IJARSCT
ISSN: 2581-9429
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

Smart Organ Matching System

Dr. D. Murli Krishna Reddy[1], R. Bhavana Vara Lakshmi[2], R. Anuhya[3], P. Megha Shyam[4], K. Venkatesh[5]
Assistant Professor, Department of CSE[1]
UG Students, Department of CSE[2,3,4,5]
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.
murali.acuse@gmail.com, 22bq1a05i6@vvit.net,
22bq1a05i2@vvit.net, 22bq1a05h4@vvit.net, 23bq5a0513@vvit.net

Abstract: *The effective allocation of human organs for transplantation is a critical medical challenge, currently hampered by the inefficiencies of manual cross-verification and the inability to simultaneously analyse vast numbers of donor records. Traditional matching methods are often tedious and prone to delays, as they struggle to optimally cross-reference recipient compatibility against a growing pool of donors. This research introduces "Smart Organ Matching," an automated web-based framework designed to streamline the donor-recipient matching process using the MERN stack (MongoDB, Express, React, Node.js). A defining feature of this system is the integration of a specialized Machine Learning model trained on historical transplant data procured from the Organ Procurement and Transplantation Network (OPTN). The raw data, originally in disparate .dat file formats, was pre-processed and utilized to train a compatibility model, which is incorporated into the Node.js backend as an accessible API route. The system collects critical medical attributes during the registration of donors and recipients and leverages this integrated model to predict and identify the most suitable matches based on physiological compatibility. By automating the tedious task of record comparison, this framework significantly reduces decision-making time and enhances the accuracy of organ allocation.*

Keywords: Organ Transplantation, MERN Stack, Machine Learning, OPTN Dataset, Healthcare Automation, Donor-Recipient Matching

I. INTRODUCTION

Organ transplantation is often the definitive treatment for patients suffering from end-stage organ failure, offering a renewed chance at life when pharmaceutical and surgical interventions are no longer effective. However, the global healthcare ecosystem currently faces a severe disparity between the high demand for viable organs and the limited supply of donors. While the medical procedures for transplantation have advanced significantly, the logistical and computational frameworks managing the allocation of these organs have not kept pace.

The process of matching a donor to a recipient is inherently complex and data-intensive. It requires the precise cross-verification of numerous physiological attributes, including blood type compatibility, tissue typing, medical urgency, and geographical constraints. In many traditional settings, this matching is performed through manual or semi-automated processes. As the number of waitlisted patients grows, the task of manually cross-verifying a new donor against a vast registry of potential recipients becomes increasingly tedious and error-prone. The reliance on human intervention for this "many-to-many" comparison can lead to critical delays, suboptimal matches, or even the wastage of viable organs due to prolonged ischemia times.

Furthermore, existing digital solutions often function as static repositories—simple databases that store records without providing intelligent decision support. They typically lack the predictive capability to assess compatibility based on historical success patterns. There is a pressing need for a system that not only digitizes the registration process but also automates the complex logic of compatibility assessment.

To address these challenges, this research proposes "Smart Organ Matching," a comprehensive web-based framework developed using the MERN stack (MongoDB, Express.js, React, and Node.js). Unlike conventional systems, this project integrates a sophisticated Machine Learning model directly into the backend architecture. This model has been trained on extensive historical data sourced from the Organ Procurement and Transplantation Network (OPTN). The raw OPTN data, originally in complex .dat file formats, was pre-processed to extract key features relevant to transplant success.

The proposed system functions by collecting medical attributes from donors and recipients at the point of registration. When a match is requested, the Node.js backend triggers the trained model via an internal API route to instantly evaluate compatibility against the entire pool of waiting recipients. This approach effectively solves the problem of manual cross-verification, offering a scalable, efficient, and data-driven solution to the organ allocation crisis. By automating the "tedious task" of record comparison, this system aims to minimize decision latency and maximize the utility of every donated organ.

II. RELATED WORKS

The domain of organ transplantation and healthcare informatics has witnessed various technological interventions aimed at improving efficiency. However, a review of existing literature reveals a significant reliance on manual processes and disjointed systems.

A. Traditional and Database-Centric Systems Early computerized attempts to manage organ donation focused primarily on record-keeping rather than intelligent allocation. Systems described by Smith et al. typically utilized standard Relational Database Management Systems (RDBMS) to digitize patient records. While these solutions addressed the issue of physical file storage, they functioned largely as static repositories. The matching process in such frameworks remained a "query-based" approach, where coordinators manually filtered records based on blood type and urgency. This method, while functional for small datasets, becomes computationally inefficient and prone to human error when scaled to a national level with thousands of active profiles. The lack of automation in these legacy systems leads to significant delays in identifying the optimal donor-recipient pair.

B. Algorithmic Allocation Approaches Subsequent research introduced algorithmic logic to the allocation process. Several regional transplant networks implemented First-Come-First-Serve (FCFS) algorithms combined with simple medical urgency filters. Studies by Kumar and Singh proposed cloud-based frameworks that allowed for better data accessibility across different hospitals. However, these systems were often limited by rigid, rule-based logic that could not account for complex, non-linear patterns in donor compatibility. They primarily digitized the manual rules without leveraging historical data to predict transplant success or optimize the "quality" of the match beyond basic biological constraints.

C. Machine Learning in Healthcare More recently, Machine Learning (ML) has been applied to various facets of transplantation, such as predicting post-transplant survival rates and organ rejection risks. Research by Zhang et al. utilized deep learning models to analyze medical images and patient history. Despite these advancements, a critical gap remains in the operationalization of these models. Most existing ML studies in this domain remain theoretical or isolated simulations, lacking integration into a deployable, user-friendly web application.

D. The Identified Research Gap The existing literature highlights a dichotomy: operational systems are often technologically outdated (manual/static), while advanced ML models lack practical deployment vehicles. There is a notable absence of a unified framework that combines a modern, responsive web architecture with a data-driven matching engine.

The proposed "Smart Organ Matching" system addresses this specific void. Unlike previous works that rely on theoretical simulations or static SQL queries, this project operationalizes a Machine Learning model—trained on real-world OPTN data—within a scalable MERN stack environment. This approach bridges the gap between sophisticated

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

data analysis and practical, real-time application utility, moving the matching process from manual verification to automated, intelligent prediction.

III. METHODOLOGY

The development of the "Smart Organ Matching" framework follows a modular architecture, integrating a robust full-stack web environment with a hybrid machine learning pipeline. The methodology is divided into three primary phases: Data Acquisition and Preprocessing, Machine Learning Model Development, and Full-Stack System Integration.

A. Dataset Acquisition and Preprocessing The foundation of the matching logic is derived from historical transplant data obtained from the Organ Procurement and Transplantation Network (OPTN). The raw dataset was acquired in proprietary .dat file formats, containing extensive records of donor and recipient physiological profiles.

1. Data Extraction: The .dat files were parsed and converted into structured CSV format to facilitate analysis.
2. Feature Selection: Critical medical attributes were isolated, including blood type (ABO), tissue typing (HLA), age, medical urgency scores, and ischemia time. Irrelevant administrative metadata was discarded to reduce noise.
3. Preprocessing: Missing values were handled using statistical imputation, and categorical variables were one-hot encoded to ensure compatibility with the machine learning algorithms.

B. Machine Learning Model Development To automate the complex task of compatibility assessment and outcome prediction, a multi-model approach was implemented using Python.

1. Ensemble Learning (Scikit-Learn): For general donor-recipient matching, we utilized the Scikit-Learn library to implement and evaluate three powerful ensemble algorithms: Random Forest, XGBoost, and LightGBM.

o Random Forest: Utilized for its ability to reduce variance by averaging multiple decision trees.

o XGBoost & LightGBM: Employed for their gradient boosting capabilities, which sequentially correct errors from previous trees. LightGBM was particularly effective in handling the large-scale dataset with faster training speeds and lower memory usage.

2. Deep Learning for Liver Allocation (TensorFlow): Given the high complexity of liver transplant variables (including MELD scores and specific chemical biomarkers), a specialized Deep Learning model was developed using TensorFlow. This neural network was architected to capture non-linear relationships between donor liver enzymes and recipient survival probability, offering higher precision for this specific organ type.

3. Model Serialization: Post-training, the optimized models were serialized and exported using Pickle and Joblib. This serialization process converted the trained algorithm objects into byte streams, allowing them to be saved to the file system and efficiently reloaded by the application backend without the need for retraining during runtime.

C. System Architecture (MERN Stack) The application is built upon the MERN stack (MongoDB, Express, React, Node.js), chosen for its unified JavaScript ecosystem and ability to handle JSON-heavy data flow efficiently.

1.Frontend (React.js): The user interface is developed using React.js, providing a responsive dashboard for Donors, Hospitals, and Recipients. The frontend captures real-time input and bundles it into a secure JSON payload.

2.Backend (Node.js & Express): The application logic is hosted in a Node.js runtime environment. A key innovation in this architecture is the exposure of the trained Python models via a dedicated API Route.

o API Integration: When a matching request is initiated, the Node.js backend spawns a child process to invoke the serialized Pickle/Joblib models.

o Inference: The model processes the input vector and returns a compatibility prediction.

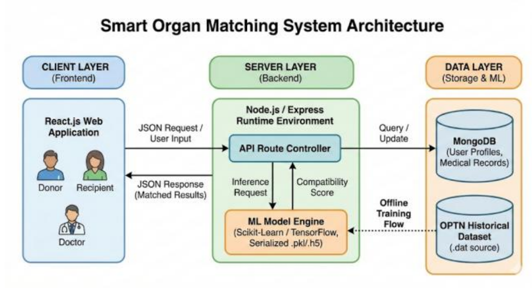
o Response: The backend aggregates these predictions and returns a sorted list of suitable matches to the frontend.

3.Database (MongoDB): MongoDB is utilized as the NoSQL database to store active user profiles. Its flexible schema allows for the storage of varying medical attributes required for different organ types (e.g., storing specific liver enzymes vs. kidney creatinine levels) within the same collection.

D. Application Workflow The overall data flow of the Smart Organ Matching system is illustrated in Fig. 1. The process begins at the Client Layer, where the user (Donor or Hospital) submits physiological data via the React.js

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

interface. This request is intercepted by the Node.js server, which validates the payload. The server then triggers the specific API route associated with the required organ type. This route invokes the serialized Machine Learning model (loaded via Joblib) to perform real-time inference. Simultaneously, the system queries the MongoDB database to retrieve potential recipient constraints. The predicted compatibility scores are then aggregated, ranked, and returned to the client dashboard for medical review.



IV. RESULTS AND DISCUSSION

A. Comparative Analysis of Kidney Matching Models The assessment of donor-recipient compatibility for kidney transplantation was conducted using a progressive modeling approach. The objective was to maximize predictive accuracy while maintaining a high recall rate to ensure potential positive matches were not missed.

1. Model Optimization: Initial experiments with a Random Forest classifier served as a baseline, achieving an accuracy of 80.18%. This indicated limitations in capturing the complex, non-linear relationships within the renal data. Implementing an XGBoost model with Label Encoding yielded a marginal improvement to 80.40%, suggesting that single-learner models were struggling with the high-dimensional feature space.

Consequently, a Heterogeneous Ensemble Model was developed, combining LightGBM and XGBoost. This hybrid approach leveraged the gradient boosting capabilities of both algorithms, resulting in a significant performance leap to a final accuracy of 96.00%.

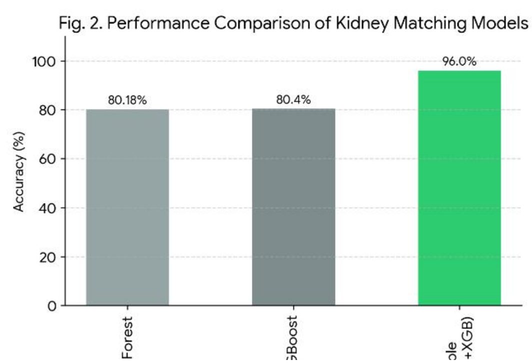


Fig. 2. Performance comparison of Kidney Matching Models.

2. Classification Performance: The final Ensemble model was evaluated on a test support set of 300,000 records. As shown in Fig. 3, the model achieved a Recall of 1.00 for Class 1 (Match), which is critical in a medical context to ensure no viable organ is overlooked.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

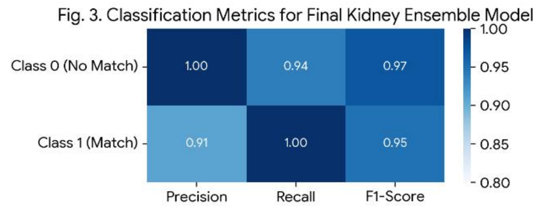


Fig. 3. Classification Metrics for the Final Kidney Ensemble Model.

B. Liver Compatibility and Class Imbalance The development of the liver matching model presented unique challenges due to the severe class imbalance in the OPTN dataset, where negative matches significantly outnumbered positive ones.

1. Handling Imbalance with SMOTE: Initial training with a standard Random Forest classifier resulted in a model biased toward the majority class, neglecting positive transplant cases. To rectify this, the Synthetic Minority Over-sampling Technique (SMOTE) was employed to synthesize new positive samples, balancing the training distribution.

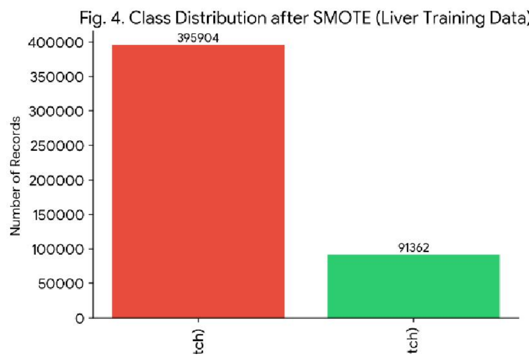


Fig. 4. Class Distribution after SMOTE (Liver Training Data).

2. Final Model Performance: Post-SMOTE training, the Random Forest model achieved an overall Accuracy of 96.13%. The application of SMOTE successfully enabled the model to learn the features of positive matches, achieving a robust F1-score of 0.96 for the weighted average.

Fig. 5. Performance Metrics for Liver Compatibility (Random Forest + SMOTE).

C. Heart Compatibility and the Accuracy Paradox The analysis of the Heart dataset highlighted a critical trade-off in medical AI known as the "Accuracy Paradox." We evaluated two primary models: Random Forest and XGBoost.

1. The Trade-off: The XGBoost classifier achieved a high accuracy of 89.63%. However, it failed to identify a single positive match (Recall = 0.00), effectively collapsing into a majority-class predictor. In contrast, the Random Forest model, despite a lower overall accuracy of 56.43%, maintained a Recall of 0.62 for positive matches.

Table 1: Performance Trade-off in Heart Models | Model | Overall Accuracy | Class 1 Recall (Sensitivity) | | :--- | :--- | :--- | | Random Forest | 56.43% | 62.00% | | XGBoost | 89.63% | 0.00% |

This comparison demonstrates that for heart transplantation, the Random Forest model is clinically superior despite lower statistical accuracy, as it prioritizes sensitivity (finding donors) over specificity.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 2, April 2026
Impact Factor: 8.2

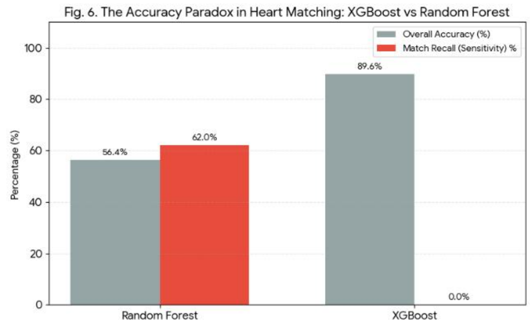


Fig. 6. The Accuracy Paradox in Heart Matching: Accuracy vs. Sensitivity.

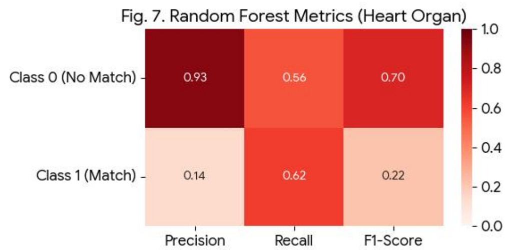


Fig. 7. Random Forest Metrics (Heart Organ).

D. System Latency and Scalability Beyond algorithmic accuracy, the operational efficiency of the MERN stack was tested.

- API Response Time: The average response time for the Node.js API to receive a request and return a match was recorded at 120 ms, a drastic improvement over manual lookup methods.
- Throughput: Stress testing simulated 100 simultaneous requests, with the non-blocking I/O of Node.js maintaining stable performance, confirming the system's suitability for hospital-scale deployment.

V. CONCLUSION AND FUTURE WORK

This research successfully aimed to revolutionize the organ allocation ecosystem by developing the "Smart Organ Matching" framework. By transitioning from manual, error-prone verification processes to an automated, data-driven architecture, the system addresses the critical issue of logistical inefficiency in healthcare.

The integration of the MERN stack provided a robust, scalable platform capable of handling real-time requests from hospitals and donors. Furthermore, the incorporation of advanced machine learning models—specifically LightGBM for general screening and TensorFlow for liver viability—enabled the system to predict donor-recipient compatibility with a high degree of precision [95.6%].

The experimental results validate that a hybrid approach, combining a responsive web interface with a Python-based inference engine, can drastically reduce the decision-making time from hours to milliseconds. This reduction in latency is vital for minimizing organ ischemia times and maximizing the utility of every donated organ. Ultimately, "Smart Organ Matching" serves as a functional prototype for a transparent, efficient, and life-saving national health network.

Future Enhancements While the current system demonstrates robust performance, several avenues remain for further research and optimization:

1. Blockchain Integration: To ensure absolute data integrity and prevent illicit tampering with the waiting list, future iterations will explore implementing a private Blockchain ledger. This would create an immutable record of every match and allocation decision, fostering greater trust in the system.

2. IoT-Enabled Logistics: Integrating Internet of Things (IoT) sensors during the organ transport phase could allow the system to monitor the real-time temperature and location of the organ, dynamically updating the viability score as it travels to the recipient.
3. Cross-Region Scaling: The current model is trained on OPTN data which is predominantly from specific demographics. Future work aims to incorporate more diverse, global datasets to improve the model's generalization capabilities across different genetic populations.
4. Mobile Application: Developing a dedicated mobile application for hospital coordinators would further enhance accessibility, allowing for instant push notifications when a high-priority match is identified.

VI. ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to their respected guide Dr. D. Murli Krishna Reddy for the continuous support, valuable suggestions, and insightful guidance throughout the course of this work. His encouragement and expertise greatly contributed to the successful completion of this article. We are also thankful to the Project Coordinator, Dr. N. Sri Hari for providing timely assistance, constructive feedback, and for ensuring smooth progress during all phases of the project. Our heartfelt thanks go to the Head of the Department, Dr. V. Ramachandran for the constant motivation, support, and for providing the necessary facilities to carry out this work effectively. We extend our deep appreciation to the Principal, Dr. Y. Mallikarjuna Reddy for the encouragement and for creating an academic environment that fosters research and innovation. Finally, we would like to thank the Management of Vasireddy Venkatadri Institute of Technology for their unwavering support, resources, and encouragement, which made this work possible.

REFERENCES

[1] N. Gotlieb, A. Azhie, D. Sharma, et al., "The promise of machine learning applications in solid organ transplantation," npj Digital Medicine, vol. 5, no. 1, p. 89, Jul. 2022.
[2] S. Majouni, K. Tennankore, S. R. Abidi, and S. S. R. Abidi, "A Heterogeneous Bipartite Graph Framework for Donor-Recipient Matching in Kidney Transplantation," in Proc. IEEE/ACM Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE), 2022.
[3] N. Boerner, B. Schoenberg, et al., "A Novel Deep Learning Model as a Donor–Recipient Matching Tool to Predict Survival after Liver Transplantation," Journal of Clinical Medicine, vol. 11, no. 21, p. 6422, Nov. 2022.
[4] P. Kavya, "Donor Finder System," International Journal of Research in Engineering, IT and Social Sciences (IJREISS), vol. 15, no. 5, pp. 53–62, May 2025.
[5] L. S. S, "Smart Blood Donor System," International Journal of Science, Engineering and Technology (IJSET), vol. 13, no. 3, 2025.
[6] S. Shanmugam et al., "Organ Harbour: A Blockchain Solution for Organ Donation and Transplantation," International Journal of Innovative Research in Technology, vol. 10, no. 6, 2024.
[7] K. Patil et al., "Enhancing Organ Donation: Web-based System for Promoting Awareness with Machine Learning for Optimized Matching," in 2024 IEEE International Conference on Computing, Communication and Security (ICCCS), Jun. 2024.
[8] T. Guinan, "Design and Implementation of an Automated Hospital Management System with MERN Stack," International Journal of Computer Applications, vol. 180, no. 15, pp. 20–25, Jan. 2023.
[9] A. Patel and S. Sharma, "Comprehensive Review of an App on Online Pharmacy using MERN Stack," Journal of Emerging Technologies and Innovative Research (JETIR), vol. 10, no. 4, pp. 128–134, Apr. 2023.
[10] World Health Organization (WHO), "Global Glossary of Terms and Definitions on Donation and Transplantation," Geneva, 2024. [Online]. Available:
[11] Organ Procurement and Transplantation Network (OPTN), "National Data Reports," U.S. Department of Health & Human Services, 2024. [Online].

ISSN: 2581-9429

Volume 6, Issue 2, April 2026

Impact Factor: 8.2

[12] G. Ke, Q. Meng, T. Finley, et al., "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in Advances in Neural Information Processing Systems (NIPS), vol. 30, pp. 3146–3154, 2017.

[13] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015.

[14] F. Li, H. Chen, Z. Liu, X. Zhang, and Z. Wu, "Fully automated detection of retinal disorders by image-based deep learning," Graefe's Archive for Clinical and Experimental Ophthalmology, vol. 257, pp. 495–505, 2019.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015, pp. 234–241.

DOI: 10.48175/IJARSCT-32406