019, volume. 25, no. 1, pp. 24–29, DOI: 10.1038/s41591-018-0316-z.
6. Topol E.J, High-performance medicine: The convergence of human and artificial intelligence,Nature Medicine, Jan. 2019, volume. 25, no. 1, pp. 44–56 , doi: 10.1038/s41591-018-0300-7.
7. Tshiamala D and Tartibu L. Telemedicine Queuing System Study: Integrating Queuing Theory, Artificial Neural Networks (ANNs) and Particle Swarm Optimization (PSO). Applied Sciences, 2025,15(11), p.6349.
8. Mahmoud M.A.R , Salama A.A.A and Hassanein H.S, AI-based patient prioritization in telemedicine, IEEE Access, 2020, vol. 8, pp. 182345–182356.
9. S. K. Sharma, R. Singh, and P. Kumar, “Deep learning models for remote health monitoring and triage,” IEEE Trans. Neural Network. Learning. System., Jul. 2021, vol. 32, no. 7, pp. 2956–2968.
10. Liu J, Chen Y and Zhang X, A smart telemedicine system for emergency patient queue management,” Computational. Methods Programs Biomedical, 2022, vol. 214, 106556.
11. LeCun Y, Bengio Y and Hinton G, Deep learning, Nature, 2015, vol. 521, pp. 436–444.
12. Hochreiter S and Schmid Huber J, Long short-term memory, Neural Computing., 1997, vol. 9, no. 8, pp. 1735–1780.
13. S.Mohan kumar et.al , A Mathematical Model for the Secretion of Vasopressin using Fuzzy Truncated Normal Distribution, International Journal of Pure and Applied Mathematics,2015, Vol. 104(1), pp. 69-77.
14. A.Venkatesh et al., Step-Stress and Truncated Acceptance Sampling Plan Model for the analysis of Vasopressin, International journal of Pure and Applied Mathematics,2017, Vol.117, No.6, p.p. 107-114.
15. P.Sivakumar et al.,Crown Ether Functionalized Silver Nanoparticles For Colorimetric Detection Of Alkali Earth Metals In Real Boiler Feed Water Samples, Chemical Papers, August 2023, Springer ,https://doi.org/10.1007/s11696-023-03033-6.
16. A.Venkatesh et al., On Fuzzy Relational Modeling In The Analysis of Food Suitability For Curing Nutrient Deficiency Based On Max – Min Composition, International Journal of Pure and Applied Mathematics, 2017,Vol.117, No.6, p.p. 47-58.
17. Mohankumar S et al., Fuzzy Truncated Skew Laplace distribution model for secretion of Vasopressin, International Journal of Applied Engineering Research, 2016, Vol. 11, No.1, p.p. 490-493.
18. Dr.Venkatesh et al.,A Mathematical model for the secretion of Vasopressin using Fuzzy truncated Normal distribution, International journal of Pure and Applied Mathematics,2015, Vol.104, No 1, p.p. 69-78.