AMERICAN JOURNAL OF MANAGEMENT AND IOT MEDICAL COMPUTING
Peer Reviewed, Referred & Indexed Journal
E-ISSN: 3069-0110
Vol.5, No.2(2) (2026)
www.ajmimc.com

Intelligent Clinical Response Orchestration via Adaptive AI-Driven Inference for Emergency Service Coordination

S. Venkata Achuta Rao[1], Manamwar Aruna[1], Md. Mushraf[1], Dongari Varun[1]
[1]Department of Computer Science and Engineering, [1]Sree Dattha Institute of Engineering and Science, Nagarjuna Sagar Road, Sheriguda, Ibrahimpatnam, Rangareddy Dist, 501510, Telangana, India.

ABSTRACT

The growing prevalence of chronic conditions such as hypertension and diabetes has intensified the need for intelligent healthcare solutions capable of supporting early-stage detection and informed medical decision-making. Continuous monitoring of multiple patient health indicators is essential, yet traditional diagnostic practices largely depend on manual assessment and basic statistical analysis, limiting their effectiveness in large-scale and data-intensive environments. These conventional systems often fail to capture complex interdependencies within medical data, struggle with imbalanced datasets, and lack the capability to deliver timely predictions in distributed settings. To overcome these challenges, this study presents a real-time clinical decision support framework driven by Artificial Intelligence (AI) and structured on a dual client–server model. The backend server is responsible for data preprocessing, model training, and prediction using Machine Learning (ML) techniques, including Complement Naive Bayes (CNB), Multinomial Naive Bayes (MNB), Perceptron, and a Tao Tree Classifier (TTC). Data preprocessing is performed using Label Encoding and K-Means Synthetic Minority Oversampling Technique (KMeans-SMOTE) to address categorical data handling and class imbalance issues. Communication between system components is facilitated through a Flask-based Application Programming Interface (API) utilizing Hypertext Transfer Protocol (HTTP), ensuring efficient data exchange. The client interface enables users to upload medical datasets, which are remotely processed to generate predictions for blood pressure levels and diabetic conditions. Additionally, Lightning Memory-Mapped Database (LMDB) is integrated to provide secure, high-speed data storage and retrieval. The proposed system enhances predictive accuracy, ensures real-time accessibility, and supports scalable, reliable healthcare delivery.

Keywords: Hypertension Prediction, Diabetes Detection, Machine Learning (ML), Client–Server Architecture, LMDB, Flask Framework

1. INTRODUCTION

The accelerating integration of Artificial Intelligence (AI) into healthcare ecosystems necessitates a comprehensive and multi-dimensional evaluation across diverse care environments, including home-based monitoring, outpatient clinical services, and ambulance-centric Emergency Medical Services (EMS), to rigorously assess its influence on patient outcomes, system efficiency, and care quality [1]. This study emphasizes a holistic and longitudinal data acquisition framework that captures patient-centric information across the entire continuum of care, spanning from remote home monitoring systems to high-dependency and intensive care units. The collected data encompasses a wide spectrum of attributes, including demographic factors (age, gender), socio-economic conditions (living standards, accessibility to healthcare), clinical history (comorbidities, prior treatments), and real-time physiological parameters (heart rate, blood pressure, oxygen saturation), enabling a more context-aware and personalized analytical approach. Despite the growing adoption of AI-powered clinical solutions, a significant gap persists in their validation within real-world, decentralized, and resource-variable environments such as community healthcare systems and emergency response infrastructures, thereby limiting their reliability, scalability, and generalizability [2]. Moreover, the deployment of AI introduces critical concerns related to data privacy breaches, cybersecurity threats, algorithmic bias amplification due to imbalanced or non-representative datasets, lack of explainability in complex predictive models, and ambiguity in legal and ethical accountability for automated decisions [3]. To address these challenges, this work proposes a robust, transparent, and adaptive AI-driven framework that integrates advanced data protection mechanisms, continuous model retraining, bias detection and mitigation strategies, and explainable AI techniques to ensure interpretability and trustworthiness in clinical decision-making. In the domain of EMS, AI-based systems demonstrate substantial operational benefits by leveraging Global Positioning System (GPS)-enabled real-time routing, traffic-aware optimization, and predictive dispatch algorithms to minimize response times and improve emergency handling efficiency [4]. Additionally, these systems facilitate intelligent hospital coordination by dynamically recommending the most suitable healthcare facilities based on patient condition severity, resource availability, and hospital capacity, thereby reducing treatment delays and mitigating overcrowding issues. Furthermore, AI systems continuously adapt to evolving scenarios by providing real-time insights and recommendations to EMS personnel, enhancing situational awareness and reducing cognitive workload. However, effective deployment requires seamless technical integration with existing infrastructures such as communication networks, hospital information systems, and electronic health records, alongside fostering user trust through adequate training and system transparency [5]. Ethical considerations, including fairness, accountability, transparency, and strict adherence to regulatory standards, remain paramount to ensuring the safe, reliable, and scalable implementation of AI-driven solutions in modern healthcare environments.

2. RELATED WORK

Tang, et al. [6] developed a collaborative multi-agent architecture driven by Large Language Models (LLMs) to support intelligent and context-sensitive decision-making in complex medical environments. Their framework, commonly referred to as MedAgents, exploits the reasoning strength of LLMs to perform zero-shot medical understanding, enabling the system to interpret clinical scenarios without requiring extensive task-specific training. The design allows multiple agents to cooperate in solving problems, particularly in time-critical settings such as emergency care and EMS coordination. The study reported improvements in reasoning capability and adaptability when dealing with diverse clinical inputs. Nevertheless, several limitations were identified, including susceptibility to hallucinated outputs, concerns regarding reliability in high-risk medical decisions, lack of transparency in model reasoning, and insufficient regulatory guidance for safe deployment in real-world healthcare systems. Schmidt Batista, et al. [7] examined the influence of Artificial Intelligence (AI)-based medical assistants on enhancing healthcare delivery and optimizing clinical workflows. Their findings indicated that AI-powered assistants significantly reduce the burden of manual documentation by automatically capturing and organizing clinical information. These systems also enhance triage efficiency by prioritizing patients based on urgency, thereby improving patient flow and reducing waiting times. Integration with Electronic Health Records (EHRs) supports seamless data exchange, real-time updates, and automated clinical recommendations. Additionally, predictive analytics capabilities enable early identification of potential health risks. Despite these benefits, the study pointed out issues such as algorithmic bias, data security concerns, ethical challenges, and reluctance among healthcare professionals due to trust and usability barriers. Nickel, et al. [8] focused on optimizing ambulance deployment strategies within EMS by addressing the problem as a stochastic and multi-objective optimization task. Their approach considered uncertainty in emergency demand and incorporated key performance indicators such as response time, coverage, and operational cost. Using sampling-based optimization techniques, the model generated robust solutions for ambulance placement and resource allocation under varying demand conditions. The study highlighted the inherent trade-offs between minimizing costs and maintaining high-quality emergency services. This framework contributes to improved strategic planning and enhances the efficiency and preparedness of emergency response systems.

Miller, et al. [9] investigated techniques for identifying trauma patients in prehospital settings using ambulance care records integrated with hospital data. Their work compared multiple identification methods and demonstrated that linking heterogeneous datasets significantly improves the accuracy of

E-ISSN: 3069-0110
Vol.5, No.2(2) (2026)
www.ajmimc.com

trauma detection. By establishing a continuous data flow from ambulance services to emergency departments and hospitals, the approach enhances situational awareness and supports more informed clinical decisions. The study also emphasized the importance of comprehensive patient tracking across different stages of care and highlighted how integrated data systems can be used to evaluate and improve trauma care performance at a broader healthcare system level. Lujak, et al. [10] proposed a distributed coordination mechanism for Emergency Medical Services (EMS) using a multi-agent Decision Support System (DSS) tailored for critical scenarios such as cardiac emergencies requiring angioplasty. Their framework incorporates a hierarchical optimization strategy that dynamically assigns patients to available ambulances and specialized treatment centers. An auction-based allocation technique is utilized to efficiently distribute limited emergency resources among competing demands, ensuring optimal utilization. The system was evaluated through simulation-based experiments, which demonstrated notable improvements in coordination efficiency, reduced response delays, and better handling of dynamic and uncertain emergency conditions. The study also highlighted the flexibility of the approach in adapting to rapidly changing operational environments.

Elfahim, et al. [11] conducted a comprehensive review of Artificial Intelligence (AI) applications in prehospital emergency care, focusing on how AI techniques can enhance decision-making, prediction, and resource management within EMS systems. Their analysis showed that AI models are capable of providing predictive insights that support proactive emergency response and improve operational efficiency. The study also emphasized the growing importance of AI in handling increasing demand and complexity in emergency healthcare services. However, several limitations were identified, including insufficient availability of high-quality and standardized datasets, lack of explainability in complex AI models, and challenges associated with integrating these technologies into existing EMS infrastructures, which hinder widespread adoption. Zota, et al. [12] explored the application of Artificial Intelligence (AI) in improving healthcare system efficiency through intelligent automation and data management. Their study demonstrated that AI-driven solutions can streamline hospital workflows by automating routine administrative tasks, organizing large volumes of medical data, and enhancing communication across different departments. This leads to faster access to patient information and supports timely clinical decision-making. The research also highlighted the importance of integrating AI systems with existing healthcare infrastructures to ensure seamless data exchange. Despite these advantages, challenges such as interoperability issues, lack of standardized data formats, and system compatibility constraints were identified as key obstacles to effective implementation.

3. PROPOSED SYSTEM

The system architecture is organized into two integrated stages: a server-side training pipeline and a client–server prediction pipeline for real-time inference. In the initial phase, the administrator uploads a structured medical dataset in CSV format, which is processed through a sequence of preprocessing operations including handling missing values using imputation techniques, transforming categorical variables via label encoding, and assigning multi-target outputs such as hypertension and diabetes status. Feature selection is then applied to reduce dimensionality and retain only the most significant predictors contributing to disease classification. To address the issue of class imbalance, the system incorporates a K-Means SMOTE-based oversampling technique, ensuring a balanced distribution of classes before model training. The dataset is subsequently divided into training and testing subsets using an 80:20 split strategy to maintain evaluation integrity. During the model training stage, multiple machine learning algorithms are implemented, including CNB, MNB, and Perceptron, which serve as baseline models. Alongside these, the proposed optimized TTC model is developed to enhance predictive performance and robustness.

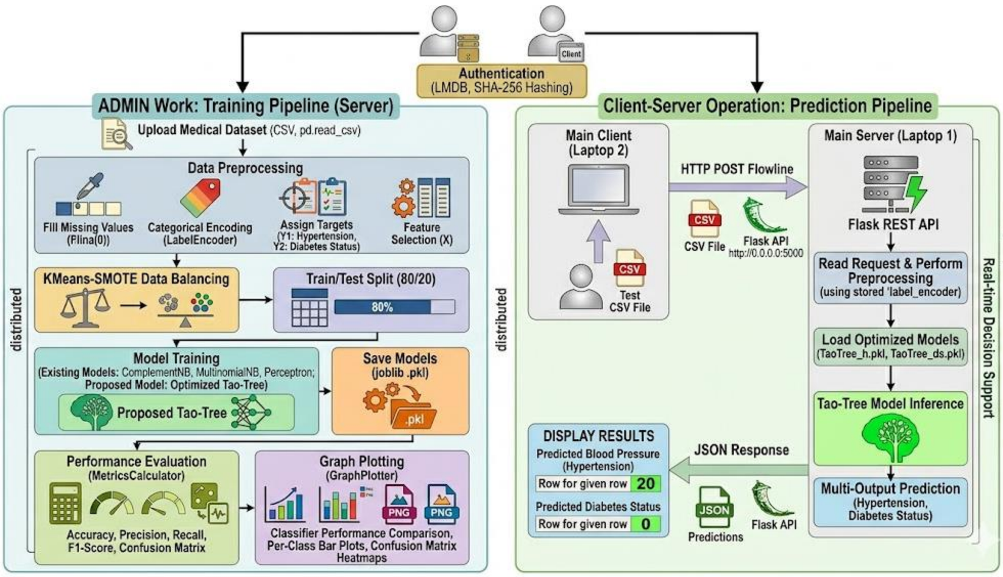

Fig. 1: Proposed system architecture

All trained models are serialized and stored in formats such as .pkl and .joblib, enabling efficient reuse during deployment. The models are evaluated using comprehensive performance metrics, including accuracy, precision, recall, F1-score, and confusion matrix analysis, supported by graphical visualizations like classifier comparison charts, per-class performance plots, and heatmaps. In the deployment phase, the system transitions to a client–server architecture where a client uploads test data via an HTTP POST request to a Flask-based REST API. Authentication is enforced using LMDB-based storage combined with SHA-256 hashing to ensure secure access. The server receives the request, applies consistent preprocessing using previously stored encoders, and loads the optimized Tao-Tree model for inference. As illustrated in Fig. 1, although multiple models such as CNB, MNB, and Perceptron are trained and evaluated during the development phase, only the optimized TTC model is selected for production due to its superior performance. The model performs multi-output prediction, simultaneously estimating hypertension and diabetes conditions, and the results are returned to the client in JSON format. This architecture enables scalable, secure, and real-time decision support suitable for distributed healthcare and emergency medical service environments.

3.1 TTC

The TTC model constructs a hierarchical tree structure where each internal node represents a decision based on a feature and each leaf node corresponds to a class label. At each split, the algorithm selects the feature that maximizes information gain or minimizes impurity, ensuring optimal separation of classes. Once the tree is built, pruning techniques are applied to remove redundant or weak branches, simplifying the final structure and preventing overfitting as shown in Fig. 2. The TTC further incorporates probabilistic refinement through a built-in optimization layer that adjusts predictions slightly to enhance stability and accuracy. This makes it particularly effective in providing transparent, interpretable, and reliable predictions for medical decision-making.

Step 1: Input Feature Representation The process begins by representing each instance as a numerical feature vector containing relevant attributes. These features serve as the basis for learning decision rules. Proper representation ensures that meaningful patterns can be extracted during tree construction.

Step 2: Root Node Initialization All training samples are initially assigned to a single root node. This node represents the complete dataset before any partitioning is performed. It serves as the starting point for recursive tree growth.

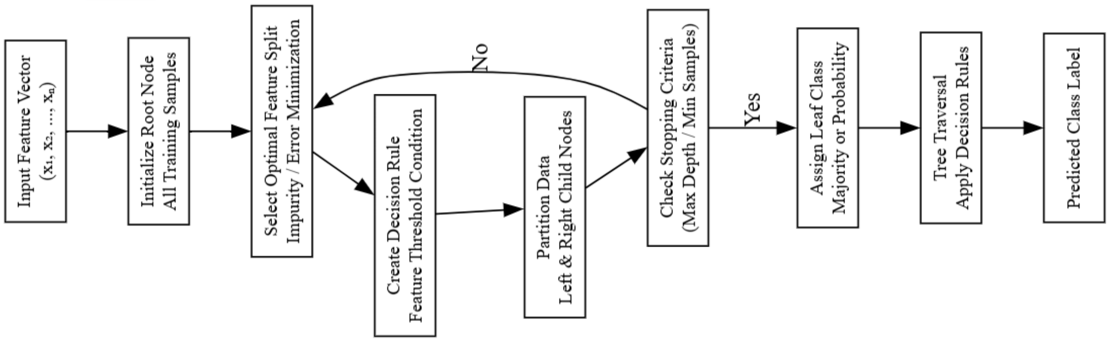

Fig. 2: Internal working of TTC.

Step 3: Optimal Feature Selection At each stage, the algorithm evaluates all candidate features to identify the best splitting attribute. The selection is based on minimizing classification error or impurity. This step ensures that each split improves class separation.

Step 4: Decision Rule Construction A decision rule is created using the selected feature and an appropriate threshold. This rule defines how data is divided into different branches. It forms the core decision logic of the tree.

Step 5: Data Partitioning The dataset is divided into child nodes based on the decision rule. Each child node contains a subset of samples sharing similar feature characteristics. This hierarchical partitioning allows the model to capture complex patterns.

Step 6: Stopping Criteria Evaluation, the algorithm checks predefined stopping conditions such as maximum tree depth or minimum number of samples. If the conditions are not met, further splitting is performed. This prevents excessive tree growth and overfitting.

Step 7: Leaf Node Assignment When stopping criteria are satisfied, the node is converted into a leaf node. A class label is assigned based on majority voting or probability estimation. This label represents the final decision for that branch.

Step 8: Tree Traversal for Prediction For prediction, new instances traverse the tree by following the learned decision rules from the root to a leaf. Each decision guides the instance to the appropriate branch. This traversal process is efficient and interpretable.

Step 9: Final Output Generation The class label assigned at the leaf node is returned as the final prediction. This output represents the model's decision based on hierarchical rule evaluation. The result supports accurate and explainable decision-making.

Advantages

- Able to capture complex, non-linear relationships between features, making it suitable for real-world medical data.
- Provides interpretable decision rules, allowing predictions to be easily understood and explained.
- Handles both numerical and categorical attributes effectively without extensive feature transformation.
- Offers stable performance on balanced datasets by learning hierarchical decision structures.
- Supports multi-class classification naturally through tree-based decision paths.
- Requires minimal parameter tuning compared to many advanced machine learning models.

4. RESULTS DESCRIPTION

Fig. 3 shows the Tao-Tree classifier for binary diabetes prediction. Rows represent True Class (bottom to top: Y, N), and columns represent Predicted Class (left to right: Y, N). The diagonal shows correct predictions: True Y → Predicted Y: 951 (bright yellow), True N → Predicted N: 867 (lime green). Off-diagonal errors are minimal: only 8 cases of Y misclassified as N, and 10 cases of

N misclassified as Y. The color scale ranges from dark purple (~200) to bright yellow (~800), highlighting the model's outstanding accuracy with just 18 total misclassifications.

Fig. 4 illustrates the prediction process carried out on Laptop 2 (client) within the client–server architecture, where the user uploads a test dataset in comma-separated values (CSV) format for analysis. The figure depicts how the client system transmits the input data to the server using HTTP and receives the predicted results in real time through the API. It highlights the processing of multiple input instances, where each row of patient data is evaluated to generate predictions for blood pressure category and diabetes status. The output demonstrates the effectiveness of the deployed ML models in handling multi-attribute medical data and returning structured prediction results.

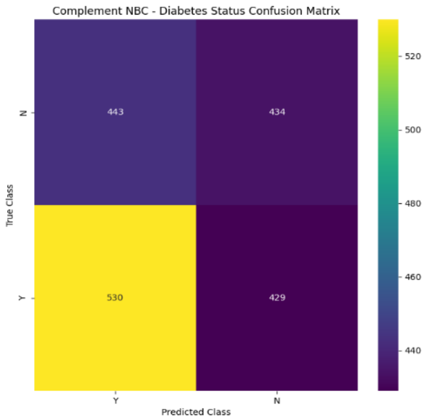

Fig. 3: Confusion matrices obtained with hypertension data using Proposed Tao-tree model.

```
ending file: F:/sak informatics/documents/KPRIT/CODES/CSE-C/CS Real Time AI Powered Decision Support System/Laptop 2/Test_data.csv to server...
redictions received:

nput Data Sent to Client
Row 1: {'Sex': 'F', 'Age': 32.0, 'Weight (kg)': 69.1, 'Height (m)': 1.71, 'BMI': 23.6, 'Abdominal Circumference (cm)': 86.2, 'Blood Pressure (mmHg)': '125/79', 'Total Cholesterol (mg/dL)': 248.0, 'HDL (mg/d
L)': 78.0, 'Fasting Blood Sugar (mg/dL)': 111.0, 'Smoking Status': 'N', 'Physical Activity Level': 'Low', 'Family History of CVD': 'N', 'CVD Risk Level': 'INTERMEDIARY', 'Height (cm)': 171.0, 'Waist-to-
Height Ratio': 0.504, 'Systolic BP': 125.0, 'Diastolic BP': 79.0, 'Estimated LDL (mg/dL)': 140.0, 'CVD Risk Score': 17.93}
redicted Blood Pressure: (Elevated)
redicted Diabetes Status: Unknown Class (Y)

nput Data Sent to Client
Row 2: {'Sex': 'F', 'Age': 55.0, 'Weight (kg)': 118.7, 'Height (m)': 1.69, 'BMI': 41.6, 'Abdominal Circumference (cm)': 82.5, 'Blood Pressure (mmHg)': '139/70', 'Total Cholesterol (mg/dL)': 162.0, 'HDL (mg/
dL)': 50.0, 'Fasting Blood Sugar (mg/dL)': 135.0, 'Smoking Status': 'Y', 'Physical Activity Level': 'High', 'Family History of CVD': 'Y', 'CVD Risk Level': 'HIGH', 'Height (cm)': 169.0, 'Waist-to-Height Rati
o': 0.488, 'Systolic BP': 139.0, 'Diastolic BP': 70.0, 'Estimated LDL (mg/dL)': 82.0, 'CVD Risk Score': 20.51}
redicted Blood Pressure: (Hypertension Stage 1)
redicted Diabetes Status: Unknown Class (Y)

nput Data Sent to Client
Row 3: {'Sex': 'M', 'Age': nan, 'Weight (kg)': nan, 'Height (m)': 1.83, 'BMI': 26.9, 'Abdominal Circumference (cm)': 106.7, 'Blood Pressure (mmHg)': '104/77', 'Total Cholesterol (mg/dL)': 103.0, 'HDL (mg/
dL)': 73.0, 'Fasting Blood Sugar (mg/dL)': 114.0, 'Smoking Status': 'N', 'Physical Activity Level': 'High', 'Family History of CVD': 'Y', 'CVD Risk Level': 'INTERMEDIARY', 'Height (cm)': 183.0, 'Waist-to
Height Ratio': 0.583, 'Systolic BP': 104.0, 'Diastolic BP': 77.0, 'Estimated LDL (mg/dL)': 0.0, 'CVD Risk Score': 12.64}
redicted Blood Pressure: (Normal)
```

Fig. 4: Predicting Test Data in Laptop 2 (Client)

The comparative analysis as shown in table 1 & 2, evaluates the performance of various classification algorithms applied for hypertension prediction. The CNB Classifier achieved a modest accuracy of 39.07%, showing limited capability in learning complex feature interactions. Similarly, the MNB model performed slightly better with an accuracy of 46.63%, indicating marginal improvement in classification precision and recall. The Perceptron algorithm, while effective in some linear classification tasks, recorded a lower performance with 38.93% accuracy, reflecting challenges in adapting to nonlinear patterns within the dataset. In contrast, the proposed TTC demonstrated exceptional performance, achieving an impressive 98.38% accuracy, along with consistently high precision, recall, and F1-score values.

Table. 1: comparative analysis for emergency medical services.

Algorithm	Accuracy	Precision	Recall	F1-Score
Complement NBC - Hypertension	39.072	40.905	38.618	28.971
Multinomial NBC - Hypertension	46.638	49.577	46.352	44.165
Perceptron - Hypertension	38.937	37.860	38.788	35.175

TTC - Hypertension	98.386	98.378	98.386	98.377

Table 2: Classifier Performance Comparison for Diabetes Status

Algorithm	Accuracy	Precision	Recall	F1-Score
Complement NBC – Diabetes Status	52.5	52.4	52.4	52.4
Multinomial NBC – Diabetes Status	52.6	52.4	52.4	52.4
Perceptron – Diabetes Status	48.0	51.5	50.1	34.3
TTC – Diabetes Status	99.0	99.0	99.0	99.0

5. CONCLUSION

The study effectively validates the capability of an intelligent, data-centric framework designed to assist clinical decision-making in emergency situations. By incorporating advanced data preprocessing strategies, techniques for addressing class imbalance, and multiple machine learning classifiers, the system accurately evaluates patient health information and produces dependable predictions for critical conditions such as hypertension levels and diabetic status. A detailed comparative analysis of the implemented algorithms reveals their individual advantages and limitations, offering clear insights into their applicability within healthcare contexts. The integration of a tree-based learning model contributes to improved consistency and interpretability of predictions, ensuring that results are more understandable and practical for real-world use. Furthermore, the system supports real-time prediction through an interactive user interface and a web-enabled service, highlighting its effectiveness in time-critical medical scenarios. The findings demonstrate that the integration of artificial intelligence significantly enhances diagnostic assistance, minimizes response delays, and supports more accurate and informed clinical decision-making.

REFERENCES

[1] O'Cathain, A., Webber, E., Nicholl, J., Munro, J., & Knowles, E. NHS Direct: consistency of triage outcomes. Emergency Medicine Journal, 20, 289-292.
[2] Kuttan, N., Pundkar, A., Gadkari, C., Patel, A., & Kumar, A. (2025). Transforming emergency medicine with artificial intelligence: From triage to clinical decision support. Multidisciplinary Reviews.
[3] Masoud, M.P., Nejad, M.K., Darebaghi, H., Chavoshi, M., & Farahani, M. (2018). The Decision Support System and Conventional Method of Telephone Triage by Nurses in Emergency Medical Services: A Comparative Investigation. Int. J. E Bus. Res., 14, 77-88.
[4] Bashiri, A., Alizadeh Savareh, B., & Ghazisaeedi, M. (2019). Promotion of prehospital emergency care through clinical decision support systems: opportunities and challenges. Clinical and Experimental Emergency Medicine, 6, 288-296.
[5] Kanani, S., Mahdavi, I., Ziaie, N., & Rahimpour Cami, B. (2024). Optimization of referral system for providing medical services to cardiac patients with cardiogenic shock manifestation under uncertainty. Scientia Iranica.
[6] Tang, X., Zou, A., Zhang, Z., Zhao, Y., Zhang, X., Cohan, A., & Gerstein, M. B. (2023). MedAgents: Large Language Models as Collaborators for Zero-shot Medical Reasoning. arXiv:2311.10537
[7] Schmidt Batista, A. (2025). The Transforming Clinical Practice: The Role of AI-Powered Medical Assistants in Enhancing Healthcare Efficiency and Decision-Making. Journal of Next-Generation Research 5.0, 1(4). https://doi.org/10.70792/jngr5.0.v1i4.121
[8] Nickel, S., Reuter-Oppermann, M., & Saldanha-Da-Gama, F. (2016). Ambulance location under stochastic demand: A sampling approach. Operations Research for Health Care, 8, 24–32
[9] Miller, M., Jorm, L., Partyka, C., Burns, B., Habig, K., Oh, C., Immens, S., Ballard, N., & Gallego, B. (2024). Identifying prehospital trauma patients from ambulance patient care records: Comparing two methods using linked data in New South Wales, Australia. Injury, 55, 111570

E-ISSN: 3069-0110
Vol.5, No.2(2) (2026)
www.ajmimc.com

[10] Lujak, M., Billhardt, H., & Ossowski, S. (2016). Distributed coordination of emergency medical service for angioplasty patients. Annals of Mathematics and Artificial Intelligence, 78, 73–100. https://doi.org/10.1007/s10472-016-9507-9

[11] Elfahim, O., Edjinedja, K. L., Cossus, J., Youssfi, M., Barakat, O., & Desmettre, T. (2025). A Systematic Literature Review of Artificial Intelligence in Prehospital Emergency Care. Big Data and Cognitive Computing, 9, 219. https://doi.org/10.3390/bdcc9090219

[12] Zota, R. D., Cîmpeanu, I. A., & Lungu, M. A. (2025). Exploring AI in Healthcare Systems: A Study of Medical Applications and a Proposal for a Smart Clinical Assistant. Electronics, 14(18), 3727. https://doi.org/10.3390/electronics14183727

Received: 08-05-2026 | Accepted: 16-06-2026 | Published: 22-06-2026 | www.ajmimc.com