International Journal of Sensors and Sensor Networks
2025, Vol. 13, No. 2, pp. 56-64
https://doi.org/10.11648/j.ijssn.20251302.14

Research Article

Design of a Framework for Switch Power Control Using Voice Signal

Blossom Oluwakorede Remi-Ofakunrin[1], Gabriel Babatunde Iwasokun[2, *], Edafe John Atajeromavwo[3], Raphael Olufemi Akinyede[4], Olufunso Alowolodu[5], Samuel Oluwatayo Ogunlana[6], David Bamidele Adewole[2], Ednah Olubunmi Aliyu[6]

[1]Department of Computer Science, Federal University of Technology, Akure, Nigeria
[2]Department of Software Engineering, Federal University of Technology, Akure, Nigeria
[3]Department of Data Science, Delta State University, Abraka, Nigeria
[4]Department of Information Systems, Federal University of Technology, Akure, Nigeria
[5]Department of Cybersecurity, Federal University of Technology, Akure, Nigeria
[6]Department of Computer Science, Adekunle Ajasin University, Akungba-Akoko, Nigeria

Abstract

Establishing systems that specifically control electric power switches based on the practical implementation of Artificial Intelligence in everyday life reduces the likelihood of accidental switch activation and potentially increases security by ensuring it responds only to authorised users. Individuals with physical disabilities also require systems devoid of direct human interventions and physical interactions to control electrical and power switches. Existing methods for achieving these tasks include smart objects, the Internet of Things, and biometric technologies, with their attendant strengths and weaknesses. This paper presents the design of a voice signal framework for remote control of power switches. The framework uses a voice sensor connected to an Arduino microcontroller to amplify the volume of the user's voice, while a voice sensor connected to a power switch relay is used to capture the voice signal for registration, training, verification and processing. The Arduino Nano 33 BLE Sense Rev 2 microcontroller sensor combines a tiny form factor with the capability to operate TinyML and TensorFlow Lite environment sensors while running at reconfigurable operating voltage. The switch relay regulates a high voltage to a minimum acceptable level based on integration with the Arduino microcontrollers. The framework also requires an external ESP8266/ESP32 Wi-Fi module to establish a connection between the microcontroller and the network as well as simple TCP/IP connections using Hayes-style commands. The system requires a power switch, an electromechanical device that uses the flow of electric current to open or close an electrical circuit. The user voice recognition is based on Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) networks. The combination of these two models guarantees an effective capturing of temporal dependencies in sequential data typical of audio signals.

Keywords

Remote Control, Power Switch, Switch Control, Voice Recognition, Arduino Microcontroller

*Corresponding author: gbiwasokun@futa.edu.ng (Gabriel Babatunde Iwasokun)

Received: 1 September 2025; **Accepted:** 10 September 2025; **Published:** 22 November 2025

1. Introduction

Power is the rate at which electrical energy is transferred or consumed in an electrical circuit. In common parlance, electric power is the production and delivery of electrical energy [1]. Existing types of power include direct current, alternating current, apparent power, active power and reactive power [2]. A power control system is a system that controls the output (production or discharging) and input (charging) of one or more power sources, such as photovoltaic systems, batteries, and electric vehicles. It plays a crucial role in maintaining the integrity and normal operation of power systems, especially with the increasing integration of renewable energy sources and distributed generators. A power control system utilizes various control measures, modelling, plans, and safety arrangements to function. It is important for sustaining the stability, efficiency, and performance of various systems. Power control strategies are applied in grid-connected power converters, home appliances, electric vehicles and renewable energy systems. Power switch control is also a vital aspect of modern electrical systems, combining traditional methods with cutting-edge technology to enhance usability, security, and efficiency. It encompasses various methods and technologies to manage the flow of electricity to devices or systems and is based on the principles of electrical basics and circuit design. Its types include manual switches, electromagnetic relays, transistor-based switches, smart switches, timers and programmable switches, home automation systems and remote control [3-7]. Different forms of power control systems have been established for regulating the output and input of various power sources and guaranteeing the safety and efficiency of power systems. Manual power control encompasses unswerving human involvement or bodily tuning to regulate the power output or input of a system. Though common and straightforward, manual power control can be time-consuming and less responsive to dynamic changes in power demand or supply. Remote power control utilises technology on its own to enable the control and monitoring of power systems from a distance. It promotes real-time adjustments and automation of power control processes and enhances efficiency and responsiveness. Remote power control systems often utilize communication networks, sensors, and actuators for remote monitoring and control of power sources, making them ideal for applications where physical access is limited or impractical. Several biometric signals, including voice, could be used to provide a secure and more convenient way to manage electrical switches [8, 9].

Voice signals could be captured using microphones based on the conversion of the sound pressure waves into electrical signals. Its pre-digitisation tasks involve amplifying and filtering the raw signal to reduce or suppress unwanted noise or interference [10]. Various studies on voice or signal processing as the basis for remote power or switch control have contributed immensely to current and power usage optimality, formulation of multi-tier architecture for simple circuits with unique inter-connections, transmission of uplink signal, separation of speech from non-speech segments in audio signals and development of voice-controlled devices. The limitations of the various studies include a lack of consideration for privacy and security, failure to establish practical functions, delayed response, accuracy issues arising from noisy environments, computational complexity, failure with larger and more diverse activities and restrictions in terms of language support and complexity of setup and maintenance. The limitations present some research gaps, and the need to fill some of them strongly motivated this research.

2. Literature Review

Yang et al. [11] presented a voice control system that encompasses voice encoding, display, and processing modules. While the voice encoding module is used to analyze and process voice signals to determine their source and response information, the processing module controls the display to rotate towards the sound direction and transmits the response information for display. A practical study of the system confirmed its ability to exercise control over different types of information through voice commands as well as its lack of consideration for privacy and security. The authors in [12] shed light on the strengths and weaknesses of the existing power control systems in addition to examining the variations in power electronics that incorporate different types of power converters. Special focus was placed on the technical and theoretical framework supporting power switch and converter control techniques like hysteresis, sliding mode, predictive and artificial intelligence. The study established the state of the existing control systems alongside their features, block diagrams and vectors, but failed to establish their practical functions.

In [13], a low-power Spiking Continuous Time Neuron (SCTN) model for sound signal processing was proposed. The model is based on accurate classification of sound signals using the Spiking Neural Network (SNN) and Real-World Computing Partnership (RWCP). The implementation of the SCTN-based resonators for sound feature extraction demonstrated high efficiency, while the integration of the preprocessing phase into the network allows the continuous processing of the audio signal, thereby eliminating external preprocessing and time-frequency representation of the sound. The adoption of low-power SCTN as a basic building block of the model promotes simple analogue circuits with unique interconnections between the neurons. The experimental study on the practical function of the model was not established. In [14], a power control system based on the optimization of the transmitting power of the uplink sounding reference signal (SRS) at the transmitting channel and the dif-

ference between the losses of the signal receiving and transmitting channels was proposed. The power control apparatus consists of a signal-receiving channel, an uplink-sounding reference signal (SRS) transmitting channel, a radio frequency transceiver, and a processor. The signal-receiving channel receives signals from external sources, which the uplink-sounding reference signal transmitting channel transmits to the SRS signals. A radio frequency transceiver then facilitates the transmission and reception of the signals and adjusts the transmitting power accordingly. A study into the model established its usefulness for adjusting the transmitting power of an uplink-sounding reference signal (SRS) based on the difference between the loss of the signal receiving channel and the loss of the SRS transmitting channel. The authors in [15] presented an ultra-low-power voice activity detection model using level-crossing sampling. The model uses level-crossing sampling to discriminate speech from non-speech parts of audio signals and the ultra-low-power voice activity detection (VAD) method to achieve average speech and non-speech hit rates. A study on the model showed its ability to achieve a power-efficient and accurate separation of speech from non-speech segments in audio signals. The study also showed the computational complexity of the model and its failure with larger and more diverse voice activity detection.

A voice-based automated control framework for electrical devices was presented in [16]. The framework utilizes voice signal processing and acoustic and language modellings for speaker recognition. Its voice signal processing component is based on Automatic Speech Recognition (ASR) technology, which requires a microphone, speech recognition software, and a computer to transcribe spoken language into written text. The framework also uses sensors to decode voice signals and a microprocessor to translate the decoded signals for executing specific commands. It is suitable for use in Android applications, Arduino Mega boards, Bluetooth modules, microcontrollers, and relays, though some quantitative analyses on performance and effectiveness are still required. A voice-based model for device control is presented in [17]. The model generates sound fields and controls sound images in specific spaces as well as introduces speaker arrays and wave field synthesis to enhance the listening experience for users while minimizing sound leakage. The various components of the model include a voice signal input unit, frequency determination unit, band controller, sound image controller, and voice output unit. The model also uses wave field synthesis, frequency determination, sound image control, and adjusting reproduced sound based on noise levels to transmit the sound images and output sound signals to different speakers based on frequency bands and cutoff frequencies. The implementation of the model showed it is suitable for use in the development of voice control devices and systems that can effectively generate sound fields and control sound images in specific spaces, such as aircraft cabins. The authors in [18] proposed a framework for a Smart Home System with Voice Control Using NLP methods. The framework is based on the human-machine interface for smart home systems with the incorporation of speech recognition for remote monitoring and management. There is an addition of utterance to command transformation of existing cloud-based speech-to-text and text-to-speech services, to achieve greater flexibility and adaptation for various automation systems and consumer electronics. The framework also adopts the use of statistical features, neural networks, deep learning, and other intelligent methods for intent detection and semantic recognition of voice commands. The experimental study of the framework justified its support for under-resourced languages and automatic intent recognition as well as its ability to function as a free alternative to existing paid online natural language understanding (NLU) services. The study also revealed the stringent reliance of the framework on cloud-based speech-to-text and text-to-speech services, which may have limitations in terms of language support and complexity of setup and maintenance. In [19], the prospect of signal processing based on Active Noise Control (ANC) was presented. The research presented a systematic review of ANC technology evolution over the past quarter-century and the application of signal processing to the ANC. A summary of the main application areas of ANC technology, the technical bottlenecks, the opportunities and outlook on future developments was presented.

3. Proposed Framework

The proposed system requires voice commands to control an electrical power switch. It uses voice activity detection for the processing of the voice signal as well as different sensors for accepting the voice signal and Wi-Fi-enabled remote control of the switch. The architecture of the proposed system is presented in Figure 1, showing the basic functionalities. A voice sensor will be connected to the microcontroller to amplify the volume of the user's voice, while a voice sensor that is connected to a relay and the power switch will read the pre-registered, pre-trained and verified voice command for processing. User's voice recognition will be based on Recurrent Neural Networks (RNNs) with LSTM networks. LSTMs are particularly well-suited for this task because they can effectively capture temporal dependencies in sequential data typical of audio signals.

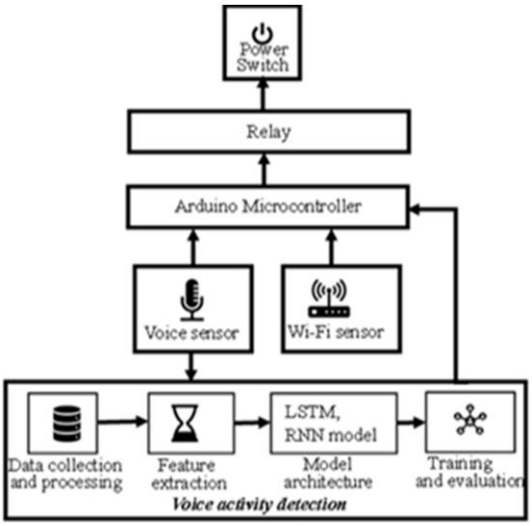

Figure 1. *Architecture of the proposed system.*

3.1. Voice Sensor with Microphone

The voice sensor shown in Figure 2 is a compact and easy-to-use voice recognition module designed for embedded systems, which can be trained to recognise voice commands and respond accordingly. Its voltage requirement ranges between 4.5V and 5.5V, and its optimal current specification is 50mA.

Figure 2. *Microphone-fitted voice sensor.*

Shown in Figure 3 is an Arduino Nano 33 BLE Sense Rev 2 microcontroller that is required for connecting the voice sensor. It combines a tiny form factor with the capability to operate TinyML and TensorFlow Lite environment sensors while running at 3.3V to its analogue and digital pins.

Figure 3. *Arduino Nano 33 BLE sense Rev 2.*

Figure 4. *A Relay.*

A typical 5V electromechanical relay rated for 10A/250VAC switches, shown in Figure 4, will act as a switch between the connecting devices. It is integrated with Arduino microcontrollers and will be used for the regulation of high voltage to a safe level for the devices. It will also be used for setting the electrical devices to the on and off modes.

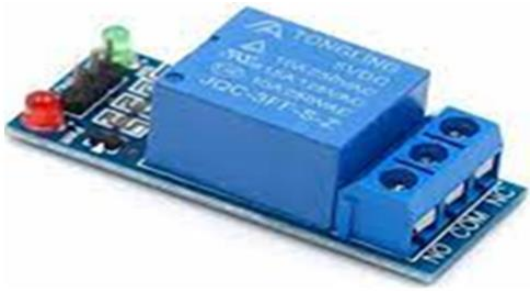

Figure 5. *Wi-Fi module.*

Figure 5 presents a typical ESP8266/ESP/32 Wi-Fi module that will be required to establish the connection between the microcontroller and the network, as well as simple TCP/IP connections using Hayes-style commands on a voltage range of 3.0V to 3.6V. The system will also operate on a power switch, which is an electromechanical device designed to use the flow of electric current to open or close an electrical circuit.

3.2. RNN and LSTM Voice Activity Detection (VAD)

The voice activity detection (structure shown in Figure 6) will be based on RNNs and LSTMs. These two models are needed to achieve low data and computational requirements, which are lacking in other models, like CNNs. VAD begins with the audio recording using the voice sensor and is followed by the application of a Wiener filtering spectral subtraction (WFSS) denoising technique as shown in Figure 7.

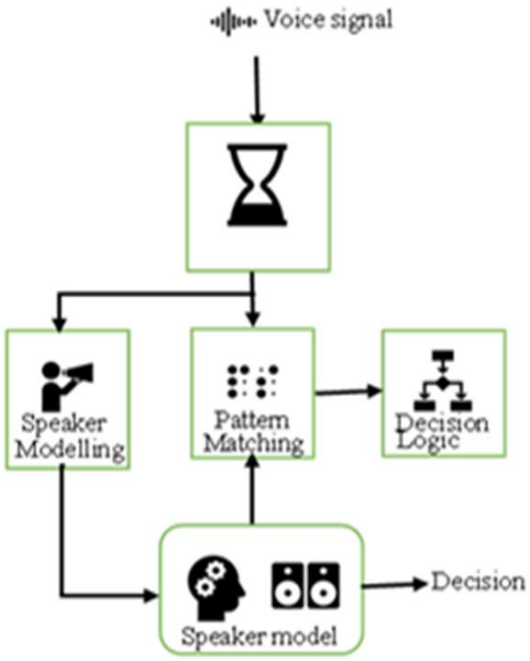

Figure 6. *Voice recognition structure.*

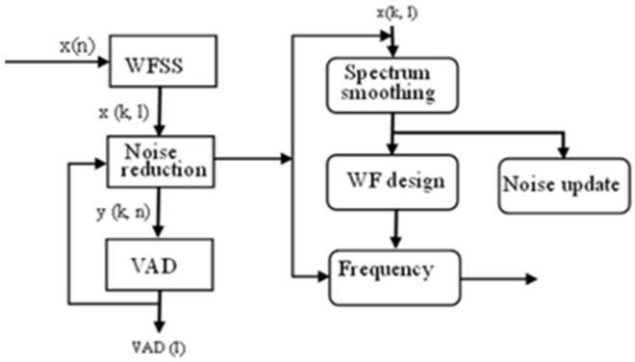

Figure 7. *Block diagram of the denoising process.*

TThe extraction of the Mel-frequency cepstral coefficients (MFCCs) or other relevant features from the denoised audio signal is then performed, and the resulting data is partitioned into training and testing sets. The estimation of MFCCs from speech involves the division of speech into short segments, estimation of the power spectrum of each segment, application of the Mel filter bank on the power spectra, summation of the energy for every filter, obtaining the logarithm of the filter bank energies, calculation of the DCT of the logarithms and taking the coefficients for every segment. The estimation of MFCCs is followed by the computation of the spectrogram based on Short-Time Fourier Transform (STFT) through the segmentation of the signal into segments of fixed length, and then the application of a window with some overlap.

he spectrogram is the squared magnitude of the STFT. If the STFT of the signal is $x(n), w(n)$ is the window and $S(\tau, k)$ is the spectrogram. The spectrum can be extracted as a slice of the spectrogram based on the formula [20]:

$$X(\tau, k) = STFT\{x(n)\} = \sum_{N=0}^{N-1} x(n)w(n-\tau)e^{-jnk} \quad (1)$$

$$S(\tau, k) = |X(\tau, k)|^2 \quad (2)$$

The partitioning of the resulting data also involves the following:

Dataset Definition: Let D be the entire dataset comprising N samples, such that:

$$D = \{(x_1, y_1)(x_2, y_2) \cdots (x_N y_N)\} \quad (3)$$

x_i is the input features and y_i is the corresponding label.

Split Proportion: A split is defined in the ratio α where $0 < \alpha < 1$. Characteristically, α is set to 0.8, meaning 80% of the data is used for training and 20% for testing. The number of training samples is $N_{train} = \lfloor \alpha N \rfloor$ and the number of testing samples is $Ntest = N - N_{train}$.

Random Shuffling: The dataset D is shuffled to ensure that the training and testing sets are representative and devoid of biases in the order of the data.

Index Assignment: The training set D_{train} and testing set D_{test} are created by selecting the first N_{train} samples and the remaining N_{test} samples after shuffling.

$$D_{train} = \{(x_{\pi 1}, y_{\pi 1})(x_{\pi 2}, y_{\pi 2}) \cdots (x_{\pi N_{train}} y_{\pi N_{train}})\} \quad (4)$$

$$D_{test} = \{(x_{\pi_{N_{train}}+1}, y_{\pi_{N_{train}}+1})(x_{\pi_{N_{train}}+2}, y_{\pi_{N_{train}}+2}) \cdots (x_{\pi_N}, y_{\pi_N})\} \quad (5)$$

The denoising operation involves spectrum smoothing, noise estimation and the design of a Wiener filter as shown in Figure 6. The spectrum smoothing is based on the average of the power spectrum over two consecutive frames and two spectral bands. The noise $Ne(k)$using a 1*st* order IIR filter based on the smoothed spectrum $Ys(k, l). Ne(k)$ is obtained from:

$$Ne(k) = \lambda Ne(k) + (1 - \lambda) Ys(k, l) \quad (6)$$

For the design of the Wiener filter (WF), the clean signal $S(k)$ is estimated by spectral subtraction:

$$S(k, l) = X\beta Sl\,(k, l) + (1 - X\beta) max(Ys(k, l) - Ne(k), 0) \quad (7)$$

The Wiener filter $H(k)$ is calculated from:

$$\eta(\mathrm{k}, \mathrm{l}) = \max\left[\frac{\mathrm{S(k,l)}}{\mathrm{Ne(k)}}, \eta_{\min}\right] \quad (8)$$

$$\mathrm{H(k, l)} = \frac{\eta(\mathrm{k,l})}{1+\eta(\mathrm{k,l})} \quad (9)$$

$\eta_{\min}$ is selected so that the filter yields a maximum attenu-

ation and $S^i(k,l)$ will be assumed to be zero at the beginning of the process, and defined thus:

$$S^i(k,l) = max[Y(k,l)H(k,l),16] \quad (10)$$

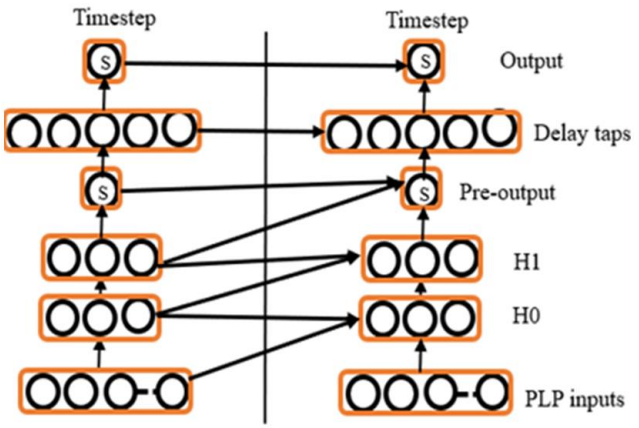

Figure 8. *Feed-forward NN architecture with recurrence added at various points.*

The filter $H(k,l)$ is smoothed to eliminate rapid changes between neighbour frequencies that may seldom cause noise.

3.3. Recurrent Neural Network

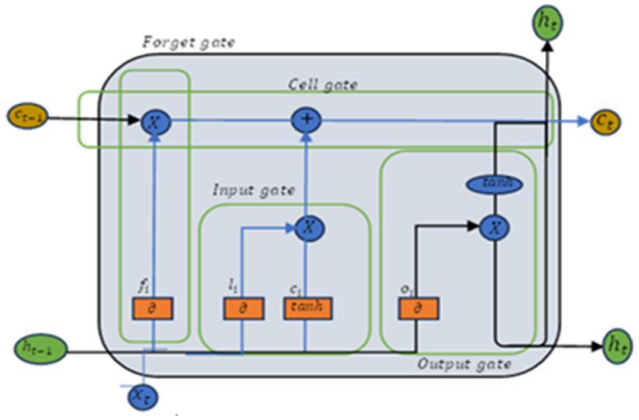

Figure 9. *The LSTM architecture.*

RNNs are parameterizable models representing computation on data sequences. In the likeness of feed-forward neural networks (NNs), which model stateless functions over $Rm \rightarrow Rn$, an RNN's computation is factored into nodes, each of which evaluates a simple function mapping its input values to a single scalar output. Feed-forward NN architecture with recurrence added at various points is presented in Figure 8. Unlike NNs, RNN nodes can receive input from nodes at previous time steps, which allows them to store and manipulate state as they iteratively process sequences of inputs and generate a series of outputs. Instead of the traditional weighted sum and non-linear activation of a multi-layer perceptron (MLP), the RNN nodes compute quadratic functions of their inputs, followed by an optional non-linearity performed as follows:

$$V(x) = f(x^T W_Q x + wLTx + w_B) \quad (11)$$

A node computes its output value $V(x)$ from the vector x of its inputs using Eq. (11); W_Q is an upper-triangular sparse matrix with weights for quadratic terms, w_L is a vector of linear weights similar to those in MLPs, and w_B is a scalar bias. The reason behind this approach is the idea that higher-order Taylor polynomials can reasonably approximate more functions than affine functions. This representation can compute products, similar to the Multiplicative RNNs, and such nodes can also evaluate the multidimensional Gaussian density (and other radial basis functions), since $N(x; \mu, \Sigma)$ can be written as $e^{(-x^T\Sigma^{-1}x + 2\mu^T\Sigma^{-1}x - \mu^T\Sigma^{-1}\mu + \ln(z))}$. z is the Gaussian normalization constant.

3.4. Long Short-Term Memory (LSTM)

The LSTM is a form of Recurrent Neural Network (RNN) that analyzes short and long-term data. Its design consists of several cells, each with three primary parts that are in charge of updating, remembering, and forgetting information [21]. The LSTM modules are independent, and they use a sigmoid gate known as the "forget gate", f_t to know if any information needs to be erased from the c_{t-1}cell.

The gate can generate several numbers in the range of 0 and 1 for every part in c_{t-1}consequent to reading the values $h_{t-1}$and x_t. The forget gate and the element-wise sum formula for the gate are as follows:

$$f_t = \sigma(Wf.[h_{t-1}, x_t] + bf \quad (12)$$

$$i_t = \sigma(Wi.[h_{t-1}, x_t] + bi \quad (13)$$

$$\widehat{C}_t = tanh(Wc.[h_{t-1}, x_t] + bC \quad (14)$$

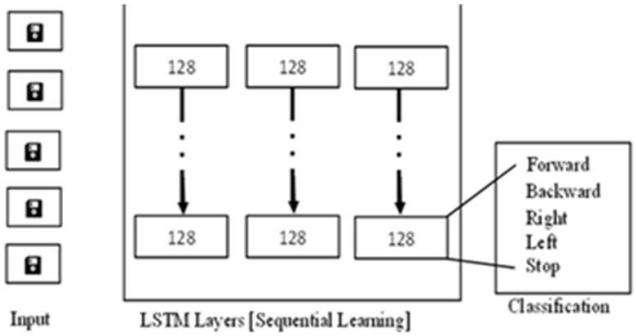

Figure 10. *Architecture of LSTM for speech recognition.*

The last part of the LSTM is the output gate (neuron layer

with the sigmoid activation function at the far right of the neuron layer line) [22]. Its output does not contribute to the state of the cell, but the gate is required to differentiate the cell state and the actual output [23]. The LSTM is derived according to the following:

$$o_t = \sigma\,(Wo.[h_{t-1}, x_t] + bo \tag{15}$$

$$h_t = o_t.tanh\,(Ct) \tag{16}$$

Figure 9 illustrates the workflow of an LSTM (Long Short-Term Memory) neural network used for sequence learning and classification, and its voice recognition equivalent is shown in Figure 10. The leftmost section shows a sequence of input data frames which could represent a time series of data, images, audio or any sequential data that is to be processed by the LSTM network. The central section comprises multiple LSTM layers, each containing 128 units (neurons) that are designed to process the sequential input data and capture both long-term and short-term dependencies within the sequence. The connections between the layers indicate the flow of information through the network, with each layer passing processed information to the next. The rightmost section gives the classification output of the LSTM network. The processed data from the final LSTM layer is fed into a classification mechanism that assigns one of the five possible classes, namely Forward, Backwards, Right, Left, and Stop. Each class is represented by a blue dot, indicating the possible actions or outcomes that the model could predict. A dropout layer is added between LSTM layers to prevent overfitting. The layer randomly sets a fraction of input units to 0 at each update during training, which helps prevent overfitting. It also randomly sets a fraction of input units to zero at each update during training time, which helps prevent overfitting. Dropout is a regularization technique that randomly sets a fraction of input units to zero during training to prevent overfitting. At the training phase, each neuron's output is set to zero with a probability p (dropout rate) while the remaining neurons' outputs are scaled by $\frac{1}{1-p}$ to keep the expected sum of inputs constant. Given that x is the input vector to the dropout layer and p is the dropout rate (probability of dropping a neuron), then r_{i} is the Bernoulli random variable that is 1 with probability 1−p and 0 with probability:

$$y_i = r_i \frac{x_i}{1-\mathrm{p}} \tag{17}$$

$$r_i \sim Bernoulli(1-p).$$

A dense layer is also required for interpreting the learned sequence patterns and performing the final classification with ReLU activation function. The output layer is expected to exhibit a single neuron with a sigmoid activation function for binary classification (speech or non-speech). Given that h is the input to the dense layer (output from the LSTM layer or the previous dense layer), W is the weight matrix of the shape (n, m) where n is the number of input units, and m is the number of output units, b is the bias vector of the shape (m) and ϕ is an activation function (such as ReLU, sigmoid, tanh), then the output y of the dense layer is given in [24]:

$$y = \phi(Wh + b) \tag{18}$$

Model training involves minimizing the loss function by updating the model parameters (weights and biases) using an optimization algorithm. The general process involves a forward pass and loss calculation. In the forward pass, the predicted output y is computed using the current parameters of the model as follows:

$$y = f(X, w) \tag{19}$$

X is the input data and w is the model parameters. During loss calculation, the model is trained to differentiate between speech and non-speech in the context of voice activity detection. Binary cross-entropy is used to measure the performance of the classification model whose output is a probability value between 0 and 1, and it is calculated from:

$$\mathrm{L}= -\frac{1}{N}\sum_{i=1}^{N} y_i \log(p_i) + (1-y_i)\log(1-p_i) \tag{20}$$

N is the number of samples, y_i is the true label of the i^{th} sample (0 or 1), p_i is the predicted probability that the i^{th} sample belongs to the positive class (output of the sigmoid activation function) [22]. Consequent to this operation is the backward pass and parameter update. In the backwards pass, the gradients of the loss function with respect to the model parameters, $\nabla_w L(w)$ is calculated while updating the model parameters is based on the Adam optimizer rules presented thus:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\nabla_w L(w) \tag{21}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)(\nabla_w L(w))^2 \tag{22}$$

$$\widehat{m_t} = \frac{m_t}{1-\beta_1^t} \tag{23}$$

$$\widehat{v_t} = \frac{v_t}{1-\beta_2^t} \tag{24}$$

$$w \leftarrow w - \frac{\eta}{\sqrt{\widehat{v_t}}+\epsilon}.\widehat{m_t} \tag{25}$$

m_t and v_t are the first and second-moment estimates, respectively, β_1 and β_2 are decay rates, typically set to 0.9 and 0.999, respectively, and $\widehat{m_t}$ and $\widehat{v_t}$ are bias-corrected estimates [25].

4. Conclusions

The paper presents the design of a voice-based framework for remote control of power switches. The framework will be suitable for the remote and contactless operation of power switches and ultimately eliminate the event of accidental switch activation, increase security by guiding against authorized users and enable individuals with physical disabilities to effortlessly operate electrical and power switches. The implementation of the framework is ongoing, with Python and Java providing the programming terrains while MySQL provides the platform for the creation and management of the template and reference databases. The choice of MySQL is premised on its open-source nature, ease of use, scalability, high performance and strong communal support. The programming terrains will adopt Java Database Connection for communication with the resources at the database level.

Abbreviations

RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
SCTN	Spiking Continuous Time Neuron
SNN	Spiking Neural Network
RWCP	Real-World Computing Partnership
SRS	Sounding Reference Signal
VAD	Voice Activity Detection
NLP	Natural Language Processing
ANC	Active Noise Control
WFSS	Wiener Filtering Spectral Subtraction
MFCC	Mel-Frequency Cepstral Coefficient
DCT	Discrete Cosine Transform
STFT	Short-Time Fourier Transform
TETFund	Tertiary Education Trust Fund

Acknowledgments

The noble role played by the Federal University of Technology, Akure, Nigeria's Centre for Research and Development towards the success of this research is greatly acknowledged.

Author Contributions

Blossom Oluwakorede Remi-Ofakunrin: Conceptualization and Methodology

Gabriel Babatunde Iwasokun: Conceptualization, Methodology, Project administration, Resources, Writing, Preparation of original draft.

Edafe John Atajeromavwo: Methodology, Project administration.

Raphael Olufemi Akinyede: Conceptualization, research administration, writing, review and editing of original draft.

Olufunso Alowolodu: Research administration, review and editing of original draft.

Samuel Oluwatayo Ogunlana: Conceptualization and Methodology

David Bamidele Adewole: Conceptualization and Methodology

Ednah Olubunmi Aliyu: Conceptualization and Methodology

Funding

This research was funded by the Nigerian government's 2023 Research Grant through the National Tertiary Education Trust Fund (TETFund).

Data Availability Statement

Not applicable.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Hambley A. (2025), Electrical Engineering, https://easypdfs.cloud/downloads/4922968-Allan%20Hambley%20Electrical%20Engineering

[2] Glover J. D., Thomas J. O., Mulukutla S. S. (2022), Power System Analysis and Design, Cengage Learning, 7th Edition.

[3] Wang Z., Defang L., Yunan S., Xiaoyi P., Feng L., John C. S. L., and Kui R. (2022), A Survey on IoT-Enabled Home Automation Systems: Attacks and Defenses, IEEE Communications Surveys and Tutorials, 24(4).

[4] Neha M., and Yogita B. (2017), Literature Review on Home Automation System, International Journal of Advanced Research in Computer and Communication Engineering, 6(3).

[5] Wanzala J. N. and Atim M. R. (2024), Design and simulation of a smart master switch system based on multi-input XOR logic gate, Discover Electronics, 1: 23.

[6] Subramaniam K., Husin S. H., Anas S. A. and A. H. Hamidon (2014), Multiple Method Switching System for Electrical Appliances using Programmable Logic Controller, WSEAS Transactions on Systems and Control, 4(6).

[7] Sakshi S., Manish K. M., Nisha D. (2023), Home Automation System, International Journal of Novel Research and Development, 8(1); 96-100.

[8] Abe B. C., Araromi H. O., Shokenu E. S., Idowu P. O., Babatunde J. D., Adeagbo M. O., Itanrin H. O. (2022), Biometric Access Control Using Voice and Fingerprint, Engineering and Technology Journal, 7(7).

[9] Yadav H., Bansal U. (2021), A Novel Low-Voltage Low Power FGMOS and CMOS Resister Current Mirror.

[10] Schafer R., and Rabiner L. (2011), Real-Time Digital Hardware Pitch Detector. IEEE Transactions on Acoustics, Speech, and Signal Processing, 24(1), 2-8.

[11] Yang C. H., Gu Y., Liu Y. C., Ghosh S., Bulyko I., Stolcke A. (2023), Generative speech recognition error correction with large language models and task-activating prompting, in: 2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), IEEE, 1-8.

[12] Al-Ogaili A., Ramasamy A., Juhana T., Tengku H., Al-Masri A., Hoon Y., Jebur M., Verayiah R., Marsadek M. (2020). Estimation of the Energy Consumption of Battery-Driven Electric Buses by Integrating Digital Elevation and Longitudinal Dynamic Models: Malaysia as a Case Study. Applied Energy. 280. https://doi.org/10.1016/j.apenergy.2020.115873

[13] Bensimon, M., Greenberg, S., and Haiut, M. (2021). Using a Low-Power Spiking Continuous Time Neuron (SCTN) for Sound Signal Processing. *Sensors*, *21*(4), 1065. https://doi.org/10.3390/s21041065

[14] Wang B. (2021), Power Control Apparatus and Method, and Electronic Device.

[15] Maral F., Hamidreza R., Nassim R. and Hamed A. (2023). Ultra-Low-Power Voice Activity Detection System Using Level-Crossing Sampling, Electronics, 12(4): 795.

[16] Amannah C. I. and Nlerum P. (2022). Voice-Based Automation Control Platform for Home Electrical Devices, Available: https://www.researchgate.net/publication/340405809_Voice-Based_Automation_Control_Platform_for_Home_Electrical_Devices

[17] Junji A., Satoshi A., Takahiro Y. and Kenichi, K. (2022). Voice Control Device and Voice Control System.

[18] Iliev, Y., and Ilieva, G. (2023). A Framework for Smart Home System with Voice Control Using NLP Methods. *Electronics*, *12*(1), 116. https://doi.org/10.3390/electronics12010116

[19] Dongyuan S., Bhan L. and Woon-Seng G., (2023). Active Noise Control in the New Century: The Role and Prospect of Signal Processing, Internoise, Available: https://arxiv.org/abs/2306.01425

[20] Samia D. S. M., Bessa E., Blumstein D. T., Nunes J. A. C. C., Azzurro E., Morroni L., Sbragaglia V., Januchowski-Hartley F. A., and Geffroy B. (2019). A Meta-Analysis of Fish Behavioural Reaction to Underwater Human Presence. *Fish and Fisheries*, 20, 817-829.

[21] Hajiaghayi M., Vahedi E. (2019). Code Failure Prediction and Pattern Extraction Using LSTM Networks. 55-62. https://doi.org/10.1109/BigDataService.2019.00014

[22] Graves A., Mohamed A. R., and Hinton, G. (2013), Speech Recognition with Deep Recurrent Neural Networks. 2013, Available: https://arxiv.org/abs/1303.5778

[23] Srivastava N., Hinton G., Krizhevsky A., Sutskever I., and Salakhutdinov R. (2014), Dropout: A Simple Way to Prevent Neural Networks from Overfitting, Journal of Machine Learning Research 15, 1929-1958.

[24] Kingma, D. P., and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *CoRR, abs/1412.6980*.

[25] Yuliy I. and Galina I. (2022). A Framework for Smart Home System with Voice Control Using NLP Methods, *Electronics* 2023, *12*(1), 116; https://doi.org/10.3390/electronics12010116

Research Field

Blossom Oluwakorede Remi-Ofakunrin: Signal processing, biometric authentication, system security

Gabriel Babatunde Iwasokun: Software engineering, artificial intelligence, signal processing

Edafe John Atajeromavwo: Artificial intelligence

Raphael Olufemi Akinyede: System security, cybersecurity, cloud computing

Olufunso Alowolodu: Cybersecurity, cloud computing

Samuel Oluwatayo Ogunlana: biometric security, signal processing

David Bamidele Adewole: software engineering, artificial intelligence.

Ednah Olubunmi Aliyu: Soft computing