**Enhancing Organ Donation and Transplantation Using Deep Learning and Blockchain**

**REPORT**

**IT5712 PROJECT - I**

***Submitted by***

**AAKASH LS (2020506001)**

**ASHWIN A (2020506016)**

**ADITHYA SUBRAMANI R(2020506006)**

***Guided By***

**Dr M R Sumalatha**

**Head of the Department**



**DEPARTMENT OF INFORMATION TECHNOLOGY MADRAS INSTITUTE OF TECHNOLOGY CAMPUS ANNA UNIVERSITY, CHENNAI-600 044.**

**JULY 2023 - DECEMBER 2023**

**Introduction:**

**Addressing Organ Scarcity:** The growing demand for organ transplantation coupled with a persistent scarcity of viable donor organs has driven the need for innovative solutions. This project aims to tackle the challenges associated with kidney organ donation and transplantation.

**Revolutionizing Technology**: By harnessing the power of Deep Learning (DL) and blockchain technology, this project seeks to revolutionize the organ transplantation process. The integration of a novel DL model, the Diffused Feature Fusion Network (DFFN), enhances organ viability classification, mitigating overfitting, and improving accuracy.

**Data Security and Transparency:** The utilization of blockchain ensures the secure and transparent storage of recipient details .This not only guarantees data integrity but also upholds confidentiality in the transplantation process.

**Streamlined Healthcare**: With the development of a comprehensive web interface, healthcare professionals will have the capability to store patient records directly on the blockchain, simplifying and accelerating the organ transplantation process. An Explainable Artificial Intelligence (XAI) chatbot enhances awareness, providing donors and recipients with comprehensive information on transplantation, waiting times, and viable alternatives. These elements collectively represent a holistic approach to meet the pressing needs in kidney organ donation and transplantation.

**Problem Statement:**

**Problem:** Current kidney organ transplantation faces issues with organ scarcity, data security, and inefficient organ viability classification.

**Solution:** This project addresses these challenges through advanced Deep Learning models, blockchain technology for secure data storage, and an Explainable AI chatbot for transparency and awareness.

**Existing/Related Work:**

*Deep Learning Assisted Kidney Organ Image Analysis for Assessing the Viability of Transplantation by Ali Elmhamudi and Aliyu Abubakar (2022)* in the IEEE SKIMA Journal addresses the need for efficient and secure kidney organ analysis to improve the quality of life for transplant recipients. However, it faces challenges related to complexity, scalability, and data privacy.

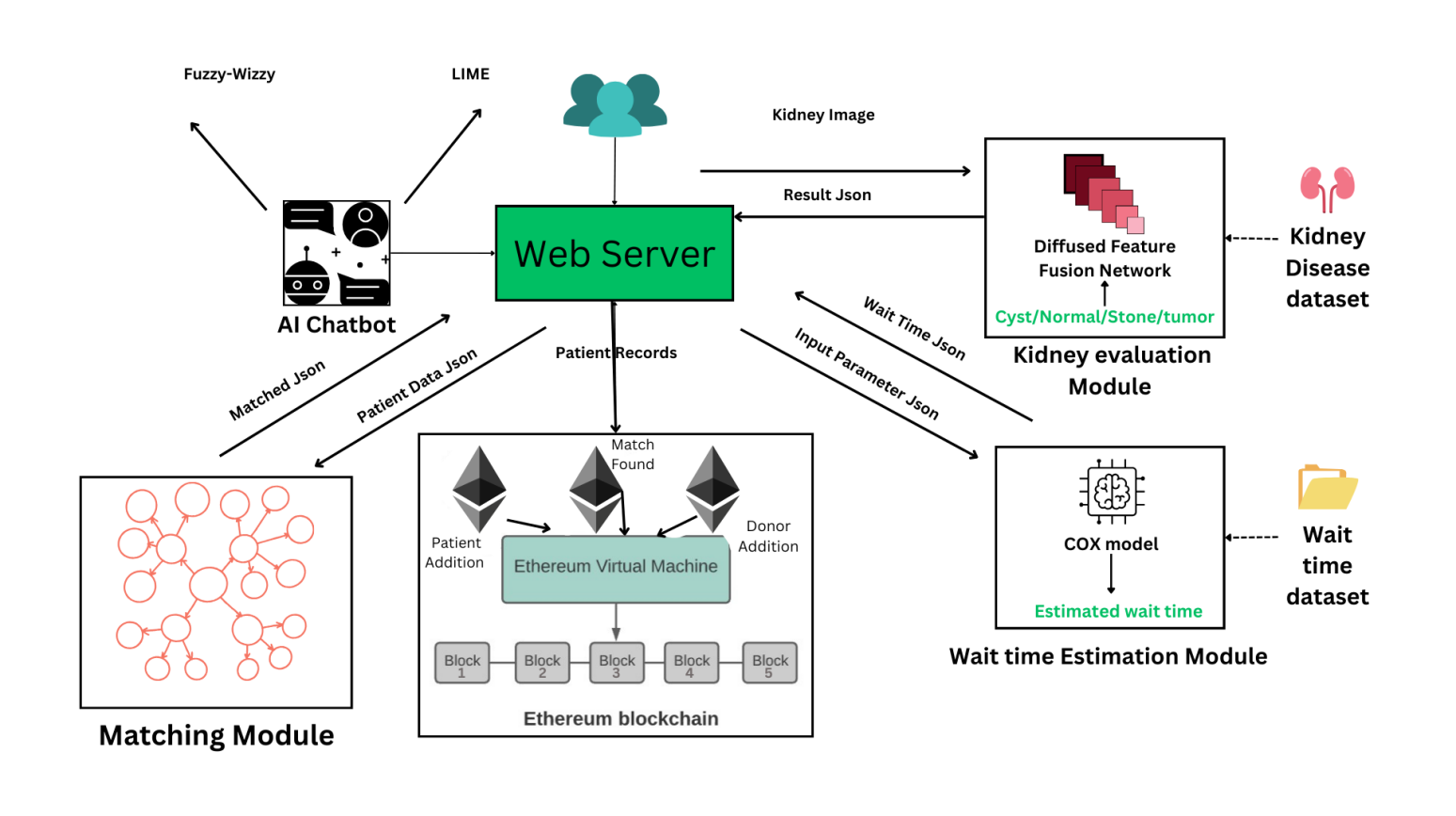
*Organ Bank Based on Blockchain presented by Navjeevan Choudary et al. (2022)* in the IEEE CONECCT Journal introduces a blockchain-based organ bank that enhances transparency, trust, and the efficiency of claims processing. Yet, it grapples with scalability challenges and regulatory uncertainties.

*Intelligent Organ Transplantation System Using Rank Search Algorithm to Serve Needy Recipients by Dr. AdithyaPothan Raj V et al. (2022)* in the IEEE ICSES Journal focuses on resource optimization and improved efficiency. Nonetheless, it faces challenges in terms of complex implementation and potential bottlenecks*.*

*"A Great Way to Start the Conversation": Evidence for the Use of an Adolescent Mental Health Chatbot Navigator for Youth at Risk of HIV and Other STIs by Gabriella Sanabria et al. (2023)* published in SpringerJournal, offers an efficient solution with the potential to improve the quality of life. However, it also grapples with issues of complexity, scalability, and data privacy.

*A Hybrid Approach for Urban Expressway Traffic Incident Duration Prediction with Cox Regression and Random Survival Forests Models authored by Axiang Ke et al. (2017)* in the IEEE Computer Society presents an approach that enhances performance. However, it involves computational complexity.

**Modular Architecture:**

****

The modular architecture of this project comprises four core components:

(1) Deep Learning models for precise organ viability classification (2) a blockchain-based system for secure and transparent recipient data management, (3) a Cox model for accurately predicting waiting times, and (4) an XAI chatbot for knowledge dissemination and interaction. These modules collaboratively enhance kidney organ donation and transplantation by improving organ classification accuracy, ensuring data security, and fostering public understanding and engagement in the process, while accurately predicting waiting times.

**Implemented Modules:**

**1. Deep Learning Module for viability prediction:**

This DL component, powered by the Diffused Feature Fusion Network (DFFN), predicts organ viability by analyzing MRI scans. It ensures that only suitable organs are chosen for transplantation, enhancing the success of transplant procedures while minimizing the risk of complications.

**Algorithm:**

init = tf.keras.initializers.GlorotUniform()

model = Sequential()

# Input layer

model.add(Input(shape=(224, 224, 3)))

# Convolutional layers

model.add(Conv2D(32, (3, 3), activation='relu', kernel\_initializer=init, padding='same'))

model.add(Conv2D(32, (3, 3), activation='tanh', kernel\_initializer=init, padding='same'))

model.add(BatchNormalization())

model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(64, (3, 3), activation='relu', kernel\_initializer=init, padding='same'))

model.add(Conv2D(64, (3, 3), activation='tanh', kernel\_initializer=init, padding='same'))

model.add(BatchNormalization())

model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(128, (3, 3), activation='relu', kernel\_initializer=init, padding='same'))

model.add(Conv2D(128, (3, 3), activation='tanh', kernel\_initializer=init, padding='same'))

model.add(BatchNormalization())

model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(256, (3, 3), activation='relu', kernel\_initializer=init, padding='same'))

model.add(Conv2D(256, (3, 3), activation='tanh', kernel\_initializer=init, padding='same'))

model.add(BatchNormalization())

model.add(MaxPooling2D((2, 2)))

# Flatten layer

model.add(Flatten())

# Fully connected layers

model.add(Dense(256, activation='relu', kernel\_initializer=init))

model.add(Dropout(0.5))

# Output layer

model.add(Dense(4, activation='softmax', kernel\_initializer=init))

**2. Blockchain Smart Contract Module:**

This component ensures secure and transparent management of recipient data, safeguarding sensitive information like HLA values, while promoting trust and accountability within the organ transplantation ecosystem.

**Algorithm:**

// Define the KidneyRecipientRegistry smart contract

contract KidneyRecipientRegistry {

// Define the Recipient struct to store recipient data

struct Recipient {

// Attributes for recipient data

}

// Mapping to store recipients by their unique ID

mapping(uint => Recipient) public recipients;

// Function to add recipient details to the blockchain

function addRecipient() public {

// Check for duplicate recipients

// Add recipient data to the mapping

}

// Function to calculate the estimated waiting time for a recipient

function calculateWaitingTime() public view returns (uint) {

// Calculate the waiting time based on recipient attributes

// Ensure non-negative waiting time

}

}

This smart contract is later deployed using thirdweb so that it can be accessed anywhere and once the web app is created the smart contract can be linked into it and the details of the recipient can be added directly into the deployed contract (blockchain).

**3. Cox-Proportional Hazards Model for Wait Time Prediction:**

The Cox model evaluates patients' previous medical records to estimate their waiting time for kidney transplantation. By considering various parameters, it calculates waiting times, contributing to better allocation, and scheduling of organ transplants while enhancing patients' anticipation and understanding of the process.

**Algorithm:**

# Define categorical and column names

categorical\_columns = []

prediction\_columns = []

classification\_columns = []

# Load models and label encoders

models = []

label\_encoders = {}

classification\_models = {}

model = pickle.load()

scaler = pickle.load()

def loadModels():

# Load label encoders and classification models

for col in categorical\_columns:

label\_encoders[col] = pickle.load()

for model in models:

classification\_models[model] = pickle.load()

def process\_input\_args(input\_data):

# Process input data and apply label encoding and scaling

processed\_input = getData(input\_data)

# Convert categorical data to their equivalent values

processed\_input = convertToNumericalEquivalent(processed\_input)

# Normalize the input parameters

processed\_input = MinMaxScaler(processed\_input, 0, 1)

return processed\_input

def predict\_waiting\_time(input\_parameters):

# Predict waiting time

# remove unwanted parameters

input\_parameters = RemoveUnwantedParmaters(input\_parameters)

# pass the parameters to the trained model

wait\_time = cox.predict(input\_parameters)

return wait\_time

**4.XAI Chatbot**

The XAI chatbot answers recipient and donor queries, offering insights into organ transplantation and factors affecting waiting times, fostering awareness, and increasing interaction in the organ donation community.

**Algorithm:**

function chatbot\_response(user\_query):

best\_match = None

best\_score = 0

# Iterate through the dataset to find the best matching user query

for each row in data:

score = fuzzywuzzy.ratio(user\_query.lower(), row['User\_Query'].lower())

# If the current query has a higher match score, update best\_match and best\_score

if score > best\_score:

best\_score = score

best\_match = row['User\_Query']

# Check if a suitable query was found

if best\_match is not None:

response = data[data['User\_Query'] == best\_match]['Response'].iloc[0]

else:

# If no suitable query is found, provide a default response

response = "Sorry, I didn't understand your question."

return response

**Work for Current Semester:**

For the remainder of this semester, the focus will be on enhancing the efficiency of the Diffused Feature Fusion Network DFFN. This involves optimizing the backend processes to ensure faster and more responsive calculations and trying out new possible combinations of architectural layers of CNN model. Also the chatbot that helps users with general organ donation related queries has to be enhanced using more training data to increase the efficiency of the questions answered. Additionally, the implementation of a web interface will be initiated, providing a user-friendly platform for interacting with the Cox model. This interface will facilitate easier input of parameters and retrieval of predictions, contributing to a more seamless user experience.

**Planned Work for the Next Semester:**

The upcoming semester will bring about significant developments in the project. The primary focus will be on implementing a matching API module for donors and recipients. This module will consider various parameters, including location, to find the best match efficiently. Simultaneously, efforts will be directed towards the implementation of blockchain technology in a test net environment. This marks a crucial step towards establishing a secure and transparent system for organ transplantation. The goal for the next semester is to integrate all developed modules into a comprehensive and functional Organ Transplantation Network.

**Conclusion:**

In conclusion, the ongoing semester lays the groundwork for optimizing the Cox model and introducing a user-friendly web interface. These improvements will contribute to the model's accessibility and usability. Looking ahead, the subsequent semester will witness the integration of a matching API module, leveraging location data for enhanced donor-recipient pairing. The incorporation of blockchain technology will further strengthen the project's security and transparency. The culmination of these efforts aims to establish a fully operational Organ Transplantation Network, showcasing the potential for innovation and advancement in the field.