REPORT PROJECT

HealthAI: Intelligent Healthcare Assistant Using IBM Granite Companion (React)

YEAR : 2024 – 2025

COLLEGE NAME : K.C.S KASI NADAR COLLEGE OF ARTS & SCIENCE

CODE : UNM203

DEPARTMENT : COMPUTER SCIENCE

PROGRAM : B.C.A

SEMESTER: V

PROJECT SUBMITTED TO: UNIVERSITY OF MADRAS / NAAN MUDALVAN

Course Name : GEN AI WITH IBM

TEAM LEADER: Chenchu Gayathri C

MEMBERS:

1. CHENCHU GAYATHRI C

2. MANJU SHREE G

3. MADHUMITHA S

4. VEDHAPRIYA B

HealthAI: Intelligent Healthcare Assistant Using IBM Granite Companion (React)

Introduction

Health is a dynamic state of complete physical, mental, and social well-being, not merely the absence of disease or infirmity, as defined by the World Health Organization. It is a multidimensional concept influenced by biological, psychological, environmental, and sociocultural factors, encompassing one's ability to live a fulfilling and productive life by realizing their potential and coping with life's challenges.

The WHO Definition of Health

A Holistic State: Health is viewed as a positive concept involving the integration of three key dimensions.
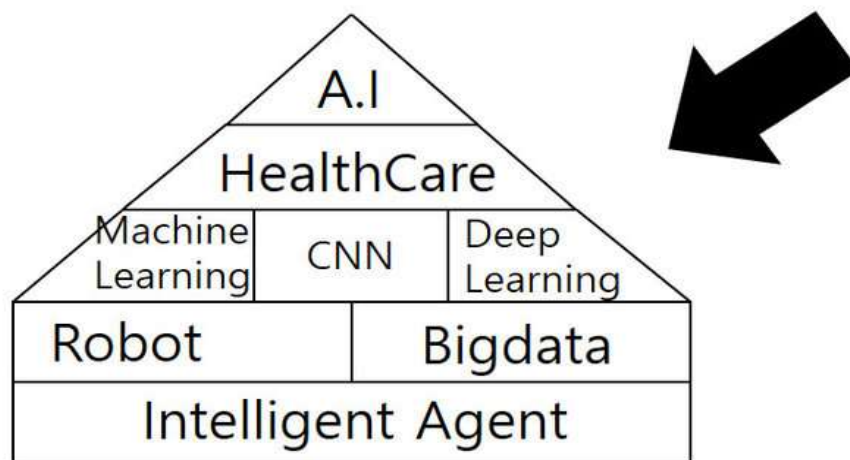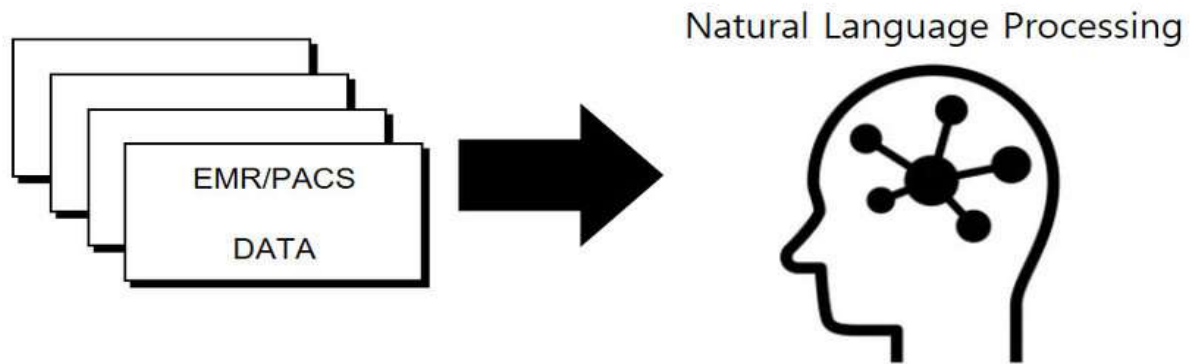
More Than Absence of Disease: It's not just about not being sick, but about thriving in all aspects of life.

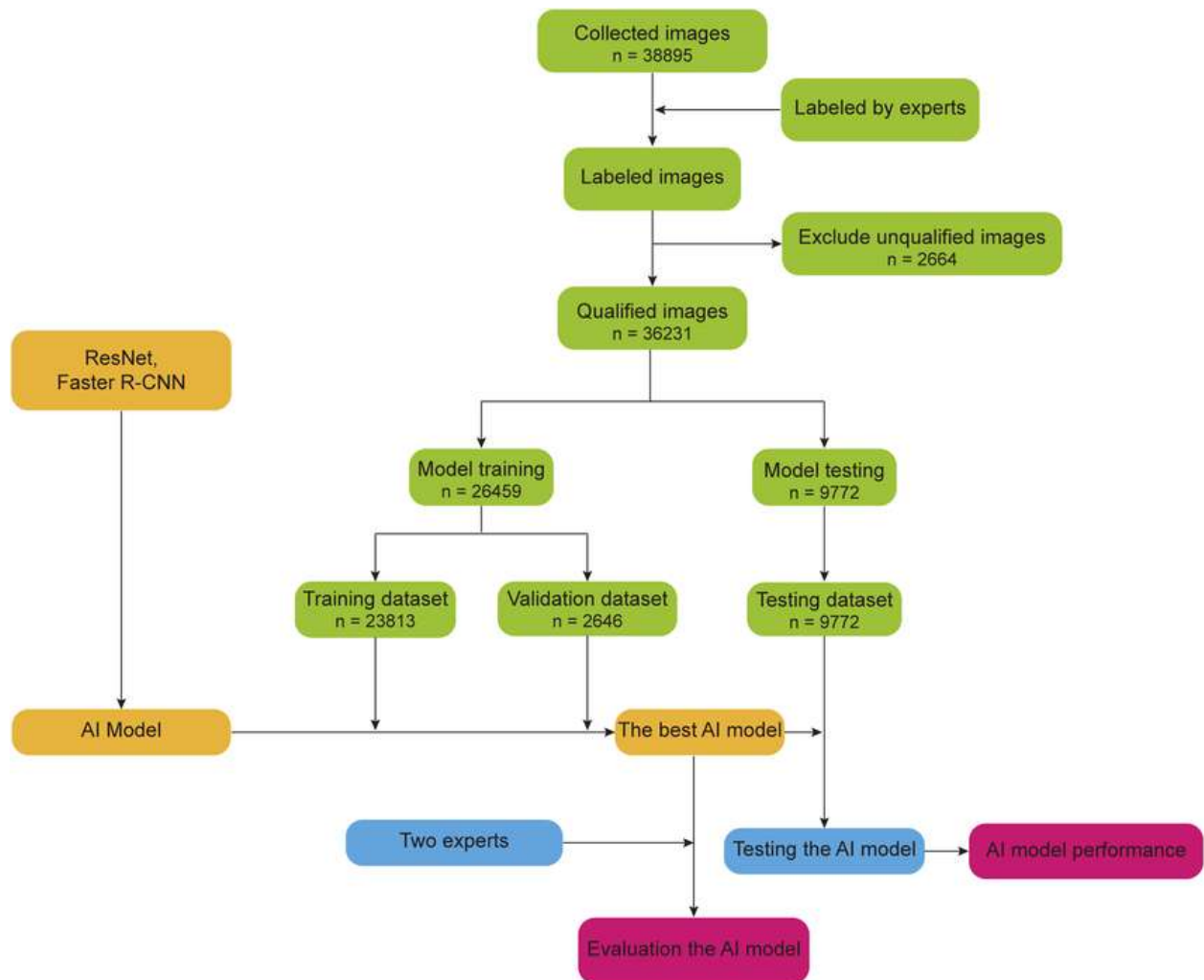The Three Dimensions of Health

Physical Health: Involves the proper functioning of the body's anatomical structure and physiological systems, allowing all parts to work harmoniously.

Mental Health: Encompasses the ability to learn, think clearly, handle daily life events, work productively, and manage stress.

Social Health: Relates to one's ability to interact positively with others, form relationships, and function effectively within their community and society.

Natural Language Processing

EMR/PACS DATA

A.I
HealthCare
Machine Learning | CNN | Deep Learning
Robot | Bigdata
Intelligent Agent

Flowchart:

Sample Coding for Health AI:

```
Import gradio as gr

Import torch

From transformers import AutoTokenizer, AutoModelForCausalLM


# Load model and tokenizer

Model_name = "ibm-granite/granite-3.2-2b-instruct"

Tokenizer = AutoTokenizer.from_pretrained(model_name)

Model = AutoModelForCausalLM.from_pretrained(

    Model_name,

    Torch_dtype=torch.float16 if torch.cuda.is_available() else torch.float32,

    Device_map="auto" if torch.cuda.is_available() else None

)


If tokenizer.pad_token is None:

    Tokenizer.pad_token = tokenizer.eos_token


Def generate_response(prompt, max_length=1024):

    Inputs = tokenizer(prompt, return_tensors="pt", truncation=True, max_length=512)


    If torch.cuda.is_available():

        Inputs = {k: v.to(model.device) for k, v in inputs.items()}


    With torch.no_grad():

        Outputs = model.generate(

            **inputs,

            Max_length=max_length,
```

```python
        Temperature=0.7,

        Do_sample=True,

        Pad_token_id=tokenizer.eos_token_id

    )


    Response = tokenizer.decode(outputs[0], skip_special_tokens=True)

    Response = response.replace(prompt, "").strip()

    Return response


Def disease_prediction(symptoms):

    Prompt = f"Based on the following symptoms, provide possible medical conditions and general medication suggestions. Always emphasize the importance of consulting a doctor for proper diagnosis.\n\nSymptoms: {symptoms}\n\nPossible conditions and recommendations:\n\n**IMPORTANT: This is for informational purposes only. Please consult a healthcare professional for proper diagnosis and treatment.**\n\nAnalysis:"

    Return generate_response(prompt, max_length=1200)


Def treatment_plan(condition, age, gender, medical_history):

    Prompt = f"Generate personalized treatment suggestions for the following patient information. Include home remedies and general medication guidelines.\n\nMedical Condition: {condition}\nAge: {age}\nGender: {gender}\nMedical History: {medical_history}\n\nPersonalized treatment plan including home remedies and medication guidelines:\n\n**IMPORTANT: This is for informational purposes only. Please consult a healthcare professional for proper treatment.**\n\nTreatment Plan:"

    Return generate_response(prompt, max_length=1200)


# Create Gradio interface

With gr.Blocks() as app:

    Gr.Markdown("# Medical AI Assistant")
```

```python
        Gr.Markdown("**Disclaimer: This is for informational purposes only. Always consult healthcare professionals for medical advice.**")

    With gr.Tabs():
        With gr.TabItem("Disease Prediction"):
            With gr.Row():
                With gr.Column():
                    Symptoms_input = gr.Textbox(
                        Label="Enter Symptoms",
                        Placeholder="e.g., fever, headache, cough, fatigue…",
                        Lines=4
                    )
                    Predict_btn = gr.Button("Analyze Symptoms")


                With gr.Column():
                    Prediction_output = gr.Textbox(label="Possible Conditions & Recommendations", lines=20)


            Predict_btn.click(disease_prediction, inputs=symptoms_input, outputs=prediction_output)


        With gr.TabItem("Treatment Plans"):
            With gr.Row():
                With gr.Column():
                    Condition_input = gr.Textbox(
                        Label="Medical Condition",
                        Placeholder="e.g., diabetes, hypertension, migraine…",
                        Lines=2
```

```
        )
        Age_input = gr.Number(label="Age", value=30)
        Gender_input = gr.Dropdown(
            Choices=["Male", "Female", "Other"],
            Label="Gender",
            Value="Male"
        )
        History_input = gr.Textbox(
            Label="Medical History",
            Placeholder="Previous conditions, allergies, medications or None",
            Lines=3
        )
        Plan_btn = gr.Button("Generate Treatment Plan")


    With gr.Column():
        Plan_output = gr.Textbox(label="Personalized Treatment Plan", lines=20)


    Plan_btn.click(treatment_plan, inputs=[condition_input, age_input, gender_input,
history_input], outputs=plan_output)


App.launch(share=True)
```

Project Execution :

Disease Prediction   Treatment Plans

Enter Symptoms

Fever

Analyze Symptoms

Possible Conditions & Recommendations

consult a healthcare professional for appropriate medication.

General Recommendations:

- Monitor and maintain body temperature using a thermometer.
- Stay hydrated, especially if experiencing vomiting or diarrhea.
- Rest and manage fatigue.
- Separate the sick person from others to prevent spreading infections.
- Follow the advice of healthcare professionals regarding medication use.

Please remember: This information is for general guidance only and should not replace a consultation with a healthcare provider. Always consult a doctor for accurate diagnosis and treatment tailored to your specific situation.

Gender

Male ▼

Medical History

Previous conditions

Generate Treatment Plan

assistance to quit smoking through behavioral therapy, nicotine replacement therapies, or prescription medications if needed.

10. Alcohol Consumption:
   - Home Remedy: Limit alcohol intake to no more than one drink per day for women and two drinks per day for men.
   - Recommendation: Adopt moderate alcohol consumption guidelines advised by your healthcare provider or organizations like the ADA.

By following this personalized treatment plan, you can better manage your diabetes condition while incorporating home remedies and general medication guidelines. Never hesitate to consult your healthcare professional for personalized advice tailored to your unique needs.

Project demo link:

IBM-PROJ/Demo video(HealthAI).mp4 at main · manju12092004/IBM-PROJ

project source code:

https://github.com/manju12092004/IBM-PROJ/blob/main/welcome_to_colab.py