Paper_id_59 Paper_id_59

PAPER_ID_59

Quick Submit

Quick Submit

Karpagam Academy of Higher Education

Document Details

Submission ID

trn:oid:::1:3402760321

Submission Date

Nov 8, 2025, 10:25 AM GMT+5:30

Download Date

Nov 8, 2025, 10:26 AM GMT+5:30

File Name

PAPER_ID_59.docx

File Size

1.9 MB

5 Pages

2,662 Words

16,280 Characters

89% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

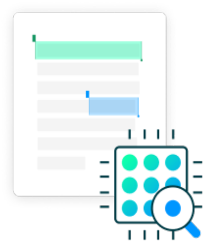
20 AI-generated only 89%
Likely AI-generated text from a large-language model.

0 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.



False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

Data-Driven Optimization of Hospital Operations Using Predictive Analytics

Sudharson D
Associate Professor
Artificial Intelligence and Data Science
Kumaraguru College Of Technology
Coimbatore, Tamil Nadu
sudharsondorai.ads@kct.ac.in

Dr. Sakthi Govindaraju
Professor
School of Computer Science and Engineering
Galgotias University, Greater Noida
sakthihit@gmail.com

Dr. K Vignesh
Assistant Professor
Dept of Management Sciences,
Kumaraguru College of Technology Business School,
Coimbatore, Tamil Nadu
drkvignesh1980@gmail.com

Magudesh S
UG Scholar
Artificial Intelligence and Data Science
Kumaraguru College Of Technology,
Coimbatore, Tamil Nadu
magudesh.22ad@kct.ac.in

Charaneesh A P
UG Scholar
Artificial Intelligence and Data Science
Kumaraguru College Of Technology
Coimbatore, Tamil Nadu
charaneesh.22ad@kct.ac.in

Dhayanithimaran A
UG Scholar
Artificial Intelligence and Data Science
Kumaraguru College Of Technology
Coimbatore, Tamil Nadu
dhayanithimaran.22ad@kct.ac.in

***Abstract*—Hospitals are often busy, with overflowing emergency rooms and staff shortages. This issue can impact patient care. Our system helps by using hospitals' past data to predict future problems before they happen. The system predicts busy periods and identifies delays or overflow of patients. Using advanced analytics, the system predicts busy periods and identifies areas that are going to be overcrowded. All the information is displayed in the simple dashboard, giving staff knowledge about the availability of beds and staff. It reduces the ER wait times and improves patient flow. Data-driven tools can help hospitals react to problems and actively prevent them, creating a safer and more efficient environment for everyone.**

Keywords—Predictive analytics, hospital operations, ER overcrowding, Machine learning, Real-time dashboards.

I. Introduction

Hospitals operate in a fast-paced environment where each minute can make a huge difference between life and death. Still, there are a few problems that need to be addressed: overcrowded emergency rooms, time-taking patient admission processes, and unavailability of certain resources. These issues not only create difficulty for hospital staff and budgets but can also directly impact patient health and safety.

In today's digital era, enormous amounts of operational data are collected, including admission records, discharge logs, bed availability, staff schedules, and more. Yet, most of this information is not used, which implies the fact that hospitals react to problems as they happen rather than prevent them.

Our research addresses this challenge by combining predictive analytics with real-time visual dashboards to give hospital administrators a clear view of operations. By forecasting patient surges, identifying problems, and simulating resource changes before they are made, hospitals can move from problem management to proactive planning, delivering better care, reducing costs, and ensuring smoother operations.

II. LITERATURE AND RELATED WORK

Hospitals often face operational challenges like overcrowded emergency departments (ED), slow patient discharges, and inefficient resource usage, which can disrupt care and increase costs. Research tells us that predictive analytics and machine learning help tackle these problems effectively.

For example, a 2022 study found [13] that predicting ED patient arrivals works better when factors like holidays and weather are included, outperforming older techniques like time-series models. This kind of forecasting, using tools like Prophet or ARIMA, helps hospitals to plan staffing efficiently for handling busy periods. Similarly, a 2021 study showed [14] that machine learning predicts ED arrivals more accurately than older methods, especially for short- and medium-term planning.

Beyond managing arrivals, predicting when patients can be discharged is important to free up beds and reduce the delay.

A study used hospital records [15] to identify patients likely to discharge within 24 hours, speeding up bed turnover. Recently, one study used machine learning [16] to spot delays in discharges caused by follow-up care's needs, showing how explainable models can help hospital staff make faster decisions. These studies support using tools like Random Forest to improve discharge planning and bed management.

Clustering techniques like K-Means, are also proving useful. A 2020 study showed [17] how these methods can highlight overcrowded spots in a hospitals, letting the staff focus in fixing problem areas like specific wards or units. To bring these insights into life, hospitals are turning to visual dashboards. A 2018 study described [18] an ED dashboards that gives staff quick, real-time updates while keeping the patients information private. Today's hospital goes further, combining real-time bed tracking with predictive analytics to reduce wait times and save more money.

Recent studies further highlight this trend. Deep-learning models like N-BEATSx can predict Emergency Department (ED) patient boarding six hours in advance using only non-clinical data like weather and holidays. This model showed great potential for proactive capacity planning. Other machine learning models—TSiTPlus for hourly forecasts and XCMPlus for daily forecasts—accurately predicted ED waiting counts, helping hospitals manage demand fluctuations effectively.

A 2024 study in BMC Medical Informatics and Decision Making applied feature engineering and machine learning (e.g., XGBoost) across 11 international ED datasets to predict daily arrivals [4, 228]. The models achieved MAPE between 5%–14%, outperforming traditional methods and proving the effectiveness of engineered features [229]. Interactive real-time decision-support dashboards have also been deployed, such as at Johns Hopkins during COVID-19. This tool integrated live data, predictive analytics, and optimization, helping administrators make evidence-based decisions during surges.

A 2025 PubMed study used Random Forest, ANN, and XGBoost [1, 2] to identify patients likely to experience delayed discharge (ALC patients) based on admission-time features. The XGBoost model achieved an AUC of 0.97, showing that early detection of discharge risks can significantly aid in flow optimization. These uses of predictive models align with discharge-delay and redeployment forecasting, emphasizing wellness through early action.

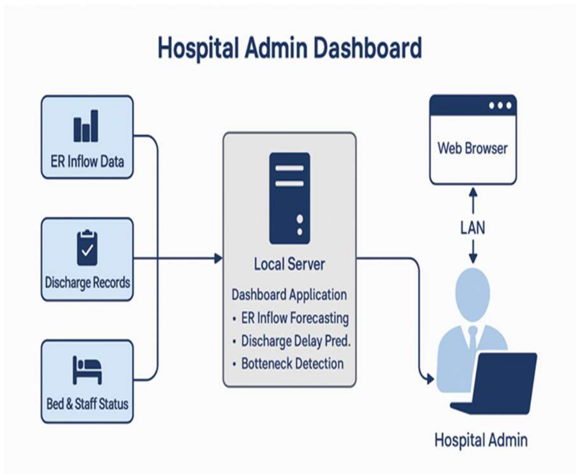
However, there are still challenges. Most studies focus on single hospitals, and few combine forecasting, discharge prediction, and overcrowding into one single system. Additionally, using cloud-based tools can raise issues about data privacy and connectivity. The proposed system in this research aims to solve these kind of issues by creating a framework that integrates forecasting, machine learning, and clustering into a real-time dashboard. Importantly, it runs on the hospitals local network, keeping data secure and accessible without relying on the cloud.

III. METHODOLOGY

The development of the proposed hospital management system followed a structured, step-by-step approach to ensure both practicality and accuracy. We began by collecting historical hospital data relevant to our objectives — including emergency room admission logs, discharge records, bed occupancy statistics, staff rosters, and patient transfer details. This raw data was cleaned to remove inconsistencies, fill missing values, and standardize formats, ensuring it was reliable for analysis.

Once the dataset was prepared, different machine learning models were applied for specific purposes: Time-series forecasting using Prophet and ARIMA was used to predict short-term patient inflow to the ER, capturing recurring patterns and peak periods. Random Forest classification was implemented to identify patients at higher risk of discharge delays, based on their treatment type, length of stay, and other clinical indicators. K-Means clustering was used to detect departments prone to overcrowding by grouping them based on patient load, staff availability, and bed turnover rates.

The outputs from these models were integrated into a real-time monitoring dashboard. This dashboard displays current ER bed availability, forecasted patient demand, staff allocation status, and congestion alerts. Additionally, a built-in simulation tool allows hospital administrators to test different operational scenarios—such as adding more beds or reallocating staff—before making actual changes, helping them choose the most effective strategy. For security and operational reliability, the system was designed to run entirely within the hospital's local network, eliminating the need for internet connectivity and safeguarding sensitive patient information (see Fig. 1).



(see Fig. 1).

IV: SYSTEM ARCHITECTURE

The system's architecture is meticulously designed as a secure, on-premise solution that operates entirely within the hospital's local network. This on-premises model ensures robust data security, maintains patient confidentiality, and guarantees high availability without reliance on internet connectivity.

The core of the architecture, as illustrated in Fig. 1, is a **Local Server** that runs a dual-pipeline processing engine. This design simultaneously handles real-time data for immediate operational awareness and batch/stream data for advanced predictive analytics.

When new operational data—such as **ER Inflow Data**, **Discharge Records**, and **Bed & Staff Status** updates —is generated, it is fed into both pipelines concurrently:

Pipeline 1: The Real-Time Visualization Pipeline:

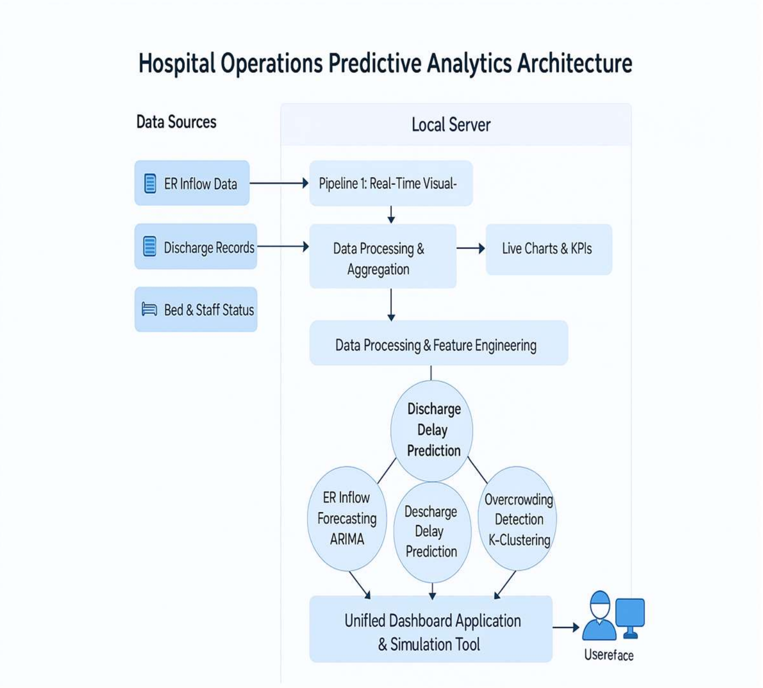
- **Purpose:** To provide an immediate, live snapshot of the hospital's current operational status.
- **Process:** This pipeline is optimized for high-speed, lightweight data processing. As data streams in, it undergoes immediate transformation and aggregation to update key performance indicators (KPIs). This involves simple calculations like counting current ER admissions, tallying available beds, and confirming staff on-duty.
- **Output:** The processed data is fed directly to the visualization layer of the dashboard. This pipeline is responsible for powering the live charts and "current status" alerts, such as "ER Bed Availability" and "Staff Allocation Status," giving administrators an accurate, up-to-the-minute view of what is happening *right now*.

Pipeline 2: The Predictive Modeling Pipeline:

- **Purpose:** To analyze historical and current data to forecast future events, identify risks, and detect complex patterns that are not visible in the real-time view.
- **Process:** This pipeline is more computationally intensive and involves several key stages:
  - **Data Preprocessing:** Raw data from the hospital logs is collected, cleaned to remove inconsistencies, fill missing values, and standardize formats. This stage also includes feature engineering, where new variables (e.g., time of day, weekday, holiday) are created to improve model accuracy.
  - **Model Execution:** The prepared data is then channeled to the appropriate machine learning models as described in our methodology:
    - **Prophet and ARIMA:** These time-series models use ER admission logs to perform **ER Inflow Forecasting**, predicting patient surges and peak periods.
    - **Random Forest:** This classification model analyzes patient records and clinical indicators to run **Discharge Delay Prediction**, flagging patients at high risk for a delayed discharge.
    - **K-Means Clustering:** This model groups data on patient load, staff availability, and bed turnover to perform **Bottleneck Detection**, identifying departments or units that are becoming prone to overcrowding.
- **Output:** This pipeline generates predictive insights—such as "Forecasted Patient Demand," "Congestion Alerts," and "Discharge Delay Risk" scores.

Presentation Layer: The Unified Admin Dashboard:

- **Purpose:** This is the convergence point where the outputs of both pipelines are presented to the end-user in a single, coherent interface.
- **Access:** A **Hospital Admin** accesses the system via a standard **Web Browser** connected securely to the internal **LAN**.
- **User Experience:** The admin is presented with a dashboard that seamlessly blends information. They can view the real-time charts from Pipeline 1 (e.g., "Current Bed Occupancy") right next to the predictive alerts from Pipeline 2 (e.g., "Forecasted Bed Deficit in 4 hours"). This dual-data approach allows them to not only react to current problems but also proactively plan for future challenges. Furthermore, the dashboard integrates a **simulation tool**, allowing managers to test "what-if" scenarios (like reallocating staff) based on the predictive forecasts before committing to a decision.



V. EXPERIMENTAL SETUP

To test how well our proposed predictive analytics system could improve hospital operations, we set up an experiment based on previous research about emergency department (ED) forecasting, discharge predictions, and real-time dashboards. Here's a breakdown of what we did, explained in a clear and approachable way:

Dataset:

We worked with anonymized records from a multi-specialty hospital, like datasets like MIMIC-IV [12] and those in ED forecasting studies [13, 283]. The data included:

- Logs of when patients arrived, their urgency level, and how they got to the hospital.
- Discharge records showing planned versus actual discharge times and reasons for delays.
- Bed availability reports for different wards.
- Staff schedules, showing who was working when.
- Data on how patients moved through different hospital departments.

Data Preprocessing:

We removed duplicate records and fixed errors in timestamps. We made sure dates and categories (like urgency codes) were in consistent formats. We also added useful details, like whether it was a weekday, holiday, or busy season, inspired by related research.

Predictive Models:

The models were chosen based on successful implementations in prior research:

- **Prophet and ARIMA** for ER forecasting.
- **Random Forest** for discharge delay prediction
- **K-Means Clustering** for identifying overcrowded areas.

Dashboards and Simulation Environment:

We created a real-time dashboard to show key hospital metrics like bed availability, ER demand, staff schedules, and alerts for overcrowding. It also included a "what-if" tool, letting administrators test different staffing or bed allocation plans before applying them. The system was securely hosted on the hospital's local network, similar to how some leading hospitals deploy AI tools.

Evaluation Metrics:

- **Forecast Accuracy:** Checked how accurate our ER predictions were using metrics like Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE).
- **Discharge Predictions:** Evaluated how well we predicted delays using metrics like Area Under the Curve (AUC), Precision, Recall, and F1-score.
- **Hospital Improvements:** Measured changes in ER wait times, patient flow, and bed usage before and after testing the system.

VI. CONCLUSION

Hospitals undergo many challenges on a daily basis- over crowded emergency rooms, time taking patient admission process and unavailability of certain resources. These issues not only become a sole reason for increasing operating costs but also puts health and safety of patients at stake. In this study, we introduce a practical, data-driven system designed to help hospitals attain maximum efficiency by making faster and better decisions.

We analyzed past hospital records such as admission rates, discharge summaries and resource usage and have applied advanced prediction methods : Prophet and ARIMA to predict busy time periods, random Forest for discharge delays, and K-means clustering to predict areas that may cause overcrowding. The results are presented on a real-time visual dashboard that shows live updates on critical hospital data like bed availability, ER load and staff allocation. A built-in simulation tool let's managers test "what-if" scenarios before applying changes, helping them plan ahead with knowledge of what may happen ahead.

To take care of the patient data security and confidentiality, the system runs entirely on the hospital's own internal network without dependency on the Internet. At a multi-specialty hospital it reduced emergency room waiting time periods, improved patient movement to respective wards, optimized staff development and improve overall efficiency. This research shows how predictive analytics, combined with proper choice of visual tools, can make hospitals from only solving the problem to ultimately preventing them which at last saves time, money and lives.

VII. References

[1] C. Duckworth, et al., “Predicting onward care needs at admission to reduce discharge delay using explainable machine learning,” Scientific Reports, vol. 15, p. 16033, 2025.

[2] M. Pahlevani, et al., “Developing a decision support tool to predict delayed discharge from hospitals using machine learning,” BMC Health Services Research, vol. 25, p. 56, 2025.

[3] J. Wei, et al., “Predicting individual patient and hospital-level discharge using machine learning,” PMC, 2024.

[4] B. M. Porto, and F. S. Fogliatto, “Enhanced forecasting of emergency department patient arrivals using feature engineering and machine learning,” BMC Medical Informatics & Decision Making, vol. 24, p. 377, 2024.

[5] K. S. Ho, et al., “Accurate forecasting of emergency department arrivals with internet search index and machine learning,” PMC, 2022.

[6] M. A. C. Vollmer, et al., “A unified machine learning approach to time series forecasting of emergency department demand,” arXiv, 2020.

[7] F. M. Caldas, and C. Soares, “A Temporal Fusion Transformer for long-term explainable prediction of emergency department overcrowding,” arXiv, 2022.

[8] “Innovations in discharge prediction : Machine-learning-based hospital discharge predictions can support rounds,” BMJ Innovations.

[9] “Forecasting ED occupancy with advanced machine learning models 24 hours ahead,” Sciencedirect Study.

[10] “Development and validation of ML model to estimate patient discharge dates,” Frontiers in Digital Health.

[11] “A systematic literature review of predicting patient discharges using statistical and ML methods,” PMC Review.

[12] “Insights into predictive tool development for emergency care using MIMIC-IV-ED,” Nature Benchmarking Study.

[13] A. Porto, et al., “multi-site emergency department demand forecasting with feature engineering,” Scientific Reports, 2022.

[14] A. Fialho, et al., “Forecasting emergency department demand: A review,” Health Systems, 2021.

[15] W. Wei, et al., “Predicting patient discharge within 24 hours using machine learning,” Journal of Biomedical Informatics, 2019.

[16] D. Wong, et al., “Using explainable machine learning to predict delayed hospital discharges,” BMC Health Services Research, 2021.

[17] P. Sharma, et al., “Clustering approaches to detect hospital overcrowding,” International Journal of Medical Informatics, 2020.

[18] A. Wright, et al., “Design and evaluation of a real-time ED operational dashboard,” Applied Clinical Informatics, 2018.