**VOICE ENABLED PATIENT DOCUMENTATION AND ASSISTANCE SYSTEM**

*Submitted in partial fulfilment of the requirements for the degree of*

**Bachelor of Technology**

# In

**Computer Science and Engineering**

*By*

Saraswathi Bavadharini S

20BCE0339

Under the guidance of

Prof.Manoov R

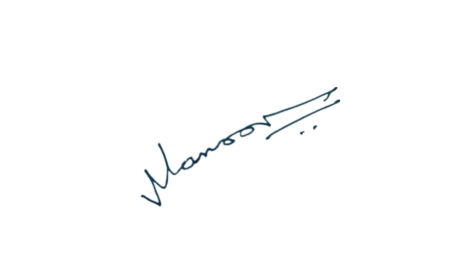
School of Computer Science and Engineering

VIT, Vellore.



# 

GUIDE

Prof.Manoov R

Department of IoT

School of Computer Science and Engineering

Vellore Institute of Technology

Place : Vellore

Date : 05-04-24

(Signature of Guide)

**PROBLEM STATEMENT**

In modern healthcare settings, the documentation of patient information and medical records is a critical aspect of delivering quality care and ensuring patient safety. However, traditional methods of data entry and retrieval often prove to be inefficient, error-prone, and time-consuming. Healthcare professionals, particularly nurses, face challenges in accessing patient details and summarizing medical documents quickly and accurately, leading to potential delays in care delivery and compromised workflow efficiency. Additionally, the increasing volume of patient data and the complexity of medical documentation further exacerbate these challenges, highlighting the need for innovative solutions to streamline the documentation process and enhance accessibility to critical information.

The problem at hand revolves around the inefficiencies and limitations of current documentation practices in healthcare settings, which hinder the timely and accurate retrieval of patient information and medical document summaries. There is a pressing need for a comprehensive solution that leverages advanced technologies, such as natural language processing (NLP), speech recognition, and artificial intelligence (AI), to enable healthcare professionals to access patient details and summarize medical documents efficiently and accurately. Furthermore, the solution must address concerns related to data security, privacy, interoperability with existing healthcare systems, user training, ethical considerations, and scalability to ensure its viability and effectiveness in real-world healthcare environments. Thus, the problem statement encapsulates the imperative to develop a voice-enabled patient documentation and assistance system that overcomes the limitations of current practices and empowers healthcare professionals to deliver optimal care through streamlined documentation processes.

**ABSTRACT**

The voice-enabled patient documentation and assistance system presented in this project aim to revolutionize the workflow of medical professionals, particularly nurses, by leveraging cutting-edge technologies in natural language processing and speech recognition. Through intuitive voice commands, users can efficiently retrieve patient details and obtain summarized versions of uploaded medical documents. The system integrates with OpenAI's powerful GPT-3 model for generating patient details based on voice prompts and utilizes BERT-based extractive summarization techniques to condense lengthy medical documents. By providing both textual and auditory outputs, the system enhances accessibility and efficiency in healthcare settings, ultimately improving patient care and streamlining documentation processes.

**LITERATURE REVIEW**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ref.** | **Paper Title** | **Author**  **and Year** | **Research**  **Question** | **Methodology/Approach** | **Limitation** |
| 1 | Benefits, Limits, and Risks of GPT-4 as an AI Chatbot for Medicine | Lee *et al.*  (2023) | Using GPT 4 and similar Generative AI tools such as Google LaMDA and GPT 3.5 in Medical conversational ChatBOTs | ChatBOTs use the GPT 4 LLM to retrieve ansers for user queries from web and this model has been tested out and found to have an accuracy of over 90% | Authenticity of Data obtained from the web by GPT models. |
| 2 | Speech emotion recognition using machine learning - A systematic review | Madanian *et al*. (2023) | Properties, methodology and working of SER model and analysing its efficiency . | Training a speech recognition (SR) system, including language corpus, nursing activities, clinical conversations, and accents. It compared documentation time and error rates between SR- generated records and keyboard entry, | The paper may overlook non-ML approaches and interdisciplinary perspectives in SER, and while it discusses challenges and solutions, it may not encompass all potential obstacles or emerging  trends. |
| 3 | Development  of the Speech- to-Text Chatbot | Shakhovska  *et al.*  (2019) | Utilizing the  Google Speech- to-Text API.data | The proposed method  involves employing prefix functions and | The study may  potentially overlook |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Interface Based on Google API |  | from social networks to focused on remote and local storage  processes. | hashing algorithms for keyword searching and verb ending identification in chatbot conversations | alternative methods and their effectiveness in real-world applications. |
| 4 | Machine learning-based speech recognition system for nursing documentation  – A pilot study | Lee *et al.*  (2023) | Machine learning-based speech recognition (SR) system's effectiveness in reducing nursing documentation workload in a psychiatry ward. | The study collected language corpus, nursing activities, clinical conversations, and accent data for SR system training in four sessions and achieved model had an accuracy score of 87.06% to 95.07% across sessions. | The study's findings are based on a pilot implementation in a psychiatry ward, potentially limiting generalizability to other nursing specialties or healthcare settings. |
| 5 | Intelligent speech technologies for transcription, disease diagnosis, and medical equipment interactive control in smart hospitals: A  review | Zhang *et al.*  (2023) | To explore the application and potential of intelligent speech technology (IST) in  addressing medical resource shortages and improving healthcare  efficiency amid | The paper introduces IST's procedure and system architecture, reviews its applications in smart hospitals, and presents a case study on stroke patient care.  Additionally, it proposes a novel medical voice analysis system architecture. | Challenges include noise interference and pronunciation differences, which may hinder the widespread application of IST in hospitals. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | challenges like noise interference and pronunciation  differences. |  |  |
| 6 | The Capability of ChatGPT in Predicting and Explaining Common Drug- Drug Interactions | Juhi A *et al.*  (2023) | To assess the effectiveness of ChatGPT in predicting and explaining common drug- drug interactions (DDIs) | Utilized 40 DDI lists from literature to converse with ChatGPT using two-stage questions, assessing responses' correctness with pharmacologists' consensus. | ChatGPT provided incomplete guidance at times, necessitating further improvement for patient use regarding DDI  awareness. |
| 7 | Deep Cross- Corpus Speech Emotion Recognition: Recent Advances and Perspectives | Zhang *et al.*  (2021) | To comprehensively survey the state- of-the-art techniques in cross-corpus speech emotion recognition (SER),  particularly focusing on deep learning methods associated with supervised, unsupervised,  and semi- | The paper reviews existing literature on speech emotion databases, traditional methods for cross-corpus SER, recent advances in deep learning techniques, and discusses challenges and future directions in the field. | Challenges such as natural data scarcity, multimodal integration, and limitations of deep learning techniques, potentially affecting the comprehensiveness of its findings are discussed. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | supervised  learning. |  |  |
| 8 | Natural Language Supervision for General- Purpose Audio Representations | Wang *et al.*  (2023) | Propose Contrastive Language-Audio Pretraining (CLAP) for joint audio-text representation learning, enabling Zero- Shot inference across 26 downstream tasks, surpassing state-of-the-art models for general-purpose audio representation | Utilize two innovative encoders for audio and text, trained with Contrastive Learning to create multimodal representations. Train audio encoder (HTSAT- 22) on 22 tasks and adapt GPT2 for text encoding, enabling joint learning of representations in a multimodal space for Zero-Shot inference. | Limited evaluation on tasks with true Zero-Shot setup.  Increased training pair diversity affects performance variably across domains. |

**PROPOSED METHODOLOGY**

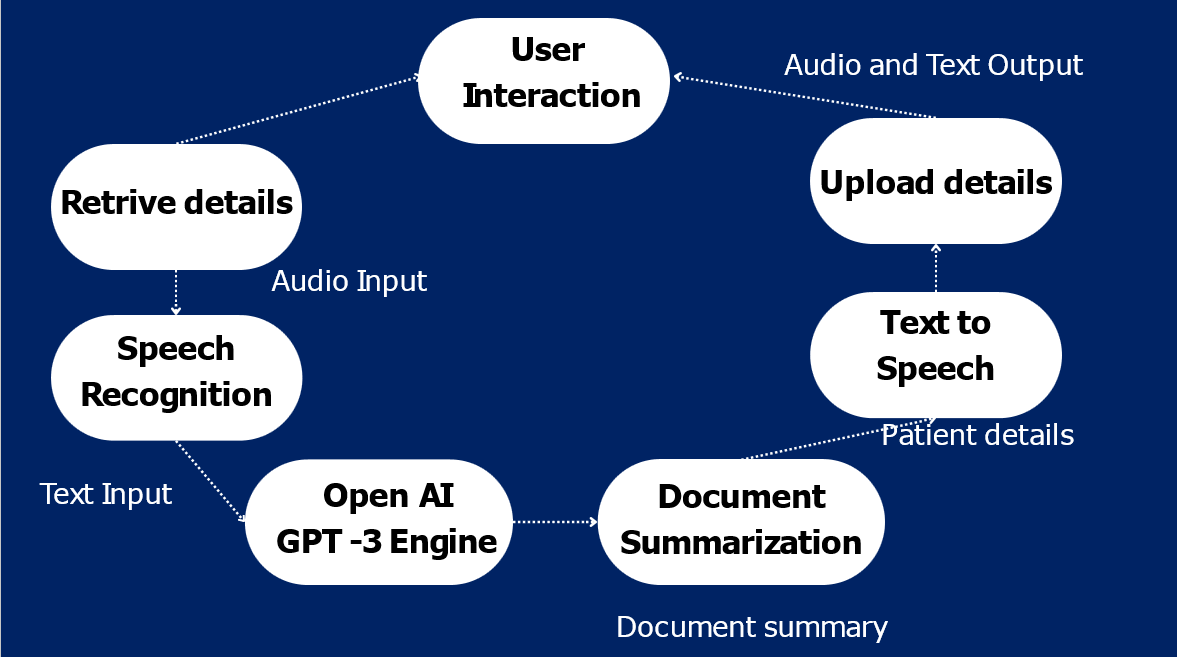
****

Fig 1 : Workflow of Project

The various modules under the methodology include :

* *Speech Recognition:*
  + Utilizes the speech\_recognition library to capture audio input from users via microphones.
  + The transcribe\_audio\_to\_text function converts the captured audio into text using Google's Speech Recognition service.
  + Implements robust error handling to gracefully manage exceptions during the speech recognition process.
* *OpenAI Integration:*
  + Integrates with OpenAI's GPT-3 API for generating patient details based on provided prompts.
  + The generate\_response function sends prompts to the GPT-3 API to generate responses.
  + Handles exceptions that may occur during the response generation process.
* *Text Summarization:*
  + Implements text summarization using the bert-extractive-summarizer library to generate concise summaries of medical documents.
  + The summarize\_text function summarizes the uploaded medical documents.
  + Provides individual summaries for each document and a combined summary.
* *User Interface (UI):*
  + Develops the user interface using Streamlit, allowing users to interact with the system through a web-based application.
  + Offers actions such as retrieving patient details and uploading patient documents through a sidebar navigation menu.
  + Displays prompts, input fields, and document upload functionality using Streamlit components.
  + Presents patient details and document summaries in the UI for easy access.
* *Text-to-Speech:*
  + Employs the pyttsx3 library to convert text responses and document summaries into audible speech for users.
  + The speak\_text function converts text into speech and plays it using the system's audio output.
  + Provides spoken feedback to users, enhancing accessibility and user experience.
* *Error Handling:*
  + Implements robust error handling mechanisms throughout the system to handle exceptions gracefully.
  + Detects and displays errors related to speech recognition, OpenAI integration, text summarization, and text-to-speech conversion.
  + Ensures the system remains stable and functional even in the presence of unexpected errors.

**IMPLEMENTATION**

***PYTHON CODE***

import streamlit as st

import openai

import pyttsx3

import speech\_recognition as sr

from summarizer import Summarizer

import time

openai.api\_key = "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"

engine = pyttsx3.init()

def transcribe\_audio\_to\_text(audio):

recognizer = sr.Recognizer()

try:

return recognizer.recognize\_google(audio)

except sr.UnknownValueError:

st.error("Could not understand audio")

except sr.RequestError as e:

st.error(f"Could not request results from Google Speech Recognition service; {e}")

except Exception as e:

st.error(f"Error occurred during speech recognition: {e}")

return None

def generate\_response(prompt):

try:

response = openai.Completion.create(

engine="gpt-3.5-turbo-instruct",

prompt=prompt,

max\_tokens=150,

n=1,

stop=None,

temperature=0.7,

)

return response["choices"][0]["text"]

except Exception as e:

st.error(f"Error occurred during response generation: {e}")

return None

def speak\_text(text):

global engine

try:

while engine.isBusy():

time.sleep(0.1) # Wait for 0.1 seconds before checking again

engine.say(text)

engine.runAndWait()

print("Text spoken successfully")

except Exception as e:

st.error(f"Error occurred during text-to-speech conversion: {e}")

def summarize\_text(text):

model = Summarizer()

summary = model(text, min\_length=50)

return ''.join(summary)

def main():

st.title("Doctor's Assistant (Nurse)")

st.sidebar.title("Actions")

action = st.sidebar.radio("Select Action", ("Retrieve Patient Details", "Upload Patient Documents"))

if action == "Retrieve Patient Details":

st.write("Please state the patient's name or ID")

with sr.Microphone() as source:

recognizer = sr.Recognizer()

st.write("Listening...")

audio = recognizer.listen(source)

st.write("Processing...")

transcription = transcribe\_audio\_to\_text(audio)

if transcription:

st.write(f"Retrieving details for patient: {transcription}")

prompt = f"Retrieve details for patient {transcription}"

response = generate\_response(prompt)

if response:

st.write("Patient Details:")

st.write(response)

speak\_text(response)

elif action == "Upload Patient Documents":

st.write("Please select the patient and upload the documents:")

patient\_name = st.text\_input("Enter Patient Name or ID")

uploaded\_files = st.file\_uploader("Upload Documents", type=['pdf', 'docx', 'txt'], accept\_multiple\_files=True)

if uploaded\_files:

st.write(f"{len(uploaded\_files)} document(s) uploaded successfully for patient: {patient\_name}")

combined\_summary = "" # Variable to store combined summaries

for i, uploaded\_file in enumerate(uploaded\_files):

text = uploaded\_file.read().decode("utf-8") # Read the file content

summary = summarize\_text(text)

st.write(f"Document {i+1} Summary:")

st.write(summary)

combined\_summary += summary + "\n" # Append individual summary to combined summary

if st.button("Show Combined Summary"):

st.write("Combined Summary:")

st.write(combined\_summary)

speak\_text(combined\_summary) # Convert combined summary to speech and output

if \_\_name\_\_ == "\_\_main\_\_":

main()

**RESULTS**

The implemented system successfully enables medical professionals to:

* Retrieve patient details by voice command. Upload medical documents for summarization.
* Obtain summarized document content in both text and audio formats for multiple documents .
* Experience streamlined workflow and enhanced efficiency in accessing patient information.