D-PREDICT: DISEASE PREDICTION USING GAUSSIAN NAIVE BAYES ALGORITHM

Jasna P. S.[1], Aiswarya M. K. [2] , Krishnapriya K.V. [3], Sisira P.C. [4], Jayasree N. Vettath. [5]
Department of Computer Science and Engineering
Sreepathy Institute of Management And Technology, Vavanoor, Kerala, India
[1]jasnaps1998@gmail.com, [2]aiswaryamk20@gmail.com, [3]kpriyakv1999@gmail.com, [4]sisira2219@gmail.com, [5]jayasree.n.vettath@simat.ac.in

Abstract—The application of machine learning in the field of medical diagnosis is increasing gradually. This can be contributed primarily to the improvement in the classification and recognition systems used in disease diagnosis which is able to provide data that aids medical experts in early detection of fatal diseases and therefore, increase the survival rate of patients significantly. The Existing systems for disease prediction are narrowed down one or a few diseases or medical conditions that are very crucial in the healthcare field. So the objective of the D-predict is that it will predict the diseases that are seen in old aged people using the concept of natural language processing and machine learning. According to the D-predict when user give their symptoms as input in the form of speech, after NLP processing the key features are extracted and using Naive Bayes classifier the disease is predicted. The system can diagnose and predict disease efficiently at an earlier stage with reasonable accuracy.

Index Terms — Data analysis, disease prediction, health care, machine learning with naive bayes algorithm.

I. INTRODUCTION

At present, when one suffers from particular disease, then the person has to visit to doctor which is time consuming and costly too. Also if the user is out of reach of doctor and hospitals it may be difficult for the user as the disease can not be identified. So, if the above process can be completed using an automated program which can save time as well as money, it could be easier to the patient which can make the process easier [1].

The disease is an abnormal condition of parts of the body or mind of humans. To find out, identify a type of disease or health problem experienced by the patient is to make a diagnosis. The diagnosis process aims to find out or identify a type of disease so that efforts can be made to immediately control the diseases, it is also an effort to prevent and overcome the spread of diseases. One way to diagnose the disease is the Anamnesis technique, which is to do the question and answer directly or indirectly between patients and health workers who can diagnose disease. Anamnesis is divided into two types, the first is Auto History, which is a question and answer process that is aimed directly at the patient or who has the disease. To be able to make Auto History, the patient is conscious, mature and communicative (ability to communicate well). The second type is Allo anamnesis, which is a question and answer process that is carried out between health workers with family or relatives of patients, usually, this process is carried out when the patient is not communicative and the patient experiences impaired memory. So it is not possible to do a question and answer process [4].

D-Predict is a web based application that predicts the disease of the user with respect to the symptoms given by the user. D-predict system has data sets collected from different health related sites. With the help of Disease Predictor the user will be able to know the probability of the disease with the given symptoms. As the use of internet is growing

every day, people are always curious to know different new things. People always try to refer to the internet if any problem arises. People have access to internet than hospitals and doctors. People do not have immediate option when they suffer with particular disease. So, this system can be helpful to the people as they have access to internet 24 hours.

II. LITERATURE SURVEY

A. Speech-To-Text Conversion

Real time speech to text conversion system converts the spoken words into text exactly the same way that user pronounces [9]. Speech recognition is the process of extracting the attributes of speech and classifying the same attribute with prerecorded datasets. To recognize a word, audio signals are tested and framed to phonemes. For this audio record are convert into wave format. Spectrum based parameters are obtained when a record is recognized. To find most informative parameters of speech signal Mel Frequency Cepstrum Coefficient (MFCC) technique is used. By using this technique new spectrum is obtained that is different from previous spectrum of spoken words.

Kalman Filter is a state estimator that produces an optimal estimate and minimizes the mean square error. Kalman filtering is an effective approach to remove non stationary noise form background or otherwise. It enhance the ability of this Real time speech recognition system. It is computationally simple and more robust to noise than HMM based SRS.

Speech Recognition Process: The process of speech recognition is divided into two modules: the first module is the training stage, in which a database is created by recording the user's speech samples in .wav format in MATLAB. Then the system is trained. The second module is the testing stage. Fig 1 shows the speech recognition process.

Feature Extraction: MFCC feature extraction technique is used to extract features of speech signal. MFCC features are based on the human ear perception that means human's ear's critical bandwidth frequencies filters the spaced linearity between the high frequency and low frequency of each word uttered by the user. The human understanding for different frequency ranges of the uttered word is shown on a nonlinear scale. Pitch period of every word is measured with a Mel scale. Fig 2 shows the dataflow of the MFCC feature extraction process.

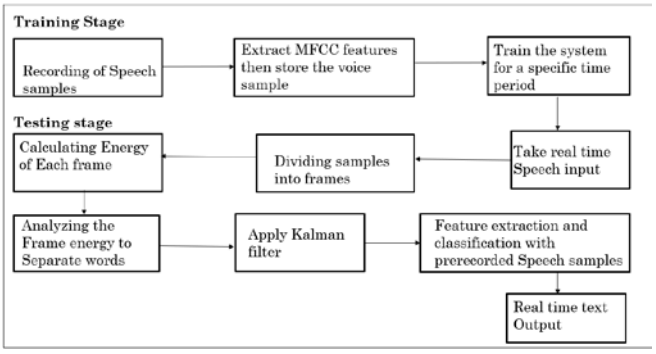

Fig. 1. Speech Recognition Process [9]

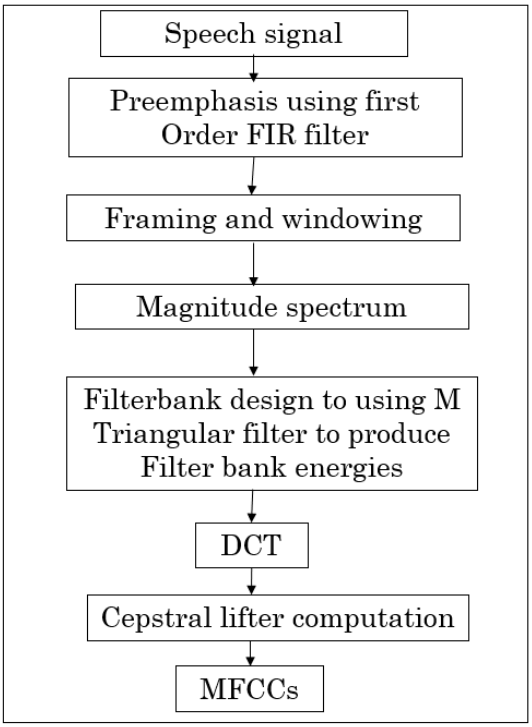

Fig. 2. MFCC Feature Extraction [9]

After the training of this system, a real time speech input was given to it through a good quality microphone. The system divided the real time speech sample into small segments of frames or continuous groups of samples. After that the energy of each frame segment was calculated. Equation 1 calculates the energy using simple energy formula [9].

$$Ex = \int_{-\infty}^{\infty} x^2 d \qquad (1)$$

Energy calculated was then analyzed by a speech detection algorithm to separate the words. The speech disclosure algorithm is applied to detect each word by processing the stored speech samples in database (derived or self-created) frame by frame. For the detection of each frame, uses a combination of signal energy and a zero crossing rate. Create an acoustical model for the detection of each uttered word. Different sound has different frequencies. To predict the different frequencies power spectral density measure is used. The output speech signal was compared with the prerecorded clean speech signal with the correlation figure. Equation 2 is used to find the correlation figure[9].

$$\rho(w_o, w_r) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} w_{oij} * w_{rij}}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N} w_{oij}^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N} w_{rij}^2}} \quad (2)$$

Kalman Filtering: After the process of word separation Kalman filter removes the unwanted noise and gives the filtered output. The Fig 3 represents the proposed bidirectional Kalman filter algorithm.

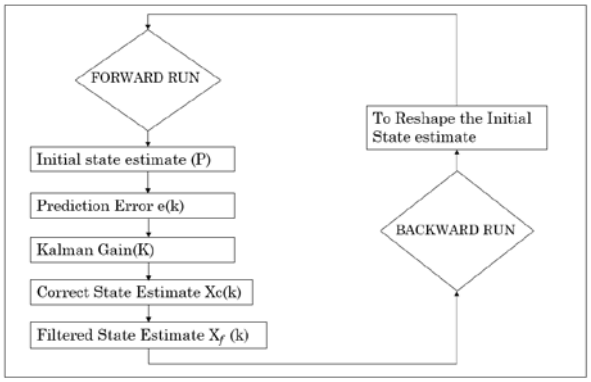

Fig. 3. Bidirectional Non stationary Kalman Filter Loop Operation [9]

The calculation in loop as follows:

1. *Real Time Speech signal is represented in Equation 3:*

$$Y(k) = s(k) + v(k) \quad (3)$$

2. *Predicted Error is calculated using Equation 4:*

$$e(k) = Y(k) + Yp(k) \quad (4)$$

3. *Kalman Gain can be calculated using Equation 5*

$$K(k) = \frac{Pp(k)C^T}{CPp(k)C^T + R} \quad (5)$$

4. *Correct state estimate calculated using the Equation 6:*

$$Xc(n) = xp(n + Ke(n) \quad (6)$$

5. *The filtered state estimate Xf (k) is equal to the correct state Estimate*

Where Y (k) is real time speech signal with noise and Yp(k) is predicted measurement variable(assumed). C is the measurement gain matrix, Pp(k) is the Auto covariance matrix of the predicted state variable. R is the auto covariance matrix of the measurement noise. xp(k) is the predicted state estimate.

B. *Natural language Processing (NLP)*

NLP is a field of computer science that deals with the interaction between computers and humans, such as Indonesian or English [3]. NLP is used to process writing in order to understand what is said by humans. The main purpose of NLP is to make machines or computers to understand the meaning of human language so that they can provide appropriate responses. NLP application in the medical field is very important as Clinical Decision Support (CDS) which helps health professionals make clinical decisions, deal with medical data about patients or with the knowledge of drugs needed to interpret the data. In NLP, several processes were included such as Tokenizing, Stemming and Stop words removal. Fig 4 represents the workflow diagram.

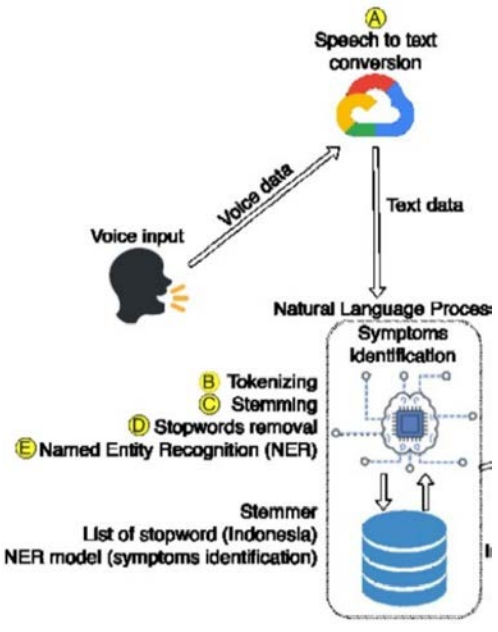

Fig. 4. Workflow Diagram [3]

Components of Workflow: *Voice Recognition*: The process in this system starts with recording the patient's voice. The recorded voice is the sound of patient complaining about symptoms of malaria disease experienced. The audio file used has a bitrate specification of 16000. As a sample, sound recording is done three times with different complaints shown in Table I.

TABLE I
VOICE RECORD DATA

No.	Speech Input
1	I have fever, headache and vomiting
2	I feels hot and shivering
3	I have pain on my muscles and have diarrhoea

NLP is a technique that processes data in the form of text, so that input in the form of sound in this study needs to be converted into text. This conversion makes the whole input into one sentence is shown in Table II.

TABLE II
SPEECH TO TEXT CONVERSION

No.	Speech To Text Conversion
1	I have fever, headache and vomiting
2	I feels hot and shivering
3	I have pain on my muscles and have diarrhoea

Tokenizing: In NLP, tokenizing is the first processes in NLP that identify basic tokens or units for the next process. In simple terms, tokenizing breaks down large text data into smaller shapes to facilitate the analysis process, such as paragraphs being sentences, or sentences being words. In this study, we do not break paragraphs into sentences but sentences into words. The results of speech to text conversion are in the form of a long sentence is shown in Table III.

TABLE III
WORD TOKENIZING

No.	Word Tokenizing
1	['I', 'have', 'fever', 'headache', 'and', 'vomiting']
2	['I', 'feels', 'hot', 'and', 'shivering']
3	['I', 'have', 'pain', 'on', 'my', 'muscles', 'and', 'have', 'diarrhoea']

Stemming: Stemming works by transforming words into their basic forms. The basic form is not always the same as the root word is shown in Table IV.

TABLE IV
STEMMING RESULT

No.	Stemming Results
1	['I', 'have', 'fever', 'headache', 'and', 'vomit']
2	['I', 'feel', 'hot', 'and', 'shiver']
3	['I', 'have', 'pain', 'on', 'my', 'muscle', 'and', 'have', 'diarrhoea']

The Nazief & Adriani (NA) algorithm used in the stemming process in this research resulted in an overall accuracy score of 95.9 percentage using 30 data, which is included in a more accurate algorithm than other stemming algorithms because it has used a basic word dictionary as a reference main.

Stop words removal: Almost all NLP implementations in the Machine Learning field use the stop words removal method. This method works by removing several conjunctions but does not affect the overall content. Stop words removal is used to improve system performance in order to effectively process the data needed is shown in Table V.

TABLE V
STOPWORD REMOVAL RESULT

No.	Stopwords Removal
1	['fever', 'headache', 'vomit']
2	['feel', 'hot', 'shiver']
3	[pain', 'muscle', 'diarrhoea']

In this research with the aim of identifying symptoms of a disease that is formed from one or several words, there are several symptoms of a disease that requires conjunctions (stop words). The stop words list in this final project is static, which means that the stop list is obtained from the agency, institution, or developer of the stop word provider for the filtering process used. Stop word static contains common words in the same language so that they can be applied to

other NLP projects, but because of their general nature, they make accuracy in the filtering process less.

The dataset is used 1621 lines consisting of 100 sentences that have been separated by the word and labeled. The training process is carried out with iterations with 800 epochs variables. The training results produced a micro-f1 score of 0.79. To carry out testing of the models that have been carried out by training, an analysis is carried out by providing input on the complaints of the disease.

Several factors that can affect the accuracy of the identification process include the NER method used and also the model generated from the training dataset. The method used in the NER process in this final project is Bidirectional LSTM-CRF. The next factor of accuracy is the model produced from the training dataset that can still be improved by increasing the size and variety of the corpus used in the training dataset to produce the model.

C. Naive Bayes classifier

The general day to day health of a person is vital for the efficient functioning of the human body [10]. Taking certain prominent symptoms and their diseases to build a Machine learning model to predict common diseases based on real symptoms is the objective of this research. With the dataset of the most commonly exhibited diseases, we built a relation to predicting the possible disease based on the input of symptoms.

1. Text Processing: For the initial step, consider the symptoms data and perform count-vectorization. This is done to sort the words in the corpus into a bag-of-words. The model is simple in that it ignores the order of words and relations rather focus on the occurrence of words the dataset.

Naive Bayes use the tf-idf parameter which helps in keyword extraction to help in faster computation. The Term Frequency and Inverse Document Frequency play a major role in prediction and understanding the dispersion of symptoms. This shows how the dataset has keywords which vary in occurrences based on a certain disease.

Next, use the above logic to perform bigram words classification of data. For example, consider ‘abdominal pain’ as one word and check similar occurrences in dataset. This is sorted into a count vector matrix of binary matrix. For the analysis for this data we use Term Frequency-Inverse Data Frequency (tf-idf) function which gives the frequency of the word in each document in the corpus [6]. It is the ratio of number of times the word appears compared to the total number of words present in the dataset. Using Equation 7 and Equation 8 can get the required tf-idf mappings [6].

$$TF(t) = \frac{\text{(Number of word occurrences)}}{\text{(Total number of words in the dataset)}} \quad (7)$$

$$IDF(t) = \log_e \frac{\text{(Total number of documents)}}{\text{(Number of documents with term t in it)}} \quad (8)$$

2. Naive Bayes: Naive Bayes is a conditional probability based algorithm. It is the most widely used and fastest algorithm since it uses less training data and strong independence assumptions. In our case, we use a built in function called Multinomial Naïve Bayes which is mainly used for discrete features such as text classification. It requires a feature count parameter which helps in determining each class while fitting the sample with appropriate weights. For this we use the tf-idf count vector as the parameter. This experiment yielded around 99% accuracy. In text classification, the main aim is to find the best class for the given document. Algorithm .1 shows how text classification can be combined as a parameter to pass tf-idf vector which contains tokens and frequency data. We score the algorithm to obtain the results. It has 2 main functions. The first function is used to train the multinomial Naive Bayes model based on the feature extraction and count vector. Each of the Count (C) of words and Document (D) can be done in a single pass through this training data. The conditional probability of Vector in each case is returned. The next step is to apply the algorithm to the tf-idf count vector (V) to assign score to each term collected as bag-of-words. The final score

is obtained as the cumulative score for given document. Prabakaran et al. [8] states that research shows a web system using Na¨ıve Bayes to provide answers complex queries to diagnose heart disease.

Most commonly used methods to evaluate the classification methods accuracy are Leave-One Out and Cross-Validation. The comparison between algorithms in [2] shows the use case scenarios for each algorithm with merits and demerits. An approach of sequential pattern mining is very feasible for the continuously emerging characteristics of stream data suggested in [7].

Algorithm.1 Algorithm used for combining text classification with Multinomial Naive Bayes [10]

```
TRAINMULTINOMIALNB(C, D)
 1  V ← EXTRACTVOCABULARY(D)
 2  N ← COUNTDOCS(D)
 3  for each c ∈ C
 4  do N_c ← COUNTDOCSINCLASS(D, c)
 5     prior[c] ← N_c / N
 6     text_c ← CONCATENATETEXTOFALLDOCSINCLASS(D, c)
 7     for each t ∈ V
 8     do T_ct ← COUNTTOKENSOFTERM(text_c, t)
 9     for each t ∈ V
10     do condprob[t][c] ← (T_ct+1) / Σ_t'(T_ct'+1)
11  return V, prior, condprob

APPLYMULTINOMIALNB(C, V, prior, condprob, d)
1  W ← EXTRACTTOKENSFROMDOC(V, d)
2  for each c ∈ C
3  do score[c] ← log prior[c]
4     for each t ∈ W
5     do score[c] += log condprob[t][c]
6  return arg max_c∈C score[c]
```

III. PROPOSED SYSTEM

A. System Architecture

The proposed system D-predict is a disease prediction system which will predict the disease based on the symptoms given by the user. The system architecture is shown in Fig 5. In the proposed system the user can give their symptoms as voice input. After recognizing the voice data, it is converted into text data using

JavaScript Speech API. In the next step NLP techniques are used to process these text data and as a result Symptom grade level pair is generated. In the final step disease prediction is done using Gaussian Naive bayes classifier and the system will display the result as disease according to the symptoms provided by the user.

The proposed system consists of mainly 4 modules

1) Dataset Description
2) Speech Recognition
3) Natural Language Processing
4) Naive Bayes Classification

B. Proposed System Outline

Proposed system consists of four phases. The first one is the Dataset Preparation. In this phase we split the dataset into train and test for choosing the model. And here we set the index column as prognosis.

Second phase is the Speech recognition phase where the symptoms are obtained from a user using microphone. It is converted to text using Speech Recognition module. JavaScript API is used for the Speech recognition process

The third phase is Natural language processing. NLTK library is used for this. It consists of mainly two processes. Stop word removal and tokenization. Stop word removal is done for removing meaningless words. Tokenization is performed for converting sentence into Tokens. At the end of this module symptoms and their grade level pair will be obtained.

The Fourth and Final phase is Prediction Module. Here we use a built-in function called Gaussian Naives Bayes Classifier. In our system, for implementing Naive bayes algorithm we use python library which is named as sklearn. And Gaussian NB class have function like fit() which will build the training model whose inputs are independent and dependent values of dataset and other function like predict() function which takes input as testing values and then it can predict the disease of the user.

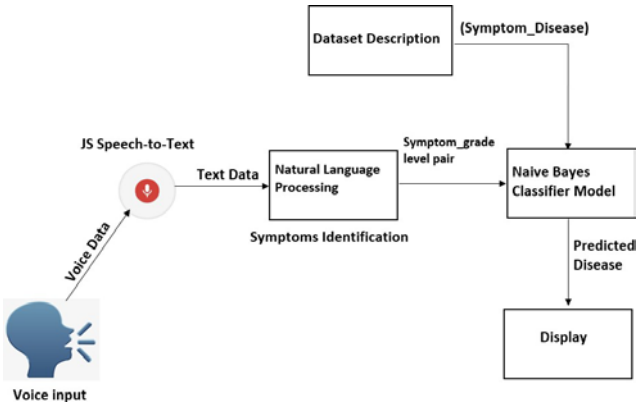

Fig. 5. System Architecture

C. *Dataset Description*

This module deals with the dataset description. For this, first of all, a dataset is required. The dataset is collected from kaggle site and is consist of 132 symptoms and 41 diseases.

First, the training data will be read and then preprocessed it using pandas. So, By printing it can get the whole training data. Then set index column as prognosis ie., as disease. Here, two variables 'x' and 'y' is used. In x, L1(list of symptoms) will be stored and In y, Prognosis (Disease) will be stored. Function np.ravel() is done in y which make the multidimensional value to continuous value ie., a flattened array. And while printing the training data we will get index column as disease. The same method is used in the testing data. After a model has been processed by using the training set, test the model by making predictions against the test set. Because the data in the testing set already contains known values for the attribute that you want to predict, it is easy to determine whether the model's guesses are correct [5].

D. *Speech Recognition*

In this module speech recognition is done using JavaScript API. It is super easy to recognize speech in a browser using JavaScript and then getting the text from the speech to use as user input. When the user input their symptoms in the form of speech, we will use the SpeechRecognition object to convert the speech into text and then display the text on the screen. We haven't used too many properties and are relying on the default values. In the HTML web page, there is a button to initiate the speech recognition and speechAPI.start() method is used to start the speech recognition [5].

Once we begin speech recognition, the on start event handler can be used to inform the user that speech recognition has started and they should speak into the microphone. When the user is done speaking, the on result event handler will have the result. The Speech Recognition Event results property returns a Speech Recognition Result List object. The Speech Recognition Result List object contains Speech Recognition Result objects. It has a getter so it can be accessed like an array. The first [0] returns the Speech Recognition Result at the last position. Each Speech Recognition Result object contains Speech Recognition Alternative objects that contain individual results. These also have getters so they can be accessed like arrays. The second [0] returns the Speech Recognition Alternative at position 0. We then return the transcript property of the Speech Recognition Alternative object. We have used Speech API. stop() method of the Speech Recognition object to stop the recognition process.

E. *Natural Language Processing*

The NLP tasks are achieved by coding with Python's 'nltk' library. The process involves two major preprocessing methods. That are Stop word removal and Tokenization.

1) Stop Word Removal: Stop word removal helps to get rid of the meaningless words in dataset. For example, filler words like 'of' or 'to' hold no meaning and are often not important. In addition, stop word removal will also reduce the dimensional space and provide a good advantage. The important thing is that, words that are not in the stop word list are considered as the tokens.

2) Tokenization: Tokenization is a straight forward process which helps in converting sentences into tokens. When user give their symptoms in the form of speech it will be in the sentences. So we need to tokenize the sentence to extract the relevant keywords. During tokenization the words other than stopwords are converted to tokens. Using the NLTK tokenization method, the sentences are spilt into tokens, making it easier to use for the clustering and classification algorithms.

F. Naive Bayes Classification

In this module the prediction of disease is done. That is predict the disease of the user based on the symptoms given by him/her. The aim of this project is disease prediction and is done by Gaussian Naive Bayes classifier.

A Gaussian Naive Bayes algorithm is a special type of NB algorithm. It's specifically used when the features have continuous values. This is simple as calculating the mean and standard deviation values of each input variable (x) for each class value. When all the data values of any particular dataset are numeric, then Gaussian Naive Bayes is used. It follows a normal distribution. Mean, and standard deviation are used to define the probability density function. It calculates the mean and standard deviation for each attribute of the dataset. After calculating this, when any test data pattern comes, then by using the mean and standard deviation calculate the probabilities for each test data. It assigns a class label to the test data which probability is close to 1. The mean can be calculated using the formula shown in Equation 9.

$$mean(x) = 1/n * sum(x) \quad (9)$$

Where n is the number of instances and x are the values for an input variable in your training data. We can calculate the standard deviation using the Equation 10.

$$standard\ deviation(x) = sqrt(1/n * sum(xi - mean(x)^2)) \quad (10)$$

This is the square root of the average squared difference of each value of x from the mean value of x, where n is the number of instances, sqrt() is the square root function, sum() is the sum function, xi is a specific value of the x variable for the i^{th} instance and mean(x)2 is the square of Mean(x) which is described in Equation 9.Probabilities of new x values are calculated using Gaussian Probability Density Function (PDF). When making predictions these parameters can be plugged into the Gaussian PDF with a new input for the variable, and in return the Gaussian PDF will provide an estimate of the probability of that new input value for that class. It is shown in Equation 11.

$$pdf(x, mean, sd) = \frac{1}{sqrt(2 * PI) * sd)} * exp(-\frac{(x - mean^2)}{(2 * sd^2)}) \quad (11)$$

Where pdf (x) is the Gaussian PDF, sqrt() is the square root, mean and sd are the mean and standard deviation calculated in equations 9 and 10 respectively, PI is the numerical constant, exp() is the numerical constant e or Euler's number raised to power and x is the input value for the input variable. We can then plug in the probabilities into the equation above to make predictions with real- valued inputs.

IV. RESULT

The proposed system is developed to predict general disease seen in old aged people in earlier stages as we all know in competitive environment of economic development the mankind has involved so much that he/she is not concerned about health. According to researches there are 40% peoples who ignores about general disease which leads to harmful disease later.

Fig. 6. Home Page

The proposed system "D-predict" is implemented using python & JavaScript. The interface of this project is done using html and CSS. The Home page of D-predict is shown in Fig 6.Here first the user needs to select the "SYMPTOM CHECKER" for input their symptoms, that is shown in Fig 7. In SYMPTOM CHECKER there is a "Start Recording" button and "Submit" button. In D-predict the user has to give their symptoms as voice input. So for this first he/she needs to click the Start Recording button. Then provide his/her symptoms by simply speaking. After recording their voice, the Submit button can be clicked to submit the input symptoms.

Fig. 7. Symptom Checker Page

Fig 8 shows the Symptom input. So when the user submit their symptoms they will go to the next page. That is confirm symptom page, shown in Fig 9. From the user input the key features are extracted and using the preprocessing methods and similarity matching we get the symptoms from the input. The user has to confirm their symptoms simply by marking the check boxes. There is a "PROCEED" button in the confirm page. After confirm the symptoms the user has to click that button for disease prediction.

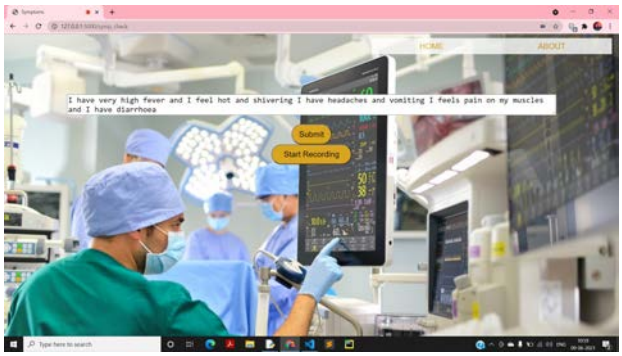

Fig. 8. Symptom Input Page

The predicted disease according to the symptoms given by the user is shown in Fig 10. In Fig 8 the input is "I have very fever and I feel hot and shivering I have headaches and vomiting I feel pain on my muscles and I have diarrhea'. So, the disease predicted is "Malaria".

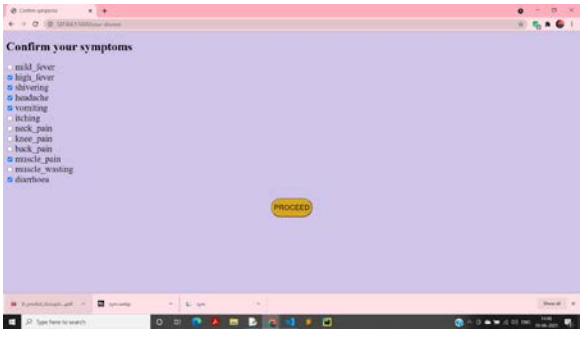

Fig. 9. Confirm Symptom Page

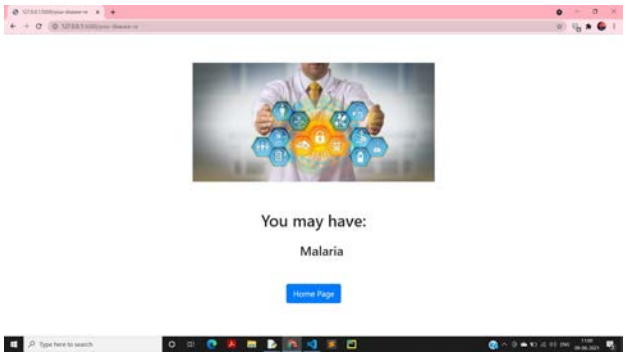

Fig. 10. Display Page

V. CONCLUSION AND FUTURE WORK

D-predict is very much useful in old aged citizen's day to day life and it is mainly more important for the healthcare sector, because they are the one that daily uses these systems to predict the disease on the basis of the symptoms. The project is designed in such a way that the system takes symptoms from the user as voice and produces output i.e, predict disease. Disease Prediction is done using NLP and ML algorithm. The User can give their symptom as input in the form of speech. After NLP processing the key features are extracted and finally the disease prediction is done using Gaussian Naive bayes classifier.

Nowadays health industry plays major role in curing the diseases of the patients so this is also some kind of help for health industry to tell the user and also it is useful for the user in case he/she doesn't want to go to the hospital or any other clinics, so just by entering the symptoms the user can get know the disease he/she is suffering from and the health industry can also get benefit from this system. If health industry adopts this project, then the work of the doctors can be reduced and they can easily predict the disease of the patient. And also as a future work medication recommendation can be implemented in the project. History about the disease for a user can be kept as a log and recommendation can be implemented for medications.

REFERENCES

[1] B. A. Thakkar, M. I. Hasan, M. A. Desai, "Healthcare decision support system for swine flu prediction using na¨ıve bayes classifier", 2010.

[2] B. Nithya and V. Ilango, "Predictive analytics in health care using machine learning tools and techniques", 2017.

[3] F. B. Putra et al., "Identification of Symptoms Based on Natural Lan- guage Processing (NLP) for Disease Diagnosis Based on International Classification of Diseases and Related Health Problems (ICD-11)", 2019.

[4] H. Q. Yu, "Experimental Disease Prediction Research on Combining Natural Language Processing and Machine Learning", 2019.

[5] Kumar, D. Lavanya, S. Madhumita, G. Rajaselvi, "Journal of Speech to Text Conversion", 2018.

[6] L. H. Patil and M. Atique,"A novel approach for feature selection method TF-IDF in document clustering", 2013.

[7] M. Hassani and T. Seidl, "Towards a Mobile Health Context Prediction: Sequential Pattern Mining in Multiple Streams", 2011.

[8] N. Prabakaran and R. Kannadasan,"Prediction of Cardiac Disease Based on Patient's Symptoms", 2018.

[9] N. Sharma and S. Sardana, "A real time speech to text conversion system using bidirectional Kalman filter in Matlab", 2016.

[10] S. Vijava Shetty, G. A. Karthik and M. Ashwin, "Symptom Based Health Prediction using Data Mining", 2019.