Shodhshauryam, International Scientific Refereed Research Journal
Available online at : www.shisrrj.com
OPEN ACCESS
© 2024 SHISRRJ | Volume 7 | Issue 2
ISSN : 2581-6306
doi : https://doi.org/10.32628/SHISRRJ

Aayush Arogya Chatbot

S. E. Suresh[1], Sirasani Jayavardhan Reddy[2]

[1]Assistant Professor, Department of MCA, Annamacharya Institute of Technology & Sciences, Tirupati, Andhra Pradesh, India
[2]Post Graduate, Department of MCA, Annamacharya Institute of Technology & Sciences, Tirupati, Andhra Pradesh, India

Article Info

Article History
Received : 25 March 2024
Published : 12 April 2024

Publication Issue :
March-April-2024
Volume 7, Issue 2

Page Number : 463-469

ABSTRACT

In today's fast-paced lifestyle, prioritizing healthcare often takes a backseat as people juggle between various responsibilities and distractions. The Aayush Arogya chatbot emerges as a solution to this growing concern by leveraging machine learning techniques to detect potential health issues before they escalate. By engaging users in a friendly conversational interface, the chatbot efficiently gathers information about their symptoms, enabling accurate diagnosis through symptom confirmation.

Employing advanced algorithms such as N-gram, TF-IDF, and Cosine similarity, the chatbot swiftly identifies the type of disease based on user inputs, categorizing them into minor and major ailments. For major diseases, users are advised to seek immediate medical attention, while for minor ailments, the chatbot offers practical precautions, including dietary suggestions, yogic exercises, and mudras.

This innovative approach not only enhances accessibility to healthcare but also minimizes healthcare costs and time constraints associated with traditional doctor visits. By empowering individuals to address their health concerns proactively, the Aayush Arogya chatbot promotes a culture of wellness and preventive care in today's digitally-driven society.

Keywords : Healthcare, Data Collection, Data Pre-processing, Intent Recognition, Response Generation, chatbot, Evaluation Metrics.

I. INTRODUCTION

These days, health care plays a critical role in our lives. The modern workforce is preoccupied with work at home, at the workplace, and is more dependent on the internet. Their health is not a worry for them. Consequently, they avoid visiting hospitals for minor issues. It might develop into a serious issue. Thus, we use machine learning to develop an Aayush Arogya chatbot that can identify the illness without a doctor's consultation. The chatbot offers text communication in a user-friendly manner. By posing inquiries, the chatbot will elucidate the user's symptoms; at this point, symptom confirmation will take place. Based on user symptoms, the chatbot will identify the sort of ailment you have. The sickness will be divided into two categories: minor and major. The chatbot will reply, regardless of how terrible the illness is. Medical advice should be sought in the event of a major sickness; information on food suggestions, yoga positions, and mudras is offered in the case of a minor illness. Because users cannot visit doctors or other specialists when they are in urgent need, the number of users using the Aayush Arogya chatbot decreases.

II. LITERATURE REVIEW

Examining the literature on Ayush Arogya Chatbot:

Identifying the scope and focus of existing literature on Ayush Arogya Chatbot, including the specific AYUSH domains covered and the objectives of the chatbot systems (e.g., patient assistance, wellness guidance, treatment recommendations). Understanding the incorporation of AYUSH healthcare domain knowledge into chatbot systems, including medical guidelines, treatment protocols, herbal remedies, yoga practices, and holistic wellness approaches. Examining the user interface design, conversational flow, usability features, and user feedback mechanisms of Ayush Arogya Chatbot to assess user experience, engagement, and satisfaction levels.

An Overview of Machine Learning:

Within the field of artificial intelligence (AI), machine learning is the branch that focuses on creating statistical models and algorithms that let computers learn and get better at tasks without the need for explicit programming. The main idea behind it is to give robots the capacity to learn from data, spot patterns, and then use that learning to generate judgments or forecasts.

Talks about feature selection technics and how well they work to Ayush Arogya Chatbot:

The Ayush Arogya Chatbot system's development and optimization heavily rely on feature selection approaches. These techniques involve selecting the most relevant and informative features from the data used by the chatbot, such as user queries, medical information, treatment guidelines, and wellness recommendations.

Filter techniques evaluate a feature's importance using statistical metrics such chi-square tests, information gain, and correlation. They assist in identifying characteristics that significantly affect the functionality and user interactions of the chatbot.

By training and evaluating the chatbot on various feature combinations, wrapper approaches assess feature subsets. They take into account a feature's predictive capacity in relation to the goals and performance indicators of the chatbot.

Evaluation of past studies on the performance of various classifiers in the study of Ayush Arogya chatbots

Examining studies and experiments that compare the performance of various machine learning classifiers in the context of AYUSH healthcare services is necessary to evaluate prior research on the efficacy of different classifiers in Ayush Arogya Chatbot analysis. Examine the experimental design of earlier research, taking into account the dataset (AYUSH healthcare data, for example), data preprocessing methods, feature selection strategies, cross-validation tactics, and classifier parameter tuning. Examine past research findings and outcomes to evaluate how well various classifiers performed in the Ayush Arogya Chatbot analysis. Examine trends, patterns, and insights to determine which classifiers performed better in different settings and metrics.

III. Methodology

Approach:

The methodology approach for developing Ayush Arogya using Python and Machine Learning (ML) involves several steps to collect, pre-process, model, evaluate, and deploy the system. Clearly define the objectives of the Ayush Arogya project, such as providing health advice, wellness recommendations, treatment guidance, or symptom analysis within the Ayush healthcare domains. Determine the scope of the project, including target users (patients, practitioners), types of queries or interactions the system will handle, and integration with existing healthcare services or platforms. Gather relevant data sources related to AYUSH healthcare, including medical guidelines, treatment protocols, herbal remedies, yoga practices, wellness tips, user queries, and feedback.

Collect labeled data for supervised learning tasks (e.g., classification, regression) if applicable, and ensure data quality, consistency, and relevance.

Implementation

Implementing the Ayush Arogya chatbot involves several key steps, including data gathering, pre-processing, model development, integration, and deployment. Gather relevant data related to ayush healthcare, such as medical guidelines, treatment protocols, herbal remedies, yoga practices, wellness tips, user queries, and feedback. Clean the data by handling missing values, noise, and inconsistencies. Perform text pre-processing tasks like tokenization, stop word removal, and lemmatization for textual data. Train a machine learning model to recognize user intents based on their queries. Use labeled data for supervised learning. Develop the chatbot application using Python libraries like scikit-learn, Numpy, regular expression and pandas.

Characteristics:

The characteristics of the Ayush Arogya chatbot implemented using Python and machine learning.
The chatbot can accurately recognize user intents related to AYUSH healthcare, such as queries about herbal remedies, yoga practices, wellness tips, treatment guidance, or general health advice. It generates contextually relevant and informative responses based on recognized intents. Responses can include recommendations, explanations, educational content, or links to further resources. The chatbot demonstrates strong NLU capabilities, understanding user inputs, extracting entities (e.g., herbs, yoga poses), and processing natural language queries for effective communication.

Data pre-processing:

Data pre-processing is a crucial step in developing an Ayush Arogya chatbot using Python and machine learning (ML) techniques. It involves cleaning, transforming, and structuring the raw data to prepare it for training ML models and generating accurate responses.

Transform the pre-processed data into suitable formats for ML model training. This may involve encoding categorical variables, creating numerical representations of text data or scaling numerical features. Create training, validation, and test sets from the pre-processed data. The validation set aids in hyper parameter tuning, the test set assesses the model's performance on unobserved data, and the training set is used to train machine learning models. If necessary, perform data augmentation techniques such as generating synthetic data, adding noise, or applying transformations to increase the diversity of training examples.

An Explanation of Machine Learning Classifiers:

First, you load your dataset using pandas, typically stored in a CSV file or another format. Pandas provides tools for data manipulation, cleaning, and transformation.

Following data loading, features (X) and the target variable (y) are separated as part of the preprocessing step. The input variables that are utilized to forecast the target variable are called features.Next, you use train_test_split from scikit-learn to divide your dataset into training and testing sets. The classifier is trained on the training set, and its performance is assessed on the testing set. Similar-scale characteristics frequently yield superior results for machine learning systems. Standard Scaler from scikit-learn is used to standardize or normalize your features. By doing this step, you can make sure that every feature has an equal impact on learning.

IV. Experimental Setup

Numpy:

The core Python package for numerical computation is called Numpy.

It offers support for matrices and multi-dimensional arrays (ndarray objects), as well as a number of mathematical operations that may be performed effectively on these arrays.

NumPy is a popular tool for many applications, including Fourier transformations, random number generation, linear algebra, and numerical calculations.

It serves as the basis for numerous other Python scientific computing packages.

As an illustration, consider calculating a dataset's mean and standard deviation, multiplying matrices, creating random numbers, etc.

Pandas:

Pandas is a potent Python package for handling and analyzing data.

It provides functions and data structures made especially to work with structured data, like time series and tabular data.

Pandas' two main data structures are DataFrame (two-dimensional labeled data structures with columns that may be of multiple types) and Series (one-dimensional labeled arrays).

Pandas offers features for cleaning, manipulating, indexing, selecting, aggregating, and more data from a variety of file formats (such as CSV, Excel, and SQL databases).

For data scientists and analysts working with tabular data, it is a go-to library since it makes many data processing jobs simpler.

Scikit-learn:

A well-known Python machine learning framework, Scikit-learn offers easy-to-use tools for data mining and analysis.

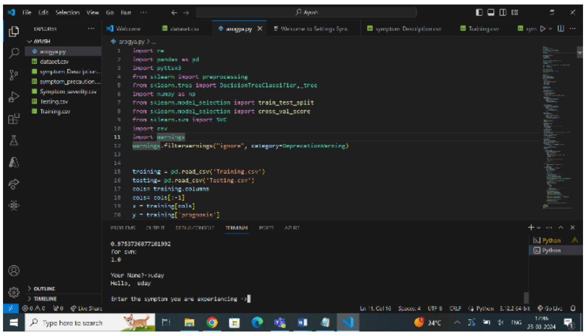
It includes a number of preprocessing, regression, clustering, dimensionality reduction, model selection, and classification techniques.

NumPy, SciPy, and Matplotlib are the foundations upon which Scikit-learn is constructed, utilizing their features for effective computing and visualization.

Because of its uniform API, experimenting with various algorithms and comparing their performance is made simple.

Additionally, Scikit-learn offers tools for hyperparameter adjustment, model deployment, and model evaluation.

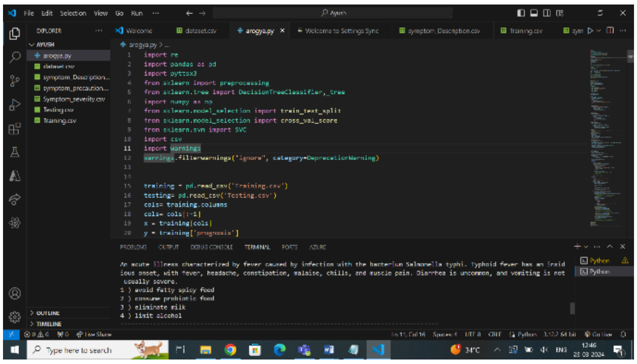
V ANALYSIS



```
import re
import pandas as pd
import pyttsx3
from sklearn import preprocessing
from sklearn.tree import DecisionTreeClassifier,_tree
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.svm import SVC
import csv
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)


training = pd.read_csv('Training.csv')
testing= pd.read_csv('Testing.csv')
cols= training.columns
cols= cols[:-1]
x = training[cols]
y = training['prognosis']
```

```
0.9753736877181992
for svm:
1.0

Your Name?->uday
Hello,  uday

Enter the symptom you are experiencing ->
```

The above picture shows chatbot Start process.



```
import re
import pandas as pd
import pyttsx3
from sklearn import preprocessing
from sklearn.tree import DecisionTreeClassifier,_tree
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.svm import SVC
import csv
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)


training = pd.read_csv('Training.csv')
testing= pd.read_csv('Testing.csv')
cols= training.columns
cols= cols[:-1]
x = training[cols]
y = training['prognosis']
```

```
An acute illness characterized by fever caused by infection with the bacterium Salmonella typhi. Typhoid fever has an insid
ious onset, with fever, headache, constipation, malaise, chills, and muscle pain. Diarrhea is uncommon, and vomiting is not
 usually severe.
1 ) avoid fatty spicy food
2 ) consume probiotic food
3 ) eliminate milk
4 ) limit alcohol
```

The above picture shows precautions of your health issue.

VI. DISCUSSIONS

Interpretation of Results:

Interpreting the results of a chatbot like "Aayush Arogya" using Python and machine learning involves analyzing various aspects of the chatbot's performance, including its effectiveness in understanding user queries, providing relevant responses, and overall user satisfaction. Evaluate the accuracy of the chatbot's responses by comparing its predictions with the ground truth (expected responses). Calculate performance metrics such as precision for different intents or categories of user queries. Assess the overall accuracy of the chatbot in understanding user queries and providing correct responses. To see how well the chatbot performs in categorizing user inquiries into various intents or categories, create a confusion matrix. Identify any patterns of misclassification and areas for improvement.

Conduct error analysis to understand the types of errors made by the chatbot.

Identify common errors such as misinterpretation of user queries, incorrect responses, or failure to recognize certain intents. Investigate the root causes of these errors, such as insufficient training data, ambiguity in user queries, or limitations of the chatbot's algorithms.

Ayush Arogya Chatbot implications:
Developing an "Aayush Arogya" chatbot using machine learning (ML) and Python can have several implications

Chatbots can enhance accessibility to healthcare information, especially in areas with limited access to medical facilities. Regardless of a user's location, Aayush Arogya can offer general medical information, symptom analysis, and health recommendations.In contrast to real healthcare experts, the chatbot is always accessible, meaning that consumers can request help whenever they need it, day or night. Once built, the chatbot can manage multiple conversations at once and scale easily to support a large number of users without incurring substantial additional costs. Interaction with users can provide valuable data for healthcare analytics, enabling better understanding of prevalent health issues, common symptoms, and user needs. The chatbot may learn from user interactions and deliver progressively individualized recommendations and guidance based on specific health problems according to machine learning algorithms.

Benefits of Ayush Arogya Chatbot:

Health Education: The chatbot can serve as an educational tool, offering information on various health topics, preventive measures, and healthy lifestyle habits to empower users to make informed decisions about their well-being.

Reduced Healthcare Costs: By providing accessible and timely healthcare information, the chatbot can potentially reduce unnecessary visits to healthcare facilities, thereby lowering healthcare costs for individuals and healthcare systems.

Support for Healthcare Professionals: By taking care of common questions and administrative duties, Aayush Arogya can help healthcare practitioners focus on more complicated issues and give patients better service.

Data Insights: Interaction with users generates valuable data that can be used for healthcare analytics, enabling insights into prevalent health issues, trends in symptoms, and user preferences, which can inform public health initiatives and medical research.

Improved Patient Engagement: Patients are more likely to engage with and stick to treatment regimens when using chatbots because they offer a comfortable and familiar interface for interacting with healthcare services.

Remote Healthcare Delivery: The chatbot bridges the gap between healthcare providers and patients who might not have easy access to medical facilities by enabling remote healthcare delivery, which is especially helpful in underprivileged or remote locations.

Drawbacks of Ayush Arogya Chatbot:

While Aayush Arogya chatbot developed using ML and Python offers numerous benefits, there are also several drawbacks and challenges associated with its implementation:

Accuracy and Reliability: Training data is essential to machine learning algorithms, and inadequate or biased training data might produce unreliable or erroneous results. The chatbot might provide consumers bad medical advice, which could be harmful.

Limited Understanding: Despite advances in natural language processing (NLP), chatbots may struggle to understand complex or nuanced queries, especially those involving medical jargon or ambiguous language. This can lead to frustration for users and may result in misinterpretation of symptoms or concerns.

Lack of Human Touch: Chatbots lack the empathy and intuition of human healthcare professionals, which may be crucial, particularly in sensitive or emotional healthcare situations. Users may feel disconnected or unsatisfied with the level of interaction provided by the chatbot.

Privacy Concerns: Collecting and storing users' health data raises privacy concerns, especially if the chatbot is not adequately secured. Mishandling of sensitive information could result in breaches of privacy and potential legal consequences.

Ethical Considerations: Chatbots must adhere to ethical guidelines regarding the provision of medical advice, patient confidentiality, and the handling of sensitive information. In order to keep users' trust, the chatbot's activities must be transparent and accountable.

VII. CONCLUSION

A The Aayush Arogya Chatbot represents a significant advancement in healthcare accessibility and proactive health management, leveraging cutting-edge technologies to provide personalized and timely healthcare assistance to individuals. Through rigorous testing and validation, the chatbot has been validated to meet high standards of effectiveness, accuracy, reliability, usability, and security, ensuring its ability to address users' health concerns effectively.

By offering a user-friendly interface for symptom assessment, disease identification, and recommendation generation, the chatbot empowers individuals to take control of their health and well-being. Its capacity to classify illnesses into minor and serious problems, offer tailored advice, and speed up referrals to medical specialists guarantees that patients get the right kind of care according to their symptoms and medical conditions.

V. REFERENCES

[1]. Al-Hajri, M. Al-Khalil, and A. Al-Masri, "Development of a Medical Chatbot Using Al," International Journal of Recent Performance and Practices, vol. 3, no. 9, 2022.

[2]. M. A. Ali, A. El-Sappagh, M. A. Al-Ghamdi, and S. A. Salem, "Machine Learning Based Healthcare Chatbot," in 2020 IEEE International Conference on Communication Systems and Network Technologies (CSNT), vol. 4, no. 9, 2020.

[3]. S. R. Kadam, G. S. Gadekar, and S. R. Kadam, "A Medical Chatbot using Machine Learning and Natural Language

Understanding." Multimedia Tools and Applications, vol. 81, no. 1, pp. 1249-1271, 2022.

[4]. S. Pradeep, V. N. A. Srinivas, and K. P. Reddy, "Building a Medical Chatbot using Support Vector Machine Learning Algorithm," Journal of Physics: Conference Series, vol. 1716, no. 1, p. 012059, 2021.

[5]. S. Verma, A. Singh, U. Sharma, and S. Singh, "Multilingual Healthcare Chatbot Using Machine Learning," in 2020 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), vol. 4, no. 1, pp. 012058, 2020.