ISSN: 2586-7652 (Print) Vol. 06, No. 01, March 2023
ISSN: 2635-7607 (Online)
International Journal of Advanced Engineering
Source: http://ictaes.org
Manuscript received: January 25, 2023; Revised: February 17, 2023; Accepted: March 6, 2023

Nepali Voice-Controlled Home Automation System

Aayushma Pant[1,*], Shrijak Dahal[2,*], Rohit Gautam[3], Saugat Guni[4], and Saroj Shakya[5]

[1,2,3,4,5]*Institute of Engineering, Tribhuvan University, Nepal*

tha074bex001@tcioe.edu.np[1], tha074bex041@tcioe.edu.np[2]

Abstract

This research work aims in using Nepali Speech for automating the home applications which is inspired by Speech recognition technology as featured in Siri and Alexa. Our research work puts forward into two divisions. First is speech recognition which uses mathematical logic like time domain conversion, STFT (Short-time Fourier Transform) for feature extractions, and hybrid CNN-LSTM neural networks for training the extracted features in converting the speech in Nepali language into predicted text in English language. The second part resembles the implementation of predictive text into IoT devices/sensors through MQTT protocols that offer bidirectional communications between database and sensors.

Keywords : *CNN (Convolutional Neural Network), Home Automation, IoT (Internet of Things), LSTM (Long Short-Term Memory), MQTT (Message Queuing Telemetry Transport) protocol, Speech Recognition.*

1. Introduction

Home automation plays a vital role in improving the living standard. It reduces human efforts, increases energy efficiency, and is time-saving. The voice-controlled home automation, an additional feature makes it easier in automating and controlling appliances like light, door, fan, AC (Air Conditioner) etc. used in house by just a command of voice especially for old elderly people. It is a technology that recognizes spoken words, phrases, sentences, and converts them into text. Facebook, Amazon, Microsoft, Google, and Apple five of the world's top tech companies are already offering this feature on various devices through services like Google Home, Amazon Echo, and Siri, Alexa. However, developing countries like Nepal are still beyond the reach. Furthermore, most of the local people don't even understand the English language. According to CBS (Central Bureau of Statistics), only 0.30% of people are English speakers in Nepal [1]. So, this motivated us to bring this

* These authors contributed equally to this research work
Corresponding author: Aayushma Pant
Author's affiliation: Institute of Engineering, Tribhuvan University, Nepal
Email: tha074bex001@tcioe.edu.np
Copyright © ICT-AES

work that carries out the home automation tasks using the native Nepali language.

This work is designed fulfilling the requirements of home automation. It uses a speech recognition model trained on Nepali commands/datasets using hybrid CNN and LSTM. Then the trained model is deployed in the Django web application where we fetch the audio signal and predict the accurate results. The predicted results are then fetched by NodeMCU (Node Micro Controller Unit) using MQTT protocols giving the response through home applications.

2. Related Works

Smart device systems have been evolving rapidly through the 2000s. It's estimated that there were already 1.5 million automated home systems in place by 2012 A.D. [2-3]. Today, IoT devices are more plentiful than ever. There are a variety of cloud-based voice services that serve as a personal assistant for users that are able to call a taxi, turn smart lights on/off, answer questions, book at restaurants and many other tasks based on the user's voice commands. A few of the well-known voice assistants are: Alexa from Amazon, which is used in products like Amazon Echo, Amazon Tap, etc. Siri from Apple Inc. is used in the Apple Home pod which serves a similar purpose. Google Assistant used in Google Home has additional features where it can recognize up to six different users and pull their respective details to converse with them. These voice assistants are capable of performing multiple tasks mostly due to the application of various subfields of AI (Artificial Intelligence). This development is demanding both academics and industry to make more advancement in automation systems.

Looking over the past works on IoT based home automation, most of the voice-controlled automation uses the Web Speech API (Application Programming Interface) that provides the facility for speech input to text output [4]. Using such API for automating things might not replicate original content, as well as these, are quite expensive for academic use. Similarly, the combination of pattern recognition and the acoustic-phonetic approach are also popular for speech recognition utilizing information like spectrogram, phonetic, and linguistic, the corpus of the audio signals, and provides better results with high accuracy [5]. Though these approaches are popular but still have no significance accuracy also is not appropriate for native local languages like Nepali.

Secondly, most of the IoT based applications use Firebase for data storage [6]. Firebase restricts the backend which only supports the static files and limits the control over server related processes. To minimize this difficulty, we used the Heroku Database that offers a higher level of flexibility with real-time logs and providing access to all SQL (Structured Query Language) databases [7].

Also, the MQTT protocol which we use in our work is different from the use of HTTP (HyperText Transfer Protocol) protocol as HTTP protocol gives direct communication between IoT devices and sensors and takes high power which can be minimized using MQTT as it offers security as well as works in low power [8-9].

Besides this, in comparison to the various algorithms used in the neural network, most of the research works used HMM (Hidden Markov Model) for speech processing but with time, this model might not give long-term flexibility [10]. To minimize the problem, we trained the model using the LSTM algorithms. LSTM models are compared to have high standard deviation and better results when the classification is made using multiple datasets [11].

3. Methodology

3.1. Datasets Collection

We collected the datasets of Nepali voice commands for speech recognition by making a website where the voice of the user is recorded and saved into our database. It consists of a 4-second audio recording on 10 different topics. It was performed by more than 200 people both men and women ranging from small age groups to old aged groups from which we were able to collect about 580 datasets. The samples were made to have variations in terms of voice, frequency, pitch, speed, and noise by using the librosa library in python. From which we collected 5800 datasets for our training purpose. The validation set was made to have 20% of our total datasets equal to 1160 datasets to classify the audio on the base of 10 different labels for performing our task. The total labels in our datasets can be shown in the screenshot of our website as in figure 1. The translation of the screenshot shown in Figure 1 is given in Figure 2.



Figure 1. Website for Dataset Collection.

The above screenshot in figure 2 shows how we saved the user spoken audio in this Heroku database and then downloaded it into our computers according to the label justified.

3.2. System Block Diagram

Our application performs with the help of the hybrid CNN-LSTM architecture, the CNN architecture only has convolutional layers and pooling layers. It signals which are one dimensional, so we use convolution 1D layers, pooling 1D layers [12]. Here CNN takes the time series data in a one-dimensional wherein the data are arranged in the order of sequential time instants., the input one-dimensional data vector is $x = (x_1, x_2, ..., x_{n-1}, x_n, y)$ where $x_n \in R^d$ denotes features (here time series data) and $y \in R$ denotes a class label. The convolutional layer uses a ReLU (Rectified Linear Unit) activation function that applies max (0, x) to each of the inputs to the ReLU represented by x. This produces the most significant features (here the selection of features with the highest values). These selected features are fed to LSTM architecture to learn the long-range temporal dependencies. Finally, a signal dense layer activated by SoftMax used to classify the input voice commands.

speech	text	translation
6,887 bytes	बत्ती बाल	Turn on the lights
11,001 bytes	बत्ती निबाउ	Turn off the lights
6,358 bytes	बत्ती बाल	Turn on the lights
24,040 bytes	बत्ती बाल	Turn on the lights
23,258 bytes	ढोका खोल	Open the door
3,584 bytes	कोठाको तापक्रम देखाउ	Check room temperature
6,556 bytes	कोठाको तापक्रम देखाउ	Check room temperature
10,326 bytes	ढोका बन्द गर	Close the door
554 bytes	पानी मोटर बन्द गर	Turn off the pump
12,049 bytes	पानी मोटर बन्द गर	Turn off the pump
3,592 bytes	पानी मोटर खोल	Turn on the pump
10,927 bytes	पानी मोटर खोल	Turn on the pump
23,162 bytes	घर सुरक्षा निष्क्रिय गर	Turn off security
22,540 bytes	घर सुरक्षित गर	Turn on security
23,627 bytes	ग्यास चेक गर	Check the gas

Figure 2. Dataset Stored in Heroku Database.

The above screenshot in figure 2 shows how we saved the user spoken audio in this Heroku database and then downloaded it into our computers according to the label justified.

According to the system diagram as shown in figure 4, our research work's workflow starts with the collection of Nepali audio datasets. Datasets are collected through the web applications that are stored in a database which were trained by hybrid CNN-LSTM model as in figure 3. This trained CNN-LSTM model consisting of necessary weights and bias are deployed in the server using the Django backend functionalities which serves as the necessary tool for our system.

In the user interface, first, the speech is provided to the mobile app interface. This speech which gets collected as blob objects is then processed in the server for feature extraction. Feature extractions include the conversion into the frequency domain, Fourier transformations, and spectrogram formations. Then the extracted features are sent to a trained CNN-LSTM model for prediction. The predicted text is then passed to the database using the SQL queries for further process.

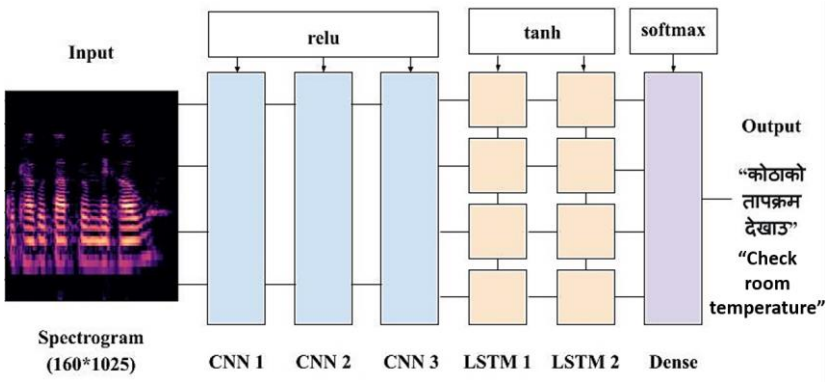


Figure 3. Hybrid CNN-LSTM Model.

For fetching data from the database, the MQTT protocol has been used. In this protocol, a topic is created in the MQTT broker where a sensor publishes data in that topic which is assigned to it and actuators subscribe to that topic to get information from the sensor. Here MQTT broker and subscriber are present in the server, MQTT subscriber fetches data from the database provided by voice command and sends it to the actuator subscribing

to that topic. A mobile app is being used in which the user can get information about the sensor and give a voice command to control the actuator.

4. Implementation Details

As stated earlier, the speech recognition is performed with the help of CNN-LSTM model helping to extract the patterns and sequence of the audio spectrogram into the predicted text. These texts are stored in the database in binary format as shown in figure 8. In the meantime, a broker is established in the local computer, that collects the data from a database stored by the publisher, and displays it in our app. Similarly, the ordered command which is stored as a binary key value in the database is used to trigger a respective subscriber. For example,

DHT11 (temperature sensor) used in our work to sense the temperature follows one to one protocol/ serial communication between the NodeMCU and sensor. The digital output is introduced with the pull-up resistor to eliminate any garbage value at the beginning of the connection. It has a maximum threshold range from 0-50℃ and humidity of 20-90% with accuracy between 2℃ and 5% respectively.

PIR (Passive Infrared) sensor, used for detecting the motion that helps us in alerting us from any thief coming into the house. We implemented triggering it using it by using a relay. The sensitivity and delay time is changed to get the desired value. A gas sensor is used for sensing the smoke and gas in the room and changes its state to high are used in our work to alert people when there is fire or gas leakage in the home.

5. Results and Analysis

The outcome is seen in the mobile app where the converted text is displayed as spoken by the user. It also has the notification section which pops up as it gets triggered with the fire sensor and the PIR sensor i.e., displaying fire alert and thief or unknown people coming at home alert respectively. Besides, it gives the information of temperature and humidity as commanded by the user. The text also can be toggled between English (Left) and Nepali (Right) as seen in figure 5.

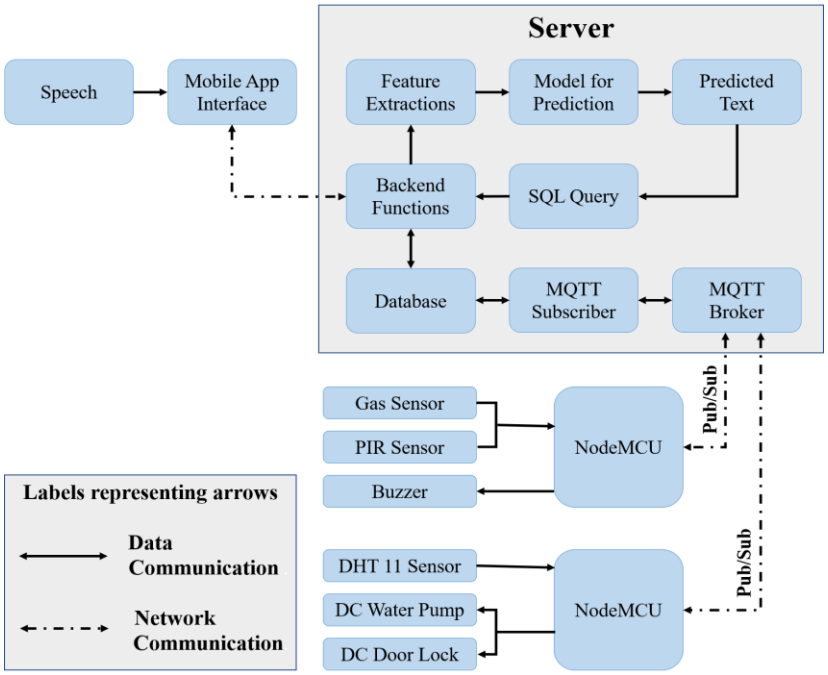


Figure 4. Block Diagram of the System.



Figure 5. Screenshots of Mobile UI.

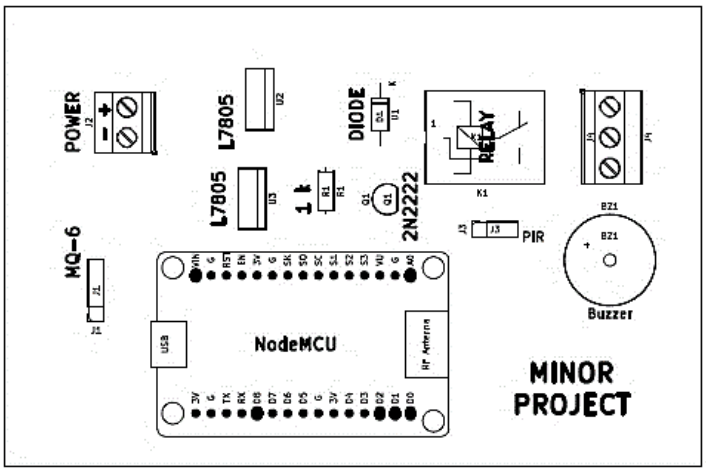


Figure 6. PCB Diagram of NodeMCU with security sensors and a relay (Top Layer).

The PCB (Printed Circuit Board) diagram of the home automation system is shown below. In figure 6, which is the top layer of PCB, there are two security sensors for detecting intruder (PIR) and sensing leakage gas (MQ-6) in the home. Also, there is a buzzer for alarm, and a relay switch which can turn the lights on/off. Figure 7, is the back side of PCB where all the circuit routes are designed.

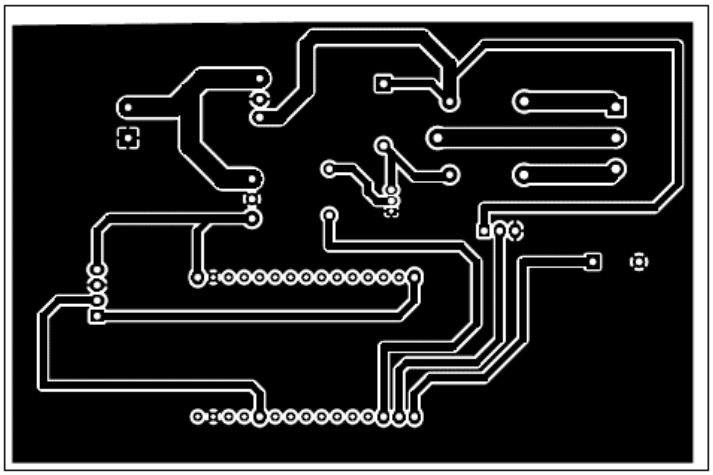


Figure 7. PCB Diagram of NodeMCU with security sensors and a relay (Bottom Layer).

```
speech=# SELECT * FROM status;
 id | is_light_on | is_motor_on | is_door_on | is_security_on | pir_state | gas_state | temperature_value | humidity_value
----+-------------+-------------+------------+----------------+-----------+-----------+-------------------+----------------
  1 | 1           | 0           | 1          | 0              | 1         | 0         |                18 |             25
(1 row)
```

Figure 8. Fields in Database.

The locally hosted Postgres database looks like as shown in Figure 8. The above diagram represents the clip of database, that stores the predicted result in the binary text form. Here the first 4 columns give the bit value of the on-off state off as directed by the voice command. The rest 4 values give the value of the PIR sensor, gas sensor, and DHT11 sensor respectively with the help of MQTT protocol.

5.1. Result Analysis

In this section, we show the results of our speech to text analysis tests using different network models. Aside from using CNN-LSTM model for speech recognition, we also compare against standard CNN and LSTM networks.

Table 1. Average accuracy of different neural network models.

Neural Network	Average Accuracy (Training / Testing)
CNN	100% / 48.62%
LSTM	90.99% / 83.79%
CNN-LSTM	95.22% / 90.69%

Our CNN-LSTM model achieved an accuracy of 40% higher than the CNN model, and 7% than the LSTM model. These results seem to indicate that our initial intuition was correct and that by combining CNNs and LSTMs we can harness both the CNN's ability in recognizing local patterns and the LSTM's ability to harness the sequence order. However, the ordering of the layers in our models plays a crucial role in how well they perform bringing high accuracy in the model.

We can see the model performance going better as increment in the epoch and decrease in the losses as shown in figure below:

5.1.1. Using CNN Model

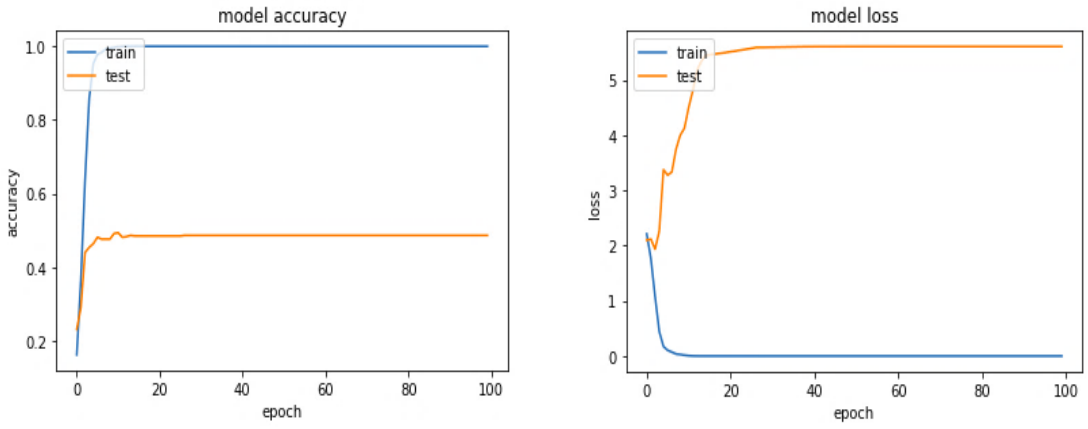


Figure 9. Model accuracy and loss versus epochs in CNN model.

Here, the large variation between training and testing clearly shows the model overfits on the known trained dataset and under fits on the unknown test dataset.

5.1.2. Using LSTM Model

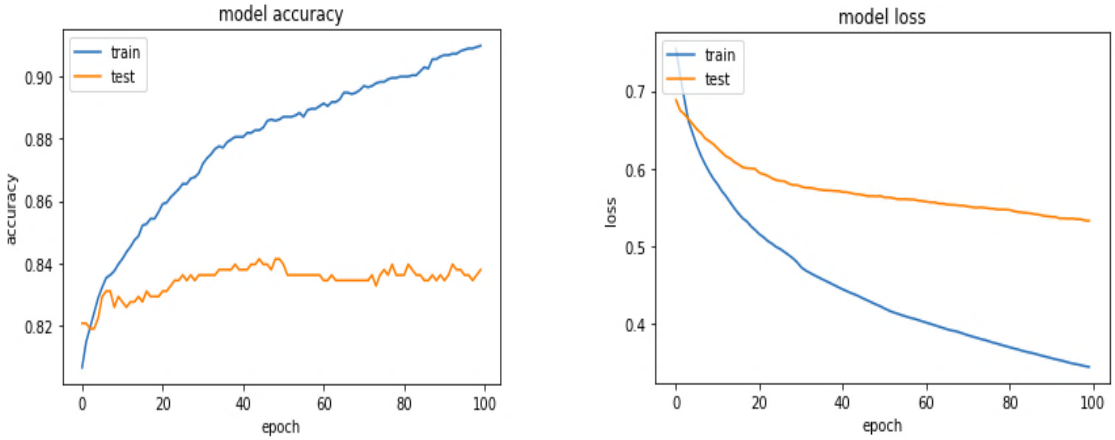


Figure 10. Model accuracy and loss versus epochs in LSTM model.

Here, the model gives a better result on a known trained dataset of about 90% accuracy but is about 84% accurate on an unknown test dataset.

5.1.3. Using CNN-LSTM Model

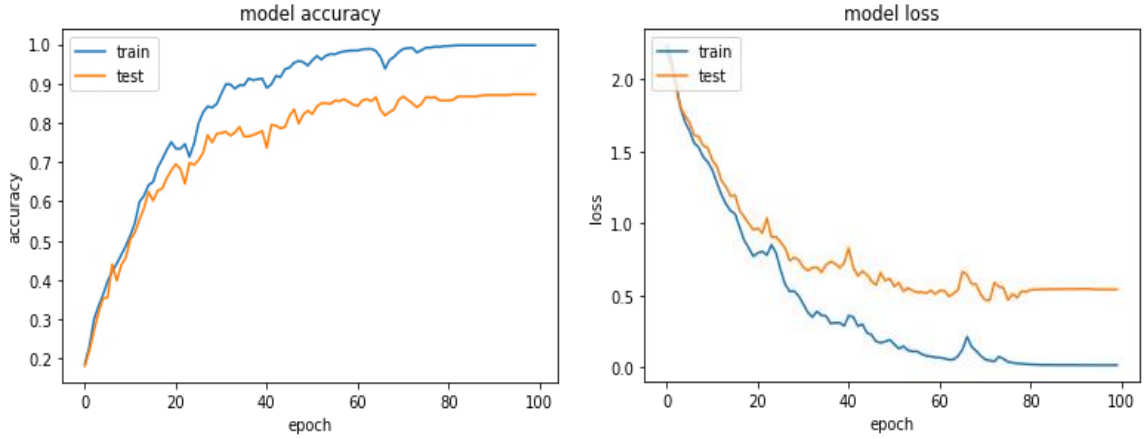


Figure 11. Model accuracy and loss versus epochs in CNN-LSTM model.

Here, the losses of training and testing dataset decrease over epoch and the accuracy of training and testing dataset gets saturated after 80 epochs.

The accuracy of the CNN-LSTM model is over 90% for the training and testing dataset. So, overall, it utilizes the pattern recognizing feature of the CNN model and sequence recognizing feature of the LSTM model and gives good result.

5.2. Confusion Matrix

Our CNN-LSTM model is evaluated using different metrics and one of them is confusion matrix. The diagonal of the confusion matrix shows the numbers of test data that is predicted true with the label assigned and the rest

of the other digits are the errors. From the confusion matrix, we can determine accuracy, precision and recall acting as a deciding factor for whether the model need to be retrained or not. Here, figure 12 shows the confusion matrix of our CNN-LSTM model.

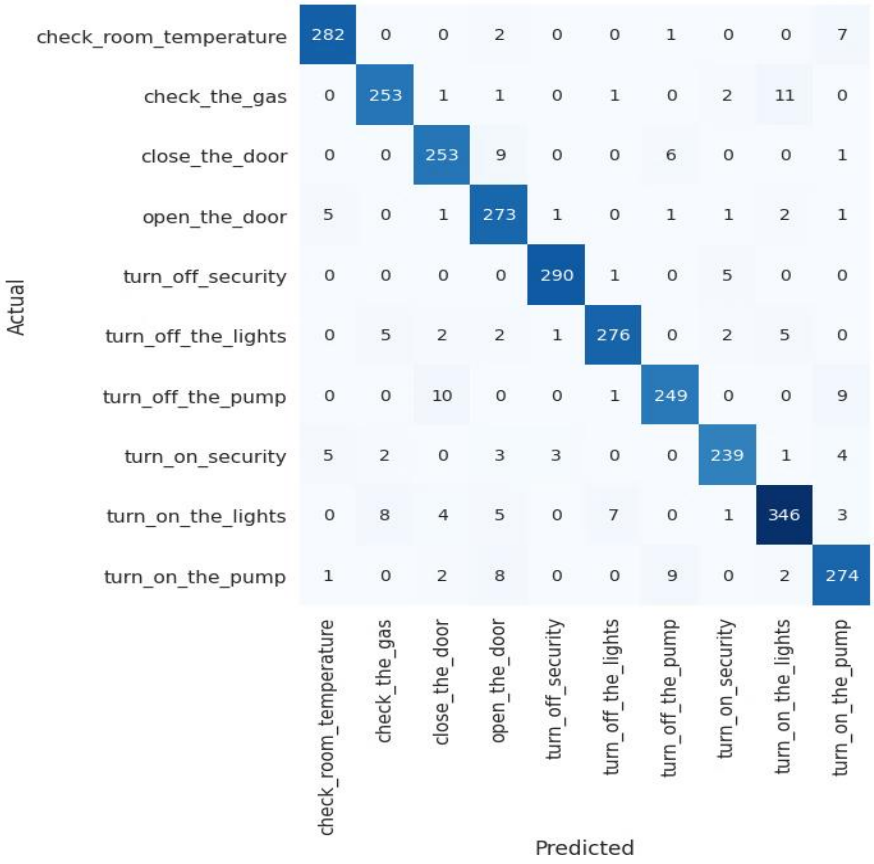


Figure 12. Confusion Matrix of hybrid CNN-LSTM model.

5.3. Classification Report

	precision	recall	f1-score	support
check_room_temperature	0.96	0.97	0.96	292
check_the_gas	0.94	0.94	0.94	269
close_the_door	0.93	0.94	0.93	269
open_the_door	0.90	0.96	0.93	285
turn_off_security	0.98	0.98	0.98	296
turn_off_the_lights	0.97	0.94	0.95	293
turn_off_the_pump	0.94	0.93	0.93	269
turn_on_security	0.96	0.93	0.94	257
turn_on_the_lights	0.94	0.93	0.93	374
turn_on_the_pump	0.92	0.93	0.92	296
accuracy			0.94	2900
macro avg	0.94	0.94	0.94	2900
weighted avg	0.94	0.94	0.94	2900

Figure 13: Classification Report of hybrid CNN-LSTM model.

It shows the precision, recall and F1-score of various labels. Here, precision refers to what per cent of predicted results are true and recall refers to what per cent of the actual positives are correctly classified. For the F1 score, it is a function of precision and recall. The combined F1-scores form macro-averaged F1-score It is also measured

with the weighted parameter to form micro weighted score. Moreover, the classifier accuracy is determined by the micro F1 score that is equal to micro precision and recall.

$$\mathbf{F1-score} = \frac{\mathbf{2} * (\mathbf{precision} * \mathbf{recall})}{(\mathbf{precision} + \mathbf{recall})} \quad (1)$$

From the result, it is analyzed that the model accuracy was about 94% in the testing datasets giving us positive remark for using it.

6. Future Enhancement

Automating House using the national language has lots of scope and can help in various ways. Our research work is working on certain few important tasks that occur in every house. So, this work can be enhanced with a larger number of applications it provides, with greater algorithms like Natural Language Processing and more data that it can understand other different languages like Maithili, Bhojpuri, Newari etc. And also, can be made better using trigger word detection like saying “Kanchi” for listening to commands without manual clicking of button.

Besides this, this research work can be taken to the next level by creating voice biometric systems for security in Banks, Hospitals etc. Also, with the increment of data, and advancements of algorithms these research works can be commercialized and can help in making smart homes that reduce human effort.

Not only this, with the implementations of AI chatbot, home history, energy management schemes like automatic turning off light when not in use, this research work can be enhanced. The model can also be made to automate the vehicle using the voice.

So, there is no doubt that by implementing all these ideas we can create a better enhanced AI-based voice-controlled home automation system.

7. Conclusion

Nepali Voice Controlled Home Automation is a new system that understands Nepali Language and automates the home as per our voice command. This research work starts from collecting different Nepali voice commands to the implementation of these data in predicting the voice through the successive training and hyperparameter tuning and testing. We got accuracy, precision, and recall of about 94%. Furthermore, this model is implemented using Django server and MQTT protocol and were able to automate the home. It can perform various tasks like pumping water, turning on/off lights, measuring temperature, security check, door locking and opening, and gas leakage checking. Hence integration of necessary sensors and actuators with the AI software model, this work assures to create a smart home automation system.

References

[1] Population Monograph of Nepal. Kathmandu: Government of Nepal, National Planning Commission Secretariat, Central Bureau of Statistics, 2014.

[2] A Brief History of Smart Home Automation. https://zeusintegrated.com/blog/item/a-brief-history-of-smart-home-automation/.

[3] Majeed, R., Abdullah, N. A., Ashraf, I., Zikria, Y. B., Mushtaq, M. F., & Umer, M. (2020). "An intelligent, secure, and Smart Home Automation System". Scientific Programming.

[4] Feng, C. & Xie, H. (2020). "The Smart Home System Based on Voice Control". In: Kountchev, R., Patnaik, S., Shi, J., Favorskaya, M. (eds) Advances in 3D Image and Graphics Representation, Analysis, Computing and Information Technology. Smart Innovation, Systems and Technologies, vol 180 (pp. 383-392). Springer, Singapore.

[5] Alhawiti, K. (2015). "Advances in Artificial Intelligence Using Speech Recognition". World Academy of Science, Engineering and Technology, Open Science Index 102, International Journal of Computer and Information Engineering, 9(6), 1432-1435.

[6] "Firebase vs Heroku: What are the Differences?". Back4App Blog. https://blog.back4app.com/firebase-vs-heroku/.

[7] Heroku Postgres. https://devcenter.heroku.com/articles/heroku-postgresql/.

[8] Yokotani, T. & Sasaki, Y. (2016). "Comparison with HTTP and MQTT on required network resources for IOT". 2016 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC) (pp. 1-6). Bandung, Indonesia.

[9] Wang, C. (2018). "HTTP vs. MQTT: A tale of two IoT protocols". Google. https://cloud.google.com/blog/products/iot-devices/http-vs-mqtt-a-tale-of-two-iot-protocols/.

[10] Chakraborty, Chandralika & Talukdar, P.H. (2016). "Issues and Limitations of HMM in Speech Processing: A Survey". International Journal of Computer Applications, 141, 13-17.

[11] Panzner, M. & Cimiano, P. (2016). "Comparing Hidden Markov Models and Long Short Term Memory Neural Networks for Learning Action Representations". In: Pardalos, P., Conca, P., Giuffrida, G., Nicosia, G. (eds) Machine Learning, Optimization, and Big Data, vol 10122 (pp. 94-105). Springer, Cham.

[12] Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., & Inman, D. J. (2021). "1d convolutional neural networks and applications: A survey". Mechanical Systems and Signal Processing, 151, p. 107398.