Fetal Anomaly Detection System
"Healthier Moms and Babies"

Jay Bambhroliya, Shivani Kotian, Jay Patel
Safal Rijal, Pooja Shekhar, Yash Sohagia
Group-3
Advanced Software Engineering

Abstract

The Fetal Anomaly Detection System (FADS) revolutionizes prenatal care by using machine learning to analyze cardiotocography(CTG) datasets. The system classifies fetal health into normal, suspect, and pathological categories, providing healthcare professionals with actionable insights for informed and correct decision-making. FADS comprises a robust architecture, including data ingestion, processing, a machine learning model, user interfaces, and a notification system. The high-level architecture of FADS involves data flow from CSV dataset uploads through data processing to result in display by a secure API. The data design outlines the key characteristics of the cardiotocography dataset, and resource estimates detail the hardware, software, and human resources required for development. FADS is designed to work smoothly with other systems, providing user-friendly design, clear explanations for machine learning decisions, and ensuring that it meets the highest healthcare standards. The estimated project timeline spans 3-4 months for development, with additional time was dedicated for model training, testing, and documentation.

1. Introduction

In the realm of obstetrics, the accurate assessment of fetal health is paramount. Recent advancements in machine learning (ML) have opened new avenues for enhancing the precision and efficacy of prenatal care. This research paper focuses on the development and implementation of a Fetal Anomaly Detection System (FADS), a groundbreaking application designed to leverage the capabilities of ML in analyzing cardiotocography (CTG) datasets[2]. The primary objective of this system is to categorize fetal health into three distinct classifications: normal, suspect, and pathological. By doing so, FADS aims to provide healthcare professionals with a robust tool for making informed decisions, thereby improving the outcomes of obstetric care.[8]

The urgency and importance of such a system are underscored by the challenges faced in current obstetric practices, where the interpretation of CTG data largely depends on the clinician's expertise and experience. This subjective element can lead to inconsistencies in fetal health assessment. FADS addresses this gap by introducing an objective and standardized method of analysis, reducing the potential for human error and variability in interpretations.[4]

The methodology underpinning FADS involves a sophisticated ML model trained on extensive CTG data.[9] This model is integrated into a user-friendly platform with dual access points: one for patients to input data, and another for healthcare providers to review the predictions. The system functions by processing the input data through the trained ML model, which then classifies the fetal health status. Both patients and doctors have the capability to view these predictions, facilitating a transparent and collaborative approach to prenatal care management.

The innovation of FADS lies not only in its technical prowess but also in its potential to revolutionize prenatal care practices. By harnessing the power of ML, FADS offers a more accurate, efficient, and accessible means of fetal health monitoring, setting a new standard in obstetric care.[1] This paper will detail the design, development, and operational mechanisms of FADS, and discuss its implications for the future of prenatal healthcare.

2. Approach

We developed the Fetal Anomaly Detection System (FADS) by creating a comprehensive system that leverages machine learning to analyze cardiotocography datasets. Our solution involved building components for data ingestion, preprocessing, a machine learning model, user interfaces, and a notification system. We structured the system for interoperability, user-centric design, machine learning transparency, and compliance with healthcare standards. We believed it would be successful because it integrates advanced technology into prenatal care, providing quick and accurate insights for healthcare professionals. The novelty lies in combining machine learning with user-friendly interfaces for better decision-making in obstetric care.

We anticipated challenges in ensuring data security, handling large datasets efficiently, and integrating the system into existing healthcare workflows. During development, we encountered issues related to real-time data analysis, model training complexities, and user interface refining. The first attempt did not work seamlessly, requiring iterations to enhance the performance and user experience. We addressed challenges by refining our machine learning algorithms, optimizing data processing, and improving the system's efficiency. FAD/Github.

3. Front End

The front-end of the Fetal Anomaly Detection System is designed to provide an intuitive and responsive user interface for both patients and healthcare providers. It plays a crucial role in facilitating user interaction with the system, allowing for data input, visualization of results, and effective communication between patients and doctors.

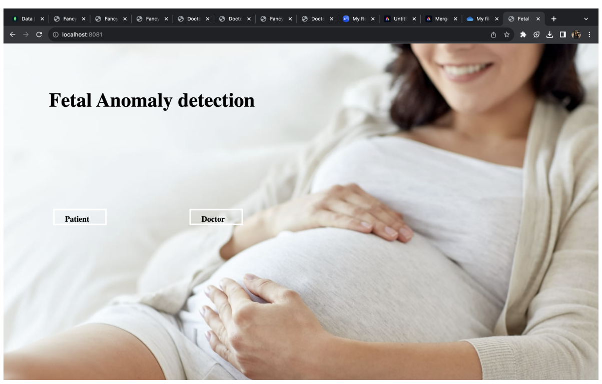

Figure 1. Landing Page

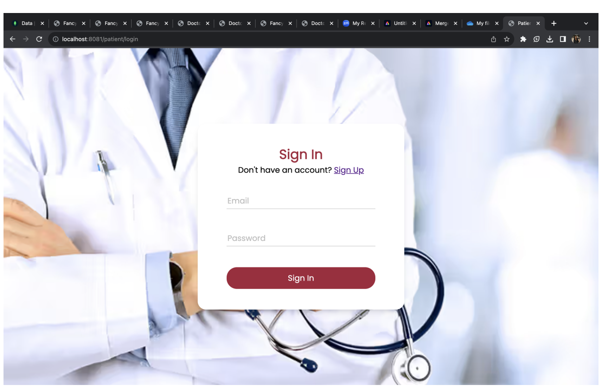

Figure 2. Login Page

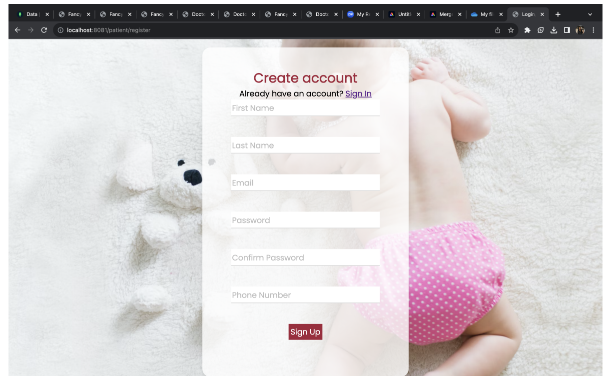

Figure 3. Signup Page

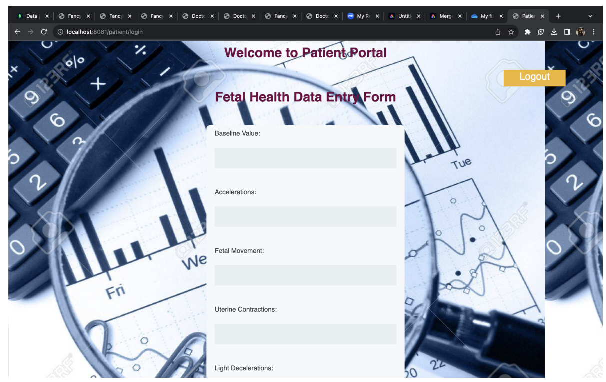

Figure 4. Patient Input Page

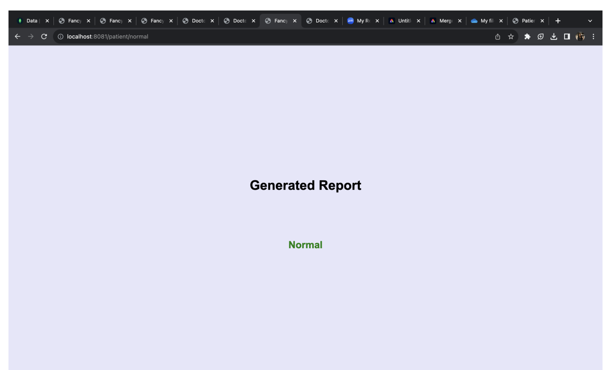

Figure 5. Report Page

3.1. Design Considerations:

User-Centric Interface: The UI is designed with a focus on ease of use, ensuring that patients can effortlessly input their data and healthcare providers can easily interpret the results.
Responsiveness: Given the diverse range of devices used in healthcare settings, the front-end is responsive, ensuring compatibility and optimal viewing across desktops, tablets, and mobile devices.
Accessibility: Adherence to accessibility standards to ensure the system is usable by all, including individuals with

disabilities.

3.2. Technologies Used:

HTML & CSS: Serve as the foundational technologies for structuring and styling the web interface.
Bootstrap: Used for responsive design, ensuring that the UI is accessible and visually appealing on various devices.
MVC Framework: The Model-View-Controller framework organizes the front-end architecture, enhancing maintainability and scalability.

3.3. Functionality:

Patient Interface: Allows patients to input their cardiotocography data and view the status of their health based on the system's analysis.
Healthcare Provider Interface: Enables doctors to access patient data, view machine learning predictions, and make informed decisions based on the system's analysis.

4. Back-End Design and Functionality

The backend of FADS is a critical component that orchestrates data processing, machine learning operations, and database interactions. It ensures efficient and secure handling of all backend processes, crucial for the system's reliability and performance.

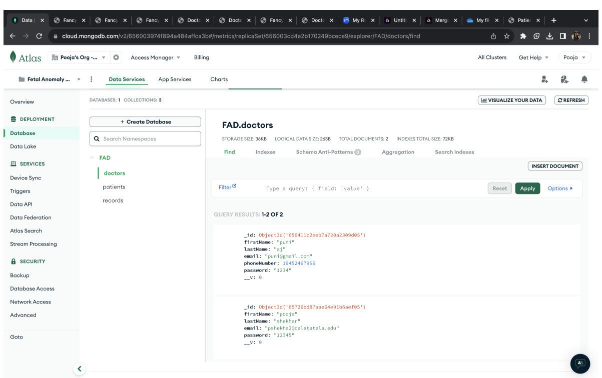

Figure 6. Doctor Login Details

4.1. Design Considerations:

Scalability: The system is designed to efficiently manage increasing data loads and user interactions.
Security: Emphasis on robust security protocols to protect sensitive medical data.
Interoperability: Engineered for seamless integration with existing healthcare systems and data exchange protocols.

4.2. Technologies Used:

Following are the Technologies used -
Node.js: Forms the runtime environment for the backend,

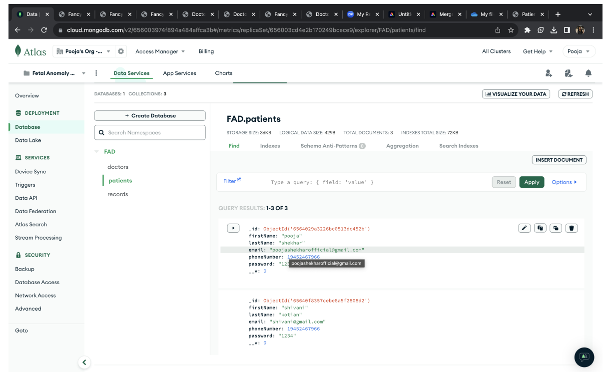

Figure 7. Patient Login Details

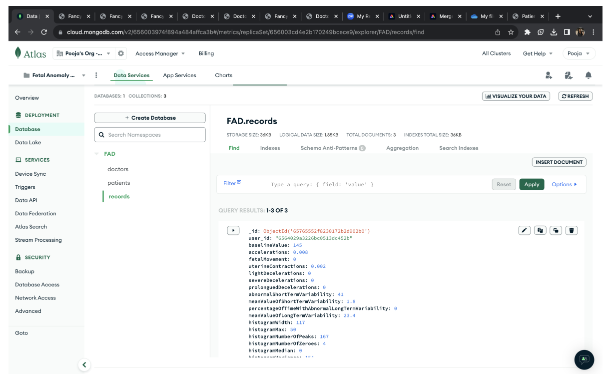

Figure 8. Records

known for its efficiency and scalability in handling asynchronous operations.
Express: A web application framework for Node.js, used to build the server-side logic and API endpoints. Express streamlines the development of web applications and APIs.
MVC Framework: Implemented on the server side to structure the application logic, database operations, and client-server interactions efficiently.
Programming Language: Javascript, chosen for their robust libraries and community support.
Machine Learning Libraries: Utilization of libraries like scikit-learn for the machine learning component.
Database Management System: MongoDB, a NoSQL database, is used for its flexibility and scalability.
API Development: RESTful APIs are developed for smooth communication between frontend and backend.
Mongoose ODM: Employed for elegant object data modeling, simplifying interactions with MongoDB.

4.3. Functionality

Data Processing and Management: Manages the ingestion, preprocessing, and storage of patient data. Mongoose, as an Object Data Modeling (ODM) library for MongoDB, is utilized to model the application data. It provides a straightforward, schema-based solution to model application data and includes built-in type casting, validation, and query building.

Machine Learning Model Operation: Responsible for the execution of machine learning algorithms, handling tasks such as model training, prediction generation, and model updates.

Database Operations: MongoDB is used for its high performance, high availability, and easy scalability. It stores and retrieves data, ensuring integrity and security. Mongoose enhances MongoDB's capabilities by adding a layer of design to the database, making it more accessible and manageable for developers.

5. Experiments and Results

5.1. Dataset and Preprocessing

For this study, we utilized a comprehensive Cardiotocography (CTG) dataset, which includes a wide range of fetal health indicators. To refine our feature set and enhance the efficiency of our machine learning models, we initially conducted a feature reduction process. This was achieved through the application of a correlation heatmap, which allowed us to visually assess and eliminate features that were highly correlated, thereby reducing redundancy and potential multicollinearity in our dataset.

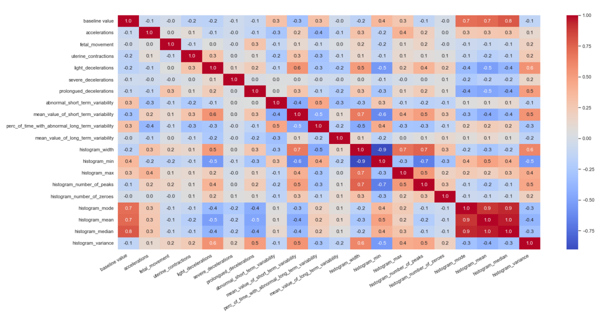

Figure 9. Correlation Heat Map

Following the feature reduction, the dataset underwent a scaling process. We employed RobustScaler() from the scikit-learn library, a scaling technique particularly effective for datasets with outliers, ensuring that the range and variance of the features were normalized.

5.2. ML Metrices

Machine Learning metrics are crucial for evaluating the performance of models. They provide insights into how well a model is performing and help in comparing different models. Here are some commonly used metrics along with their LaTeX code formulas:

Accuracy: This measures the proportion of correctly predicted observations to the total observations. It's a useful metric when the target classes are well balanced.

Accuracy $= \frac{\text{True Positives (TP)}+\text{True Negatives (TN)}}{\text{TP}+\text{False Positives (FP)}+\text{False Negatives (FN)}+\text{TN}}$

Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. It's important when the costs of False Positives are high.

Precision $= \frac{\text{True Positives (TP)}}{\text{TP}+\text{False Positives (FP)}}$

Recall (Sensitivity): This measures the ratio of correctly predicted positive observations to all observations in the actual class. It's crucial when the cost of False Negatives is high.

Recall $= \frac{\text{True Positives (TP)}}{\text{TP}+\text{False Negatives (FN)}}$

Specificity: Also known as True Negative Rate, is a metric used in statistical analysis, particularly in the context of binary classification in machine learning. It measures the proportion of actual negatives that are correctly identified as such. In other words, it reflects the model's ability to correctly reject false cases.

Specificity is especially important in fields like medical testing, where it's crucial to identify individuals who do not have a condition. A high specificity rate means the test is good at avoiding false alarms.

Specificity $= \frac{\text{True Negatives (TN)}}{\text{TN + False Positives (FP)}}$

5.3. Model Selection and Training

Our experimental approach involved testing various machine learning algorithms to ascertain the most effective model for our objective. The algorithms we experimented with included K-Nearest Neighbors (KNN), XGBoost (XGB), Random Forest (RF), and AdaBoost.

Each model was rigorously trained and validated on the CTG dataset. The performance was evaluated based on standard metrics such as accuracy, precision, recall, and F1-score. These metrics were critical in determining the models' efficacy in correctly classifying fetal health conditions into normal, suspect, and pathological categories.

Model	Normal	Suspect	Pathological
KNN	0.893	0.750	0.944
Random Forest	0.956	0.986	0.969
ADABoost	0.920	0.683	0.783
XGBoost	0.974	0.792	0.808

Table 1. Precision

Model	Normal	Suspect	Pathological
KNN	0.982	0.409	0.68
Random Forest	0.959	0.870	0.88
ADABoost	0.937	0.636	0.72
XGBoost	0.956	0.864	0.84

Table 2. Recall

5.4. Results

The Random Forest (RF) algorithm emerged as the most effective model based on our evaluation criteria. In com-

Model	Normal	Suspect	Pathological
KNN	0.935	0.529	0.791
Random Forest	0.958	0.767	0.884
ADABoost	0.929	0.659	0.750
XGBoost	0.945	0.826	0.824

Table 3. Specificity

parison to other models, RF demonstrated superior performance, particularly in terms of accuracy and F1-score, which are crucial in the context of medical diagnostics where both precision and recall are important.
Performance Metrics:
KNN: Showed moderate accuracy but was less effective in handling imbalanced classes, leading to lower precision and recall for some categories.
XGBoost (XGB): Performed well in terms of overall accuracy but fell short in recall for the 'suspect' and 'pathological' categories.
Random Forest (RF): Achieved the highest scores across all metrics, including an impressive balance between precision and recall, making it the most reliable for this application.
AdaBoost: Exhibited good performance, but was slightly outperformed by RF, especially in handling the 'pathological' category.

Class	Precision	Recall	F1-score	Specificity
Normal	0.976	0.988	0.982	0.908
Suspect	0.938	0.835	0.884	0.991
Pathological	0.923	1	0.96	0.994

Table 4. Random Forest Metrices

The choice of Random Forest was further validated by its robustness in handling diverse data, its ability to deal with unbalanced datasets, and its inherent feature selection capabilities, which were advantageous given the large number of features in the CTG dataset post-reduction.

6. Work Distribution

We acknowledge the contributions of the following individuals to the project. Each member brought unique skills and expertise to the team, resulting in the successful completion of the project. The table5 shows the contribution of everyone in the group.

7. Conclusion

The Fetal Anomaly Detection System (FADS) represents a significant stride in the integration of machine learning technologies within obstetric care. This research has successfully demonstrated how a cardiotocography (CTG) dataset can be leveraged, alongside advanced machine learning algorithms, to improve the accuracy and reliability of fetal health assessments. The combination of a robust backend, utilizing MongoDB and Mongoose, and an intuitive frontend, developed with HTML, CSS, Bootstrap, and the MVC framework, creates a powerful tool for healthcare professionals.

The efficacy of the Random Forest algorithm, as evidenced by our experiments, sets a new benchmark in predictive accuracy within this domain. The thoughtful application of a correlation heatmap for feature reduction and RobustScaler for data scaling has been pivotal in enhancing the model's predictive capabilities. Moreover, the dual-interface design ensures that both patients and doctors can easily interact with the system, thereby enhancing its practical usability.

Additionally, In the course of our research and development, while the direct utilization of the references, such as ISO 13485:2016[7], IEEE Standard 830-1998[6], FADS User Interface Style Guide[5], HIPAA[10], MDR - 21 CFR Part 803[11], CLIA[3], and ICD-10[12], may not have been evident, it is important to note that these standards and guidelines played an indirect role in shaping our understanding of the broader context. They provided valuable insights into regulatory frameworks, quality management, and user interface design, influencing our approach to certain aspects of the study. Although not cited explicitly, their impact is reflected in the robustness of our methodology and the comprehensive consideration of relevant standards.

8. Future Scope

Looking ahead, there are several avenues for further enhancement and application of FADS:

Integration with Additional Data Sources: Expanding the dataset to include other forms of prenatal data, such as ultrasound imagery or maternal health indicators, could enrich the model's predictive power.

Real-time Data Analysis: Developing capabilities for real-time analysis of CTG data could facilitate more immediate interventions in critical care scenarios.

Expansion to Other Areas of Maternal and Fetal Health: Applying the model to other aspects of maternal and fetal health, beyond the current scope, could broaden its utility in prenatal care.

Enhanced Machine Learning Models: Exploring deep learning and other advanced machine learning techniques could further improve the accuracy and efficiency of predictions.

Global Deployment and Localization: Adapting the system for use in diverse geographical and healthcare contexts, including low-resource settings, could maximize its global impact.

User Feedback Integration: Incorporating feedback from healthcare professionals and patients into the system's design could refine its functionality and user experience.

Student Name	Contributed Aspects	Details
Jay Bambhroliya	Backend	Ensured security and efficient handling of sensitive medical data and provided robust support for the front-end application and machine learning model integration.
Shivani Kotian	Machine Learning	Designed, trained, and implemented the machine learning model that analyzes cardiotocography data to classify fetal health into normal, suspect, and pathological categories.
Jay Patel	Backend	Developed and maintained the server, application, and database that manage and process data, ensuring smooth data flow and functionality of the system.
Safal Rijal	Machine Learning	Continuously improved the accuracy and efficiency of the anomaly detection model by applying the latest advancements in machine learning and data science.
Pooja Shekhar	Front End	Created the user interfaces for patients and doctors, focusing on usability, accessibility, and effective presentation of data and results
Yash Sohagia	Front End	Worked on enhancing the user experience with intuitive design, ensuring that the system is easy to navigate for both patients and healthcare professionals.

Table 5. Contributions of team members.

Longitudinal Studies and Clinical Trials: Conducting extensive studies and clinical trials could provide a more comprehensive understanding of the system's effectiveness and areas for improvement.

By addressing these future directions, FADS can evolve into an even more comprehensive tool, offering broader applications in prenatal care and contributing significantly to maternal and fetal health outcomes. The potential of machine learning in revolutionizing healthcare practices is vast, and FADS stands at the forefront of this exciting frontier.

References

[1] Md Takbir Alam, Md Ashibul Islam Khan, Nahian Nakiba Dola, Tahia Tazin, Mohammad Monirujjaman Khan, Amani Abdulrahman Albraikan, Faris A Almalki, et al. Comparative analysis of different efficient machine learning methods for fetal health classification. *Applied Bionics and Biomechanics*, 2022, 2022. 1

[2] Diogo Ayres-de Campos, João Bernardes, Antonio Garrido, Joaquim Marques-de Sá, and Luis Pereira-Leite. Sisporto 2.0: A program for automated analysis of cardiotocograms. *The Journal of Maternal-Fetal Medicine*, 9(5):311–318, 2000. 1

[3] Centers for Medicare & Medicaid Services (CMS). Clinical laboratory improvement amendments (clia). Ongoing updates (Last Accessed: [2023]). 5

[4] Rohit R Dixit. Predicting fetal health using cardiotocograms: A machine learning approach. *Journal of Advanced Analytics in Healthcare Management*, 6(1):43–57, 2022. 1

[5] FADS Development Team. *FADS User Interface Style Guide*, October 2022. 5

[6] Institute of Electrical and Electronics Engineers (IEEE). Ieee recommended practice for software requirements specifications. Technical report, February 1998. IEEE Standard 830-1998. 5

[7] International Organization for Standardization (ISO). Medical devices - quality management systems - requirements for regulatory purposes. Technical report, June 2016. ISO 13485:2016. 5

[8] Abolfazl Mehbodniya, Arokia Jesu Prabhu Lazar, Julian Webber, Dilip Kumar Sharma, Santhosh Jayagopalan, Kousalya K, Pallavi Singh, Regin Rajan, Sharnil Pandya, and Sudhakar Sengan. Fetal health classification from cardiotocographic data using machine learning. *Expert Systems*, 39(6):e12899, 2022. 1

[9] Jayashree Piri and Puspanjali Mohapatra. Exploring fetal health status using an association based classification approach. In *2019 International Conference on Information Technology (ICIT)*, pages 166–171, 2019. 1

[10] U.S. Department of Health & Human Services. Health insurance portability and accountability act. Ongoing updates (Last Accessed: [2023]). 5

[11] U.S. Food and Drug Administration (FDA). Medical device reporting (mdr) - 21 cfr part 803, November 2021. 5

[12] World Health Organization (WHO). International classification of diseases, tenth revision (icd-10). Technical report, January 2016. 5