2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS) | 978-1-6654-0610-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/CSITSS54238.2021.9683011

Edu-bot: An AI based Smart Chatbot for Knowledge Management System

Rachana G Kummar
ECE Department
BNM Institute of Technology
Bengaluru, India
rachanakummar@gmail.com

Suhas J Shetty
ECE Department
BNM Institute of Technology
Bengaluru, India
suhasjshetty039@gmail.com

Vishwas S N
ECE Department
BNM Institute of Technology
Bengaluru, India
vishwas8970@gmail.com

Vismith Upadhya P J
ECE Department
BNM Institute of Technology
Bengaluru, India
vismu2234@gmail.com

Jyoti R Munavalli
ECE Department
BNM Institute of Technology
Bengaluru, India
https://orcid.org/0000-0002-0811-1499

Abstract--- **The pandemic has brought in a lot of changes in various aspects of human life. Every sector has been affected by this and so is the educational sector. Organizing and accessing of documents was one of the challenges that students faced during the pandemic. To overcome this problem, we propose an AI system that is equipped with a chatbot. In this study, we develop a smart knowledge management system that authenticates the user via face recognition, then a chatbot helps the users in searching and accessing the required document easily. Face Recognition is implemented using libraries such as OpenCV, Tensorflow and Face_Recognition. Face_Recognition Library provided the highest accuracy so it was used in the smart knowledge management system. A chatbot was implemented to search and access the study materials such as notes, textbooks and question banks from a particular destined folder through the hyperlinks created. To make the system user friendly, GUI was designed using Tkinter library. This AI based smart knowledge management system provides access to the required document very effectively and effortlessly**.

Keywords—AI, NLP, Face recognition, GUI, Hyperlink, Tkinter library

I. INTRODUCTION

Many things changed during the covid pandemic; one such example is the education system. The delivery of education had to change from offline to online. Theory classes, laboratories, assessments, exams, training and skill development programs, everything shifted to online mode. A lot of ICT tools were explored and utilized. All the knowledge was in the form of documents on the system. The library wasn't available for students and there were no hard-bound books to refer. That's when soft copies of the book and notes became more prominent. Documents were downloaded multiple times from various sources and were saved abruptly. The computer system was flooded with a greater number of notes and books; managing and retrieving the documents effectively and efficiently was a challenge. The searching of the document takes a lot of effort particularly when the documents have been saved multiple times at multiple locations. This results in stress for students during exams. So, to address this issue, there is a need for a smart system that helps to search and retrieve the document effectively and quickly. Therefore, this study proposes a chatbot that is based on Artificial Intelligence (AI) technique to organize, search and retrieve the documents. To provide secure access to the documents, face recognition is used for authentication. In addition to documents, the proposed AI based chatbot[1] helps the users with cafeteria information (Menu and cost of food items). Currently, chatbots are used by teachers to conduct quizzes to students and their automatic evaluation. But there aren't many chatbots which can interact directly with students or users along with their authentication. Here, a chatbot has been developed using the AI technology that serves the purpose of providing security to the user and easy access of study materials in the form of hyperlink. The paper is organized as below: Section 2 explains the materials and methods; Section 3 describes the results of the chatbot and Section 4 presents the conclusion.

II. MATERIALS AND METHODS

To begin with, we explain the basics of chatbot, face recognition and hyperlinks. Chatbot is an AI feature used to generate automated responses to user's input.

There are two types of chatbots; Generative chatbots are used in this study. These chatbots are designed to give the most appropriate response provided a database of predefined intents and their corresponding responses[2].

Face recognition is the process of verifying the identity of a person by capturing the essential facial features. These systems are used in various data processing systems to identify personalities. The procedure involved in recognizing faces is explained as below.

a) Step 1: Face Detection **-** It is the process of detecting the face of a person in an image or video[4]. The person in the image or video might be either looking straight or tilted. This is the fundamental step in face recognition.

978-1-6654-0610-9/21/$31.00 ©2021 IEEE

b) Step 2: Face Analysis **-** The essential features of the detected face are captured. The geometry of the face is read by the software. The important factors are the distance between both the eyes, the depth of the eye sockets, the distance between forehead to chin, the type of the cheekbones, and the nature of the chin, lips and ears. The aim is to identify the facial landmarks that are key to distinguishing the face.

c) Step 3: Creating faceprints - The essential features of the face that is captured is converted to numerical values. The numerical code is called a face print. Each person has their own face print that is different for each individual.

d) Step 4: Comparison and Recognition - The obtained face prints are stored as an array and compared with the images in the existing dataset. If a corresponding match is found, the recognition is performed, else alternative action is performed.

A hyperlink is a combination of alphanumeric values which are designed such that when a user clicks on it, a new document is opened. Hyperlinks are used in all websites which allows users to navigate to different pages. Graphical User Interface (GUI) is a type of interface which allows enhanced interaction of users with chatbots.

A. Chatbot workflow

The flow of chatbot working is shown in Figure 1. Initially, the user input is taken from the GUI. The input is processed using Natural Language Processing (NLP). In this chatbot, the bag of words model is used. Then the processed input is matched with a predefined dataset. The dataset is a JSON file which contains intents and their corresponding responses. If the user input matches with the dataset intent and has a probability higher than the fixed threshold, a corresponding response is chosen from the set of responses and is returned to the user through the same GUI[3].

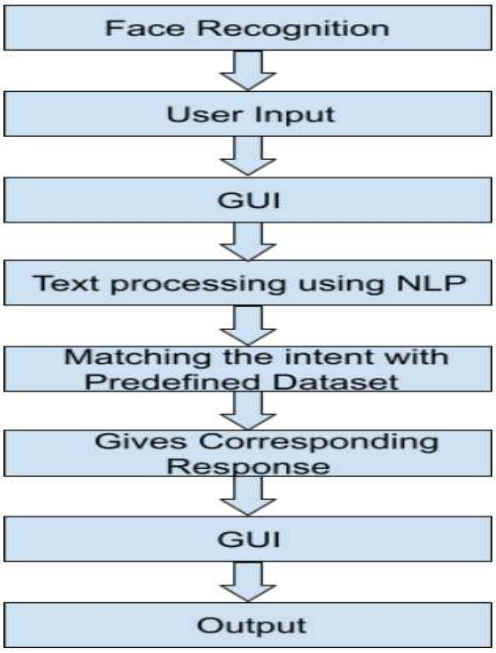

Face Recognition
User Input
GUI
Text processing using NLP
Matching the intent with Predefined Dataset
Gives Corresponding Response
GUI
Output

Figure 1: Chatbot workflow

B. Face Recognition

Face Recognition was implemented using three libraries: TensorFlow, OpenCV and Face_ Recognition, and the one with highest accuracy was selected and used further.

a) Face Recognition using the TensorFlow library:

TensorFlow is a user-friendly library which allows programmers to create graphs related to data i.e., the logical entities which describe how data navigates in graphs, or a sequence of data manipulation nodes. Each node in the graph performs a mathematical operation, and each connection or edge between nodes is called a multidimensional data array, or tensor[5]. TensorFlow provides many options for the user to process the data in ease. Nodes and tensors in TensorFlow are objects in python, and the generated TensorFlow applications are nothing but the Python applications.

But the original mathematical operations are not computed in Python. The transformation libraries available in TensorFlow are created as high-performance C++ binaries. TensorFlow applications can be used on any target application through either a local machine, in the cloud computing domain, Android and iOS devices, CPUs and GPUs.

b) Face Recognition using OpenCV:

OpenCV (Open-Source Computer Vision Library) is an open-source Computer Vision and a machine learning software library designed to solve computer vision problems. The face recognition is performed with OpenCV using a pre-trained OpenFace model[6].

Implementation process

- Adding dataset to the Datapath.
- Appending the images to the respective filename. For each category 40 images are given.
- Creating empty face_embeddings and face_names and processing the images.
- Using blobFromImage above functions are performed.
- The model architecture is defined by prototxt file whereas weights of layers is defined by caffemodel file based on these models, face embeddings are created and appended to the empty list to quantify a face.
- Detecting faces - By using above files, face will be detected but not recognized.
- 128-dimensional face embeddings are used to measure a face. The input process to compute includes: 1. The network takes input data. 2. Triplet loss function.
- We are using three images namely anchor, negative and positive respectively to train a model.
- Two images of the same person: anchor and positive whereas negative contain image of different face.
- For each face 128-dimensional embeddings are computed which results in the weights of the network. Images with the same face come closer if not embeddings are pushed away.
- Apart from embeddings, the model is trained using SVM.

- Fix a threshold value and here we are giving 50%, if the datasets have many images, we can set the threshold value to 30% still it will recognize the face.

c) Face Recognition using Face_recognition Library

The simplest and the easiest library used for recognizing faces is Face_Recognition Library. Dlib's face recognition is used to construct this library along with deep learning. Face recognition algorithm extracts features from a face image from the parts such as forehead, nose, mouth, chin, jaws, eyes. The library can be imported by using a simple face_recognition command line. Most of the operations that take place at the backend of face-recognition library are as follows:

i. Detecting face in an image: The method used for detection is called HOG (Histogram of Oriented Gradients)[7] and the image is first converted into grayscale. The current dark pixel is determined by comparing the pixel with its surrounding pixel. Then draw an arrow in all the directions of the pixels which are getting darker. For all the pixels in the image this procedure is repeated for all the images given. These arrows are called Gradient. Finally, lots of information is retrieved from the gradient of each and every pixel. Next is to reduce all the images into small squares of size 16x16. Count the gradient position in the prime direction and return the square in the image with a single arrow in the direction that had the most of the arrows facing. From a cluster of training faces to finding the face in an HOG image, we should find the part of the image that looks similar to a known HOG pattern that was extracted from a bunch of other training faces.

ii. Posing and Projecting Faces:

This is done by estimating face landmarks i.e., finding 68 particular positions on every face. After specifying the points, we can say how eyes and mouth are aligned by rotating the images that are centered.

iii. Encoding Faces:

Deep CNN is used to train the image to generate 128 measurements for each face. We measure the size of each ear, the length of the nose, the spacing between the eyes etc. The training is done by using 3 face images, 2 images of the same person and the third one is of a different person. This is also called a single "Triplet" training step. Training has been repeated for thousands of images, thus it yields 128 measurements for each person. These measurements are called "Embeddings".

iv. Recognition

Once all the images in the database are encoded, the 128 embeddings are collected in the same dataset. When the image is captured using a webcam, the three steps: detection, finding landmarks and finding embeddings are performed and then compared with the stored measurements to find the perfect match or else the person is recognized as unknown.

C. Natural Language Processing (NLP)

NLP is a futuristic technology gaining a lot of importance in analyzing human language. NLP has made a lot of impact in various fields like medical, search engines and many more[8]. It helps the computers to understand human language whether it's written or through speech. NLP makes use of Artificial Intelligence to process the human language by taking in real time data and feeding it to the computer to process it using programs that are understood by the machine[9].

NLP is categorized into two groups which mainly help in building a chatbot, they are

1. Natural Language Understanding (NLU) is a subset of NLP, which helps the machine to understand the input data. The linguistic stages correct the grammar and syntax in the sentence and find the meaning of the sentence.
2. Natural Language Generation (NLG) is a subset of NLP, used to generate meaningful statements. Few NLG models are RNN, LSTM and Markov Chain.

After taking in the input, it preprocesses, classifies and trains through an algorithm.

The preprocessing includes few steps such as:

1. Tokenization, the paragraphs and sentences are broken down into smaller words.
2. Lemmatization and Stemming, the words are reduced to their root words.
3. Removing Stop words, removing less informative words.
4. Normalization, converting text into its standard form.
    - Text Classification can be done using Bag of Words, machines are unable to learn from raw information, so the data needs to be converted into numeric values. This conversion can be done using One-Hot Encoding technique.
    - Word Embedding is performed after converting the data into numeric form. The words that appear in similar contexts will have similar vectors.
    - The vector is trained using a suitable machine learning algorithm such as Support Vector Machine, XGBoost etc.

NLP has a wide range of applications, it can be used in Chatbots, text summarization, Text Mining, Machine Translation and many more.

D. Graphical User Interface (GUI)

Graphical User Interface is a structure of interaction between user and computer software. A GUI takes user input and displays the same. It includes GUI elements like dialog box, menu, ribbons and many more. Use audio and visual effects will enhance the user interface. Such elements play an important role in building a GUI and it is very convenient for users.

GUI elements:

GUI is one of the major interfaces in which user can interact with computer. Some of the GUI components are[10]:

- Button - When a button is pressed a certain specific action will be performed.
- Dialog box - Dialog Box takes user input and also displays other details.
- Icon - Icon describes the feature or program graphically.
- Menu - Users can make a choice from the list.
- Ribbon - Contains program activities and also includes toolbars.
- Toolbar- generally present on the top of an application and performs specific functions.
- Window - Displays the current program being used on the computer screen.

Tkinter library:

The Window is the foundation element of a Tkinter GUI. Windows behave like containers inside which all GUI elements present and those elements are, buttons, text boxes and labels, these are also called widgets. Tkinter is one of the best available GUI libraries for Python. Tkinter along with Python provides a fastest and easiest way to create applications with better GUI. Also, Tkinter gives a better and powerful object-oriented interface.

Following are the steps present in creating GUI applications from Tkinter:

- First is importing the Tkinter module.
- Creating the GUI application main window which contains widgets.
- Then adding one or more widgets depending on the user requirements to the GUI application.
- Final step is taking actions against each individual event triggered by entering the main loop.

E. Hyperlink

A hyperlink is a pointer to the information that a user gets after clicking on it[11]. A hyperlink points to a full document or to a particular element within that document. Hypertext is text formed with hyperlinks. To view and create hypertext a software system is used and it is known as Hypertext system. When a user is said to follow hyperlink, it is said to be navigating or browsing the hypertext.

The hyperlink can be considered as an interface which links a source to the target. When the user clicks the link at the source it will directly navigate to the target. These links can appear as URLs, text and images. In Search Engine Optimization (SEO), the anchor text is the most important one. Anchor text is also a type of hyperlink and can be represented by a plain text.

A predefined linkage between one object and another. A hyperlink generally refers to a text or icon on a webpage. When a user clicks on it, the user gets access to another page of the current website or other website.

A hyperlink can be said as a property of an element like word, phrase, symbol or image present in a document that causes it to display another document whenever the user clicks it. In hypertext systems, such a type of element is either bolded or underlined to indicate to the user that a link to another document is available. HTML (Hypertext Markup Language) and SGML (Standard Generalized Markup Language) programming languages are used to create the Hyperlinks.

In coding, a label is bound to a "<Button-1>" event. A callback function is executed when a label is raised resulting in a new web page opening in the browser.

III. RESULTS & DISCUSSION

In this section, first we present the comparison between the three libraries used for face recognition. The comparison is as shown in Table 1. The parameters compared are dataset size, accuracy, processing time, algorithm used and the memory requirement and installation setup.

Table 1: Comparison between face recognition libraries

Parameters	TensorFlow	OpenCV	Face Recognition
Dataset size	Large number of images are required	Moderate number of images are required	Even a single image is sufficient
Accuracy	92.75%	94.3%	98%
Processing time	Very high	Moderate	Very low
Model/Algorithm used	Keras, VGG16	OpenFace	HOG (Histogram of Oriented Gradients)
Memory and installation	Large memory but easy to install	Less memory and easy to install	Large memory and difficult to install

Face recognition library was selected as it provides highest accuracy with very low processing time. The dataset used is also very less for training the model. The only issue is it requires large memory and installation is difficult but it's a one-time event.

When a user tries to interact with the chatbot, the system first opens the camera to recognize the user's face. If the face is recognized then it displays the user's name as shown in Figure 2.

Once the face matches with the stored/trained data, it will ask the user, whether the user wants to communicate with bot or only get the links of study materials. If the user selects the option of only materials, then the system returns a set of hyperlinks, which the user can click to access study materials.

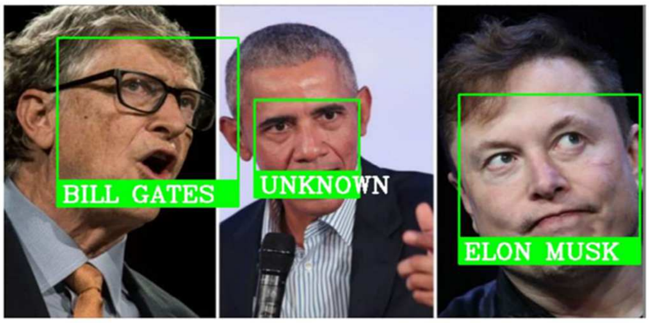
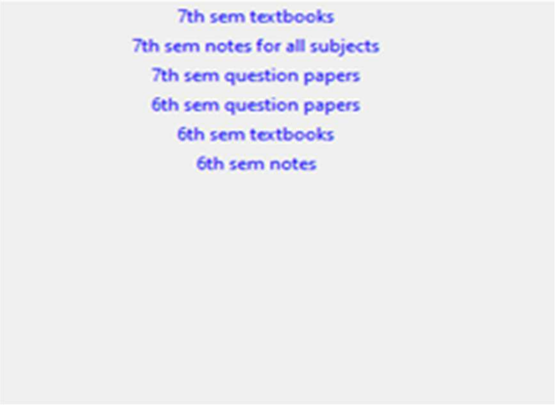


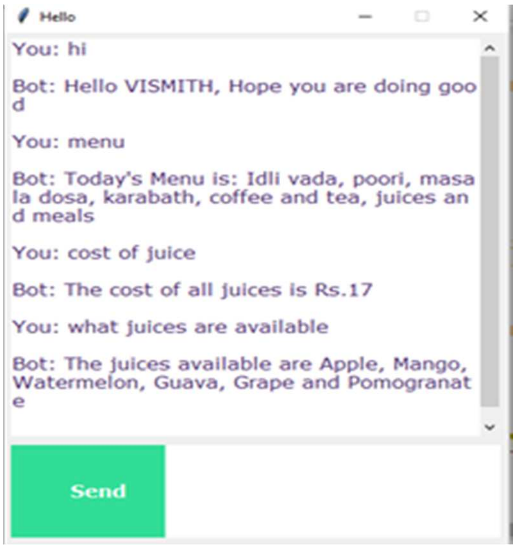
Figure 2: Face recognition

This is shown in Figure 3. If the user selects to communicate with chatbot, then the bot opens and the user can start a conversation with it. For example, if the user asks a canteen related question it displays the related answer as shown in Figure 4.



7th sem textbooks
7th sem notes for all subjects
7th sem question papers
6th sem question papers
6th sem textbooks
6th sem notes

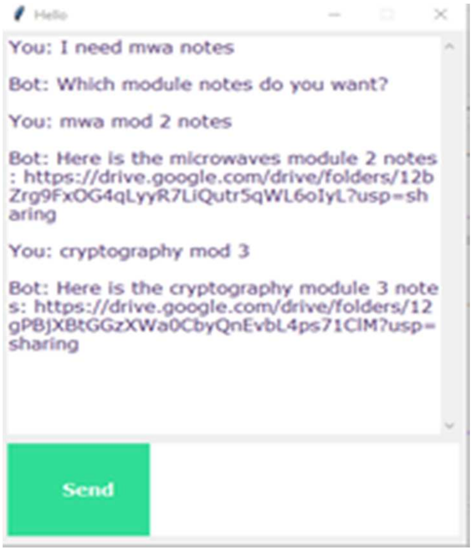
Figure 3: Hyperlinks



```
You: hi

Bot: Hello VISMITH, Hope you are doing goo
d

You: menu

Bot: Today's Menu is: Idli vada, poori, masa
la dosa, karabath, coffee and tea, juices an
d meals

You: cost of juice

Bot: The cost of all juices is Rs.17

You: what juices are available

Bot: The juices available are Apple, Mango,
Watermelon, Guava, Grape and Pomogranat
e
```

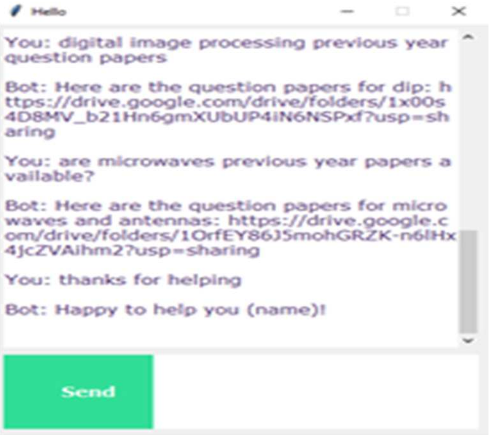
Figure 4: Chatbot Response 1

Also, when a user wants to access the notes of a particular subject, the user can ask the question by mentioning the subject name and particular chapter or module number and the bot returns the respective drive link through which he can get the documents as shown in Figure 5. Along with notes, users can get the previous year question papers for reference by specifying the subject name and get the drive link. At last, the user can have a friendly conversation with the bot as shown in Figure 6.



```
You: I need mwa notes

Bot: Which module notes do you want?

You: mwa mod 2 notes

Bot: Here is the microwaves module 2 notes
: https://drive.google.com/drive/folders/12b
Zrg9FxOG4qLyyR7LiQutr5qWL6oIyL?usp=sh
aring

You: cryptography mod 3

Bot: Here is the cryptography module 3 note
s: https://drive.google.com/drive/folders/12
gPBjXBtGGzXWa0CbyQnEvbL4ps71ClM?usp=
sharing
```

Figure 5: Chatbot Response 2



```
You: digital image processing previous year
question papers

Bot: Here are the question papers for dip: h
ttps://drive.google.com/drive/folders/1x00s
4D8MV_b21Hn6gmXUbUP4iN6NSPxf?usp=sh
aring

You: are microwaves previous year papers a
vailable?

Bot: Here are the question papers for micro
waves and antennas: https://drive.google.c
om/drive/folders/1OrfEY86J5mohGRZK-n6lHx
4JcZVAihm2?usp=sharing

You: thanks for helping

Bot: Happy to help you (name)!
```

Figure 6: Chatbot Response 3

IV. CONCLUSION

Currently, the chatbots are used in most of the industries such as business, e-commerce, etc. The AI based smart chatbot system is designed to be used in educational institutions by students to access their study materials and videos. This will help students to get all the required materials during the examination which will help them to prepare well. Additionally, students' details will also be displayed.

The usage of this knowledge management system results in saving a lot of time and money as the chatbot eliminates the effort to search for examination materials manually. This avoids the necessity to take printouts of the documents. All the materials and details will be organized in a specific manner which is easy to handle.

One of the main advantages of this bot is that it can be used in any domain by just changing the dataset. It can be implemented for different organizations by uploading corresponding datasets. Also, it can be either developed as an app if required or can be implemented in any of the organization's websites. Changing the training algorithm and increasing the number of training steps will improve the accuracy. This chatbot provides cost effective solution since there isn't much hardware involved.

The bot is authenticated through face recognition. When the bot is unlocked, the user is given two options. He can either get access to multiple documents or access documents individually. Multiple documents can be obtained by just clicking on the hyperlink in the GUI else the user can access the individual documents by interacting with the bot. If another user tries to access the bot, the kernel will die and hence security of the chatbot is maintained.

In future study, this bot can be extended to give more precise and effective responses by giving speech as an input. It can be integrated in institution’s websites further helping students to use it easily.

REFERENCES

[1] Bhaumik Kohli, Tanupriya Choudhury, Shilpi Sharma and Praveen Kumar ‘A Platform for Human-Chatbot Interaction Using Python’ IEEE, 16-18 Aug. 2018, ISBN:978-1-5386-5658-7

[2] Satyendra Praneel Reddy Karri and B. Santhosh Kumar, ‘Deep Learning Techniques for Implementation of Chatbots’ IEEE, 2020 International Conference on Computer Communication and Informatics (ICCCI), ISBN:978-1-7281-4515-0.

[3] E. Kasthuri and S. Balaji, "A Chatbot for Changing Lifestyle in Education," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV*)*,2021, pp.1317-1322, doi: 10.1109/ICICV50876.2021.9388633

[4] S. Lin and F. Su, "FCFD: Teach the machine to accomplish face detection step by step," 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 3214-3218, doi: 10.1109/ICIP.2016.7532953.

[5] T. Ahmed, P. Das, M. F. Ali and M. -. F. Mahmud, "A Comparative Study on Convolutional Neural Network Based Face Recognition," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1-5, doi: 10.1109/ICCCNT49239.2020. 9225688..

[6] S.V.Viraktamath, Mukund Katti, Aditya Khatawkar and Pavan Kulkarni, “Face Detection and Tracking using OpenCV”, July-August 2013, The Standard International Journals (The SIJ), Vol. 1, pp. 45-50. ISSN: 2321–2403.

[7] N. J. Wang, S.C. Chang and P. J. Chou. Taipei, Taiwan,’ Face Detection in Real Time Based on HOG’. IEEE, DOI:10.1109/ISPACS.2012.6473506, 2012. International Symposium on Intelligent Signal Processing and Communications Systems. pp. 333-337. ISBN: 978-1-4673-5081-5.

[8] C. Saligram, Deepali B K, Gowri K S, J Ajay Kumar and J. R. Munavalli, "Voice Controlled Personal Health Assistance Device using Natural Language Processing and Random Forest Disease Classifier" 2020 Journal of Critical Reviews, Vol. 7, Issue 15, pp.3255-3265.

[9] J. Cai, J. Li, W. Li and J. Wang, "Deep learning Model Used in Text Classification," 2018 15th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP),2018, pp.123-126, doi:10.1109/ICCWAMTIP.2018.8632592

[10] L.Kluger, "The Open Look graphical user interface and its toolkits," IEE Colloquium on User Interface Management Systems, 1989, pp. 3/1-3/3.

[11] F.Li, "Extracting Structure of Web Site Based on Hyperlink Analysis," 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008, pp. 1-4, doi: 10.1109/WiCom.2008.2538.