Enhancing Access to Medical Care: A Mobile Healthcare App Solution for Improved Health Accessibility

Nishil Kulkarni
Department of Electrical and Electronics Engineering
Birla Institute of Technology and Science, Pilani - K.K Birla Goa Campus
Goa, India
f20212499@goa.bits-pilani.ac.in

Anita Agrawal
Department of Electrical and Electronics Engineering
Birla Institute of Technology and Science, Pilani - K.K Birla Goa Campus
Goa, India
aagrawal@goa.bits-pilani.ac.in

***Abstract*—This paper introduces "HealthAssist," a mobile application revolutionizing symptom-based health assessment. With the aim of empowering users to self-assess their health status and make informed healthcare decisions, HealthAssist provides a convenient platform for symptom assessment and health management. Additionally, we present a novel algorithm for analyzing user-input symptoms and diagnosing the closest illness, thus offering users a reliable tool for personalized health assessment and management.**

***Index Terms*—health diagnostics, symptom analysis, mobile health applications, healthcare algorithms, user interface design**

I. Introduction

In recent years, technology integration into healthcare has revolutionized how individuals manage their health and well-being. Mobile applications, in particular, have emerged as powerful tools for facilitating access to medical information, monitoring health metrics, and aiding in diagnosis. Among the myriad of health-related applications, one area of increasing interest is symptom-based health assessment, where users can input their symptoms and receive insights into potential illnesses or health conditions.

This research paper introduces "HealthAssist," a mobile application designed to streamline the process of symptom-based health assessment. The primary objective of HealthAssist is to empower users to self-assess their health status, identify potential illnesses, and make informed decisions regarding their healthcare.

The development of HealthAssist was motivated by the need for accessible and user-friendly solutions to address common health concerns. With the prevalence of smartphones and the growing demand for personalized healthcare solutions, HealthAssist aims to bridge the gap between individuals and healthcare resources, offering a convenient platform for symptom assessment and health management.

II. Design and methodology

The primary objective of this research is to develop an algorithmic approach for symptom-based illness diagnosis implemented within a mobile application. The aim is to create a user-friendly tool enabling individuals to input their symptoms and receive a probable diagnosis, assisting users and healthcare professionals in early symptom recognition and triage.

The data collection procedure involved systematically extracting information from online health platforms such as WebMD. Comprehensive data on illnesses, associated symptoms, and severity levels were gathered from publicly available sources. No specific sampling criteria were applied, ensuring inclusivity of all reported conditions and symptoms. Given the secondary nature of the data, considerations of validity and reliability were not applicable.

A. Algorithm Construction

The algorithm utilizes a severity matrix, assigning ratings to various symptoms associated with common illnesses such as Common Cold, Flu, Gastroenteritis, Sinusitis, Migraine, and Urinary Tract Infection (UTI). The system identifies the most probable illness by comparing user-reported symptoms to the severity matrix, facilitating prompt medical intervention.

The provided matrix represents the severity ratings of various symptoms for common illnesses. Each row corresponds to a specific symptom, while each column corresponds to a particular illness. The severity ratings are represented by numerical values ranging from 0 to 5, where 0 indicates the absence of the symptom and higher values indicate increasing severity.

TABLE I: Symptom Severity Ratings for Common Illnesses

Symptom	**Illness**					
	Common Cold	**Flu**	**Gastroenteritis**	**Sinusitis**	**Migraine**	**UTI**
Fever	3	4	3	2	0	3
Headache	1	4	3	5	5	0
General Aches, Pain	2	4	3	4	1	3
Fatigue/Weakness	3	5	4	4	2	3
Runny Nose	4	2	0	5	0	0
Stuffy Nose/ Nasal Blockage	4	2	0	5	0	0
Sneezing	4	2	0	1	0	0
Sore Throat	4	2	0	2	0	0
Chest discomfort	3	5	0	3	0	0
Cough	3	5	0	3	0	0
Itchy/ Watery Eyes	0	0	0	2	0	0
Watery Diarrhea	0	0	5	0	0	0
Vomiting	0	0	5	0	1	0
Stomach Pain	0	0	5	0	0	0
Nausea	0	0	5	0	4	0
Being more thirsty	0	0	4	0	0	3
Sensitive to light, sound, or smell	0	0	3	2	5	0
Mood changes	0	0	0	1	1	0
Constipation	0	0	0	0	0	0
Bloating	0	0	0	0	0	0
Food cravings	0	0	0	0	3	0
Lack of appetite	0	0	0	0	4	0
Redness in the eyes	0	0	0	0	0	0
Frequent urination	0	0	0	0	0	5
Pain or burning while urinating	0	0	0	0	0	5
Blood in the urine	0	0	0	0	0	4

For example, the cell at the intersection of the ”Fever” row and the ”Flu” column has a value of 4, indicating that fever is rated as severe (4) for the flu. Similarly, the cell at the intersection of the ”Runny Nose” row and the ”Common Cold” column has a value of 4, indicating that runny nose is rated as severe (4) for the common cold.

1) Distance Calculation and Comparison: The distance metric measures the straight-line distance between two points in a multi-dimensional space. In this context, the input vector represents the symptoms gathered from the user, while the target vectors correspond to the columns (illnesses) of the severity matrix. To calculate the Euclidean distance, the algorithm iterates over each element of the input and target vectors, computing the squared difference between corresponding elements and summing them. Finally, the square root of this sum is taken to obtain the Euclidean distance, which serves as a measure of similarity between the input and target vectors. By calculating the Euclidean distance between the input and target vectors, the algorithm quantifies the similarity between the user's symptoms and each illness profile.

The Euclidean distance between the input vector $\mathbf{I}$ and a target vector $\mathbf{T}_i$ can be calculated as:

$$\text{Distance}(\mathbf{I}, \mathbf{T}_i) = \sqrt{\sum_{j=1}^{m} (\mathbf{I}_j - \mathbf{T}_{ij})^2}$$

where:

- m is the dimensionality of the vectors (i.e., the number of symptoms),
- $\mathbf{I}_j$ represents the jth element of the input vector $\mathbf{I}$ (the severity of the jth symptom reported by the user),
- $\mathbf{T}_{ij}$ represents the jth element of the ith target vector $\mathbf{T}_i$ (the severity of the jth symptom for the ith illness).

The algorithm then selects the index k corresponding to the target vector $\mathbf{T}_k$ with the minimum distance from the input vector $\mathbf{I}$:

$$k = \text{argmin}_i \, \text{Distance}(\mathbf{I}, \mathbf{T}_i)$$

This index k corresponds to the closest illness based on the entered symptoms, indicating the condition that best matches the user's reported symptoms.

2) Magnitude Calculation: For each target vector $\mathbf{T}_i$, the algorithm calculates its magnitude, which represents the length or size of the vector. This magnitude is computed using the Euclidean norm:

$$\text{Magnitude}(\mathbf{T}_i) = \sqrt{\sum_{j=1}^{m} \mathbf{T}_{ij}^2}$$

where $\mathbf{T}_{ij}$ represents the jth element of the ith target vector $\mathbf{T}_i$. The magnitude provides a measure of the overall severity profile of the illness represented by the target vector.

3) Confidence Calculation: The algorithm calculates a confidence score for the match based on the exponential decay of the distance between the input and the selected target vector. This decay is computed as:

$$\text{Decay} = 100 \times \left(e^{-\frac{\text{Distance}(\mathbf{I}, \mathbf{T}_k)}{\|\mathbf{T}_k\|^2}} \right)$$

where $\text{Distance}(\mathbf{I}, \mathbf{T}_k)$ is the Euclidean distance between the input vector $\mathbf{I}$ and the selected target vector $\mathbf{T}_k$, and $\|\mathbf{T}_k\|$

is the magnitude of $\mathbf{T}_k$. Higher distances result in lower confidence scores, as indicated by the exponential decay function. To ensure a minimum level of confidence even for less certain matches, a minimum confidence threshold of 10% is enforced. The algorithm then calculates the confidence score as follows:

$$\text{Confidence} = 100 - \text{Decay}$$

This adjusts the decay value to provide the required confidence score, with higher values indicating greater confidence in the match.

B. User Interface

The "HealthAssist" application is built using Flutter Dart, offering a seamless cross-platform experience. The user interface (UI) of "HealthAssist" is meticulously designed to offer a seamless and intuitive experience for users seeking symptom-based health assessments. The UI features a clean and modern design with a user-friendly layout that prioritizes ease of navigation and accessibility.

Upon opening the app, users are guided through an introductory sequence spanning four screens, acquainting them with the app's functionality. Subsequently, users encounter a series of screens for symptom input, each featuring a symptom and a slider for adjusting symptom severity. Navigation between screens is facilitated by back and next buttons positioned at the bottom of the interface, enabling users to traverse the process effortlessly.

As users progress through the symptom input phase, they provide information about their symptoms using the intuitive slider interface. Once all symptoms are entered, the app transitions to a diagnosis generation screen, signaling the completion of symptom input. Here, users await the app's analysis, which is displayed on the subsequent screen, presenting the diagnosis and a confidence score.

Consistency in navigation and interface design ensures a user-friendly experience throughout the "HealthAssist" app. The back and next buttons serve as familiar anchors, allowing users to revisit and adjust their inputs as needed.

In Figure 3, sample output diagnosis for the symptoms of Urinary Tract Infection (UTI) is displayed. The input includes a range of symptoms commonly associated with UTI, such as frequent urination, pain or burning sensation while urinating, and blood in the urine. This visualization aids in understanding the diagnostic process and the potential outcomes derived from the input symptom data.

(a) (b) (c) (d)

Fig. 1: Introductory Screens of the App

III. Conclusion and Future Work

In conclusion, "HealthAssist" represents a significant advancement in symptom-based health assessment, providing users with a convenient and accessible platform for self-assessment and health management. By leveraging the ubiquity of smartphones and incorporating an innovative algorithm for symptom analysis and diagnosis, HealthAssist bridges the gap between individuals and healthcare resources, ultimately enhancing overall health outcomes. The success of HealthAssist underscores the potential of mobile applications in revolutionizing healthcare delivery and empowering individuals to

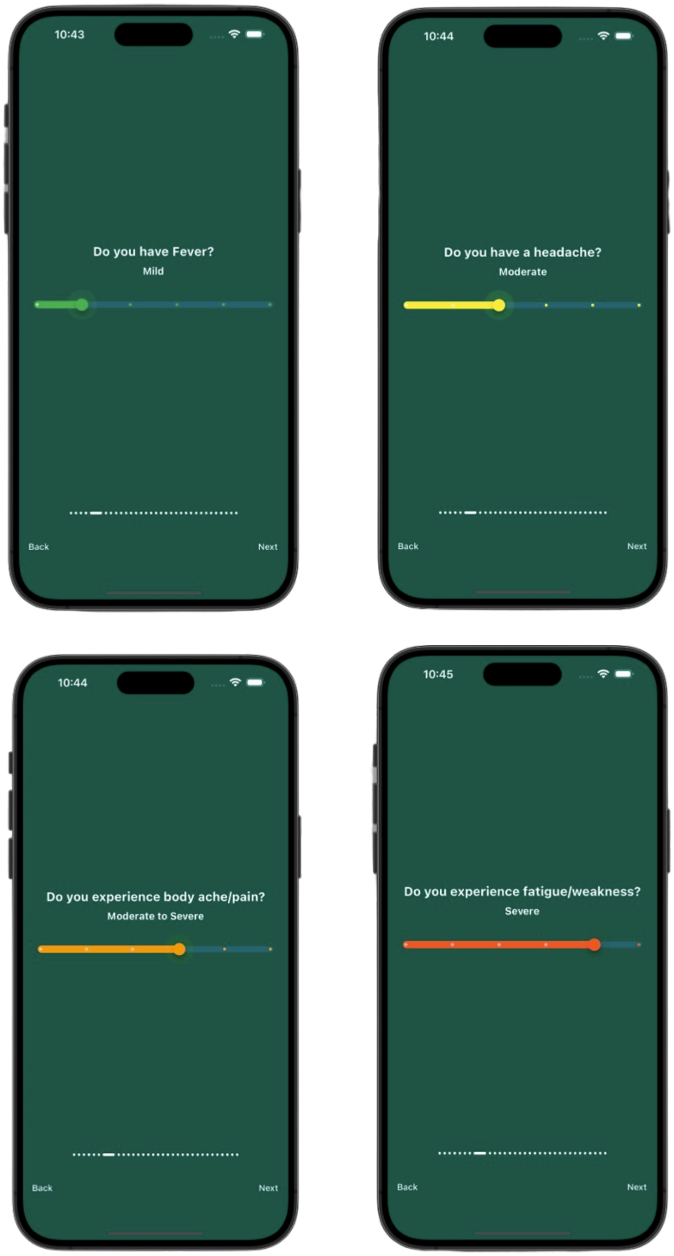

Fig. 2: Symptom Input Screens of the App

take control of their health. Particularly in rural communities with limited healthcare access, HealthAssist can prove to be a valuable tool for enabling remote health assessments and connecting users with essential healthcare information and resources. It's important to note that HealthAssist primarily focuses on the treatment of common illnesses and does not replace the need for medical tests or professional medical advice in cases requiring specialized diagnosis or treatment.

Future research and development efforts for HealthAssist could focus on several key areas to further enhance its functionality and impact. Focus will be on providing a report in a clear and concise format, with each potential illness

Fig. 3: Diagnosis display screens of the App

accompanied by a brief description and recommended next steps, such as seeking medical attention or self-care measures. Adding features for tracking symptoms over time, accessing past health assessments, and setting reminders for medication or appointments will also be incorporated.

Additionally, expanding the scope of the application to include a broader range of common illnesses and symptoms would improve its diagnostic accuracy and usefulness to users. Integrating machine learning techniques into the algorithm could enable HealthAssist to adapt and improve over time based on user feedback and real-world data. Finally, collaboration with healthcare professionals and institutions to validate the accuracy and effectiveness of HealthAssist would enhance its credibility and trustworthiness among users. Overall, continued innovation and refinement of HealthAssist hold the potential to significantly impact the field of digital health and improve healthcare accessibility and outcomes for individuals worldwide, especially in underserved rural communities.