BAREKENG: Journal of Mathematics and Its Applications
June 2025 Volume 19 Issue 2 Page 0973–0986
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol19iss2pp0973-0986

# OVERCOMING OVERFITTING IN MONKEY VOCALIZATION CLASSIFICATION: USING LSTM AND LOGISTIC REGRESSION

**Suryasatriya Trihandaru[1], Hanna Arini Parhusip[2*], Abdiel Wilyar Goni[3]**

[1,2,3]*Master of Data Science Study Program, Faculty of Science and Mathematics, Universitas Kristen Satya Wacana*
*Jln. Diponegoro 52-60, Salatiga, 50711, Indonesia*

*Corresponding author's e-mail: * **hanna.parhusip@uksw.edu***

***Article History:***

*Received: 18th August 2024*
*Revised: 30th January 2025*
*Accepted: 25th February 2025*
*Published: 1st April 2025*

***Keywords:***

*MFCC;*
*Classification;*
*LSTM;*
*Logistic Regression.*

*ABSTRACT*

*The problem of overfitting in a classification task involving animal vocalizations, namely squirrel monkeys, golden lion tamarins, and tailed macaques, is handled in this project. Acoustic features extracted for the audio data used in this research are MFCCs. The classification of subjects was done using the LSTM model. However, several architectures with LSTM also presented the problem of overfitting. To overcome this, a logistic regression model was used, which had a classification accuracy of 100%. These results indicate that for such a classification problem, logistic regression may be more appropriate than the complex architecture of LSTMs. Several LSTM architectures have been presented in this study to give an overall review of the observed challenges. Although the capability of LSTM in handling sequential data is very promising, sometimes simpler models might be preferred, as indicated by the results. This is a single-dataset work, and the findings may not generalize well to other domains. The work contributes much-needed insight into the choice of models for audio classification tasks and identifies the trade-off between model complexity and performance*



***How to cite this article:***

S. Trihandaru, H. A. Parhusip and A. W. Goni., "OVERCOMING OVERFITTING IN MONKEY VOCALIZATION CLASSIFICATION: USING LSTM AND LOGISTIC REGRESSION," *BAREKENG: J. Math. & App.*, vol. 19, iss. 2, pp. 0973-0986, June, 2025.


Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

**Research Article · Open Access**

## 1. INTRODUCTION

Speech recognition is currently one of the needs in the development of various activities today. The development of virtual assistants such as Siri, Google Assistant, and Alexa allows us to make calls, send messages, search for information, and control smart home devices with just voice [1]. Similarly, cars with a navigation system with voice commands, hands-free phone calls, and voice-based car entertainment controls make driving safer and more comfortable [2]. Nowadays, there are also smart home devices: lights, thermostats, and other devices that can be controlled through voice commands, creating smarter and more automated homes [3]. However, these devices are made to interact more with human life.

While speech recognition technologies have been highly explored in human interactions, the application of these technologies in recognizing animal sounds is relatively unexplored, especially for rare species such as monkeys. To the best of the authors' knowledge, there is a lack of comprehensive studies focusing on the automatic recognition and classification of monkey vocalizations using advanced techniques such as Long Short-Term Memory (LSTM) [4] and Mel-Frequency Cepstral Coefficients (MFCC). Besides, this gap highlights the novelty of this research, intending to bridge this domain by proposing methods that monitor the vocalization of rare animals to support wildlife conservation efforts. Several studies have been conducted using MFCC and LSTM in voice recognition. One of them is research by [5], which uses LSTM for speech emotion recognition in Tamil with an accuracy rate of 84%. Other studies use MFCC for speech recognition, resulting in an accuracy of 88.21% [6]. Other research related to the use of LSTM for speech emotion recognition using MFCC results in 89% accuracy [7]. This research aims to build a system that can recognize monkey voices based on voice recognition and deep learning with the MFCC algorithm as a voice feature extraction and LSTM and logistic regression as the training algorithms.

The novelty of this research lies in both aspects: 1) monkey vocalization recognition and 2) classification methods for monkey vocalization recognition. First, the study will be concerned with the less explored area of recognizing and classifying the vocalizations of the rare species of monkeys, such as Squirrel Monkeys, Golden Lion Tamarins, and Tailed Macaques. In this manner, the focus on these species contributes to the continuously developing bioacoustics and wildlife monitoring field. This includes investigating and comparing advanced methods for the classification task: Long Short-Term Memory versus logistic regression on the considered problem. The resulting curve showing model complexity versus performance serves as a lesson on the relevance of choosing an appropriate method, given the different characteristics of different datasets and the specific tasks one may be asked to perform. The dual foci of their work underline its contribution toward wildlife conservation as well as methodology in machine learning.

Squirrel Monkeys, Golden Lion Tamarin, and Tailed Macaque were selected since their vocalizations are fundamentally different in nature from one another, making them perfect test subjects for performance evaluation of various classification methods. These species were selected because they insert enough variability into the dataset, enabling one to have a more robust analysis of the capability of the proposed system to distinguish between the unique features of every sound. The sources of information included recordings of vocalizations that were obtained from online audio repositories and publicly available datasets. Details are mentioned in subsequent sections of the manuscript. These sources are preferred because they are very reliable and available; hence, they have high-quality recordings suitable for feature extraction and classification tasks. We will revise the manuscript to incorporate this explanation within the introduction so that, from the outset, it clearly outlines the rationale and the source of data to ensure greater clarity and coherence.

Computing developments can help humans identify monkey behavior based on vocal communication with technology, deep learning, and voice recognition. Voice recognition is a subfield of signal processing that can recognize or distinguish objects from sounds based on patterns and characteristics of sounds. Sound feature extraction is a stage to obtain sound information that is used to distinguish one sound from another. The recognition of monkey sounds, for example, is studied by paying attention to the context of affiliation, cohesion, agonism, predation, and mating in an animal voice [8]. The Mel-Frequency Cepstral Coefficients (MFCC) is one of the methods to convert sound signals in the time domain to the frequency domain using the Fast Fourier Transform, then converts the sound signal into a scale. Mel uses Bank Filter and gets a grade cepstrum, which is in the form of a matrix [9]. After the sounds are converted into the time domain to frequency, the Long Short-Term Memory is employed to do the classification.

Overfitting is considered in research work that employs complex models, such as Long Short-Term Memory (LSTM). The complexity in the LSTM architecture easily leads to overfitting with small datasets or

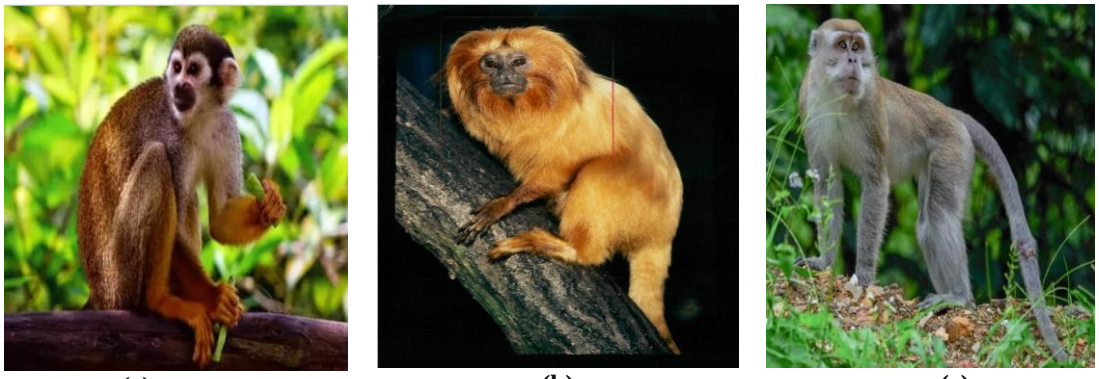
where there is not much variability. For this purpose, different approaches were used: dropout regularization, tuning hyperparameters, and the use of simpler models like logistic regression. We show that the logistic regression is very capable of generalizing well and reaches 100% accuracy in this work.

## 2. RESEARCH METHODS

In this article, the use of MFCC and LSTM technology as a means of recognizing the sound of a monkey is covered. At first, a collection of seven different types of sounds made by monkeys during their communication was included in the dataset. But for the first stages of training, the algorithms employed LSTM and logistic regression, and only three types of monkey sounds were chosen. Specifically, Squirrel Monkeys, Golden Lion Tamarin, and Tailed Macaques were used as examples for MFCC and LSTM modeling and recognition. The three types of apes are illustrated in **Figure 1**. Each audio signal, which possesses unique spectral properties, is extracted by the essential features using the MFCC technique. These features are then classified using LSTM models, a family of models used in the processing of sequential data. However, challenges such as overfitting call for the use of simpler models like logistic regression, which still performs better in this work. These findings would contribute to the wider area of automatic wildlife monitoring research and bioacoustics studies.

### 2.1 Data Source

In this section, we analyzed a dataset consisting of monkey vocalizations sourced from YouTube videos. This dataset was used to investigate the effectiveness of logistic regression models in classifying these vocalizations. While the results demonstrated promising accuracy, the potential for overfitting was not explicitly addressed in this study. A more in-depth exploration of overfitting mitigation techniques, such as regularization or cross-validation, would be beneficial for improving the robustness of the model.



**(a)** **(b)** **(c)**

**Figure 1. The types of monkeys: (a) Squirrel Monkey, (b) Golden Lion Tamarin, and (c) Long-Tailed Macaque**

### 2.2 Mel-Frequency Cepstral Coefficients (MFCC)

Mel-Frequency Cepstral Coefficients (MFCC) is a type of feature extraction method used in speech processing and analysis [10]. MFCCs are commonly used in speech recognition, speaker identification, and emotion recognition systems. The MFCC technique involves several steps, including pre-emphasis, Framing, Windowing, Fourier transform, Mel-frequency wrapping, and analysis of Cepstral [11]. The feature extraction process using MFCC consists of several stages which are outlined as follows.

### 2.3 Pre-Emphasis

Pre-emphasis is a signal processing technique used to increase the relative strength of higher frequencies in a signal by **Equation (1)**.

$$s_{pre}(n) = s(n) - \alpha\, s(n-1) \quad (1)$$

where $0.9 < \alpha < 1$ is the pre-emphasis factor; $s_{\mathrm{pre}}(n)$ is the sound signal after pre-emphasis on the sound at time $n$, and $s(n)$ the sound signal at time $n$. The reference for selecting the pre-emphasis factor $\alpha$ (typically in the range of 0.9 to 1) is commonly based on empirical studies and standard practices in speech processing **[12]**.

In this article, it is used $\alpha = 0.97$.

### 2.4 Framing and Windowing

After the stage pre-emphasis, the signal is divided into several frames or smaller segments to analyze signals over time. Every frame usually overlaps (overlap frame) to ensure signal continuity. After the signal is divided into several frames, the next process is to use the window. Windowing is produced by multiplying each frame with the windowing, which tapers the edges frame to zero. This is done to reduce spectral leakage that occurs when the signal is analyzed using a transform Fourier**[13]**. Function Window commonly used are Hamming window with **Equation (2)**, but other Windowing functions like Hamming and Blackman can also be used.

$$w[n] = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \quad (2)$$

where, $0 \le n \le N-1$, $N$ is the length of the window; $w[n]$ is the value of the window on the time index $n$. The parameters that appear in $w[n]$ have already appeared in the literature, which gives the modeler the opportunity to be able to create adjusted parameters so that they become novelties in this study. The values of 0.54 and 0.46 were obtained from optimizations performed to achieve certain characteristics of the Hamming window. Essentially, these coefficients are chosen so that the Hamming window has a lower sidelobe compared to a rectangular window but still has an acceptable main lobe width. Broadly speaking, the voices of the three apes undergo framing and windowing so that they can be processed by LSTM. One example of the process of pre-emphasis and windowing is depicted in **Figure 2**.

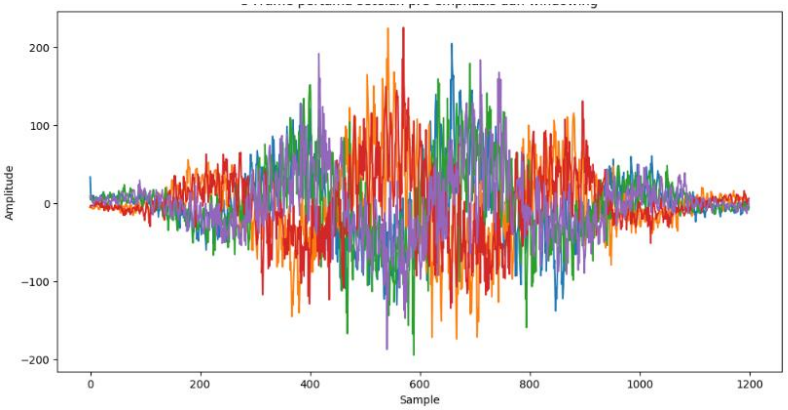



**Figure 2. The first 5 Frames, after Pre-Emphasis and Windowing from One Example of Sound Waves**

### 2.5 Power Spectrum

The next process is the calculation of the frequency spectrum using Fast Fourier Transform (FFT). This process is known as a power spectrum, which aims to obtain information about the distribution of frequency energy in each frame of the sound signal **[14]**.

RFFT (Real-valued Fast Fourier Transform) is a variant of FFT that is optimized specifically for real-number data. To calculate the absolute value of FFT (Fast Fourier Transform) or RFFT, we can use the following formula:

$$\text{Power Spectrum} = \frac{|RFFT(x)|^2}{m} \tag{3}$$

Where:

$|RFFT(x)|$ : the absolute value of the RFFT result of the *x* signal;

$M$ : the total number of samples in the signal.

We first use the real-valued Fast Fourier Transform (RFFT), which is an FFT variant that is optimized for real number data. RFFT efficiently calculates the frequency spectrum of a real number sound signal without having to calculate the negative frequencies that would be symmetrical in the context of the real signal. The general formula for RFFT is not fundamentally different from FFT, but since the input signal is a real number, some optimizations can be applied. The RFFT formula can be expressed as:

$$X_k = \sum_{n=0}^{M-1} x_n e^{-j\frac{2\pi kn}{M}}$$

where $X_k$ is the coefficient at the *k*-th frequency, $x_n$ the input signal sample at *n* time, $M$ is the total number of samples in the signal, and *j* is the imaginary unit. To obtain the amplitude or magnitude of the spectrum (absolute value), the following formula can be used:

$$|X_k| = \sqrt{Re(X_k)^2 + Im(X_k)^2}$$

The result of RFFT is a series of complex coefficients that represent the amplitude and phase of the various frequency components in the input signal. In particular, the RFFT result for a real number input signal will have certain characteristics because the input signal is a real number. The first half of the RFFT results reflects the amplitude of the frequencies present in the signal, and the second half reflects the amplitude of the symmetrical frequencies (negative frequencies), which is generally irrelevant in the context of real-number sound signals.

Examples of visualization of RFFT results on voice signals can provide a better Figure. If we look at the result of the RFFT as a plot of amplitude to frequency, then the positive frequency axis will cover half of the *x*-axis, and the negative frequency axis (which is symmetrical) is negligible. The equation $X_k = RFFT(x)[k]$ shows that $X_k$ is the complex coefficient at the *k*-th frequency, and RFFT(*x*) represents the result of applying the Real-valued Fast Fourier Transform (RFFT) to the signal $x$. The $k$ index ranges from 0 to $M/2$ where $M$ is the total number of samples in the signal.

Suppose a graph with an *x*-axis and a *y*-axis. The *x*-axis covers the range of positive frequencies (0 Hz to half of the signal loading rate), while the y-axis covers the amplitude of each of those frequencies.

If we have a sound signal with a length of $M$ and we apply RFFT, we will get an $M/2 + 1$ positive frequency coefficient. An amplitude graph of frequency will show peaks that reflect the dominant frequency component in the sound signal. The higher the amplitude of a frequency, the more significant the frequency contribution to the signal.

A periogram is a tool for measuring the distribution of frequency energy in a signal at a specific interval. It provides a more detailed figure of the distribution of frequency energy over a given period.

The relationship between Periogram and RFFT is as follows:

RFFT provides an overview of the frequency distribution in a signal, but it may not provide as clear information as a periogram in terms of how frequency energy changes over time. Periograms provide more detailed spectral information and may be used to evaluate how much energy is contained in a signal at a given frequency over a given time interval. The tool used to measure signal strength at various frequencies in the context of spectral analysis is referred to as a spectrum analyzer or spectrum analyzer. This tool is a special electronic device designed to analyze the frequency spectrum of a signal.

### 2.6 Bank Filter

The next step in the extraction of the MFCC feature is to calculate the value of the bank filter formed based on the periodogram. Typically, the commonly used value for the number of bank filters is 40 (nfilt = 40). Before processing the bank filter, the frequency is converted to the mail scale using **Equation (4)**, where $f$ is the frequency of the sound sample divided by two. This conversion is based on the observation that human perception of frequency is logarithmic. This equation describes a logarithmic transformation that converts frequencies into a Mel scale.

The Mel scale has the goal of mimicking the perception of human hearing, which is better at distinguishing sounds at low frequencies than at high frequencies. After converting the frequency to the mel scale, **Equation (5)** is used to convert it back to the frequency scale. In this equation, $f$ is the frequency in the Hertz scale, and $m$ is the frequency in the Mel scale. This conversion is the opposite of **Equation (4)** and allows a return to the original frequency scale.

$$mel_{points} = 2595\log_{10}\left(1 + \frac{f}{700}\right) \tag{4}$$

$$Hz_{points} = 700\left(10^{\frac{m}{2595}} - 1\right) \tag{5}$$

The value of 2595 is a constant derived from the transformation of frequency to the Mel scale. The value of 700 is the frequency limit between pulsive sounds and continuous sounds in human hearing. Using this formula, the lower frequencies have greater differences in the Mel scale compared to the higher frequencies. This reflects the fact that humans are more sensitive to frequency differences in the lower range than in the higher range. This formula is used extensively in sound processing, including in the extraction of features such as Mel-Frequency Cepstral Coefficients (MFCC). The bank filter in MFCC is also known as a triangular filter because its response is 1 at the center frequency and decreases to 0 at the center frequencies of adjacent filters. **Equation (6)** is used to describe the triangular bank filter response.

$$H_m(k) = \begin{cases} 0, & k < f(m-1) \\ \dfrac{k - f(m-1)}{f(m) - f(m-1)}, & f(m-1) \le k \le f(m) \\ \dfrac{f(m+1) - k}{f(m+1) - f(m)}, & f(m) < k \le f(m+1) \\ 0, & k > f(m+1) \end{cases} \tag{6}$$

where:

$H_m(k)$ : Output to $k$ from the filter bank

$f(m-1)$ : Lower limit of the filter bank ($M-1$)

$m$ : The value of the interval from 1 to the number of filters (nfilt+1)

$k$ : Bank filter index.

Finally, after using the output of the filter bank, we can use it to be the input of our LSTM model.

### 2.7 Long Short-Term Memory Revisited (LSTM)

Long Short Term Memory is another form of Recurrent Neural Network (RNN) method in machine learning that can be applied in predicting time series data, such as velocity and direction on sea surface **[15]**, its hybrid with autoregression for cryptocurrency price prediction **[16]**. It is used to deal with the RNN problem of vanishing gradient. For traditional RNNs, as presented by **[17]** and **[18]**, the gradient for the loss function might turn out very small. Therefore, LSTM introduced a forget gate in its architecture to maintain this problem. The following steps describe the general procedures in LSTM for sound classifications.

a. Feature Extraction: Extract features from sound signals through the process of MFCC **[11]**.

b. Data preparation: Prepare data in the form of a sequence of MFCC features that will be used as inputs for the LSTM model.

c. Model Architecture: Design an LSTM model architecture suitable for tasks in sound classification.

Typically, this will contain a few layers of LSTMs finally fed into a dense layer for classification. Finally, the model is trained and evaluated with the standard metrics in evaluation. These are the accuracy, precision, recall, and the F1 score. To overcome the overfitting, some steps are considered here.

There are some strategies that can be pursued to handle overfitting. First, an important one is model simplicity. This can be done by decreasing the number of layers since fewer layers are synonymous with simpler models that are more difficult to overfit to. Moreover, decreasing the number of units in every layer reduces the model's ability to overfit the training data, as there is a reduction in noise. Other methods like regularization also work well, such as dropout and L1 or L2 regularization for example. In brief, the strategy of dropout prevents co-adaptation and enables robust feature learning by randomly turning off a certain percentage of neurons while learning. On the other hand, L1 and L2 regularization improve models by adding a penalty attribute to the loss function to reduce large weights **[19]**. Early stopping is another method, where after training the model, its performance on a validation set is put to the test. If the model proves to be incompetent and the validation loss goes up, then the model is stopped. This further improves the function of the model because it avoids the situation of overfitting the noise in the training set. Another approach is to expand upon the training data with techniques like time shifting or jittering **[20]**, which raises both the size and variety of the data. Additionally, batch normalization, which consists of normalizing activations in each layer during training, can improve generalization while also speeding up training. Hyperparameter tuning stands out since it involves a learning rate, a dropout rate, and other variables. With the various methods above, the LSTM architecture is made in stages in this study in the following flowchart:

(a) The first Model LSTM is illustrated in **Figure 3**.

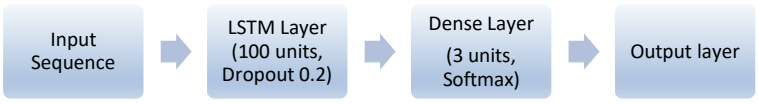



**Figure 3. The Architecture of the First Model LSTM used in this Research**

(b) The second Model of LSTM is illustrated in **Figure 4**.

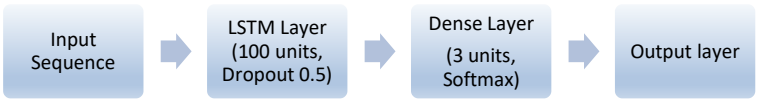



**Figure 4. The Architecture of the Second Model LSTM used in this Research**

(c) The third Model of LSTM is depicted in **Figure 5**.

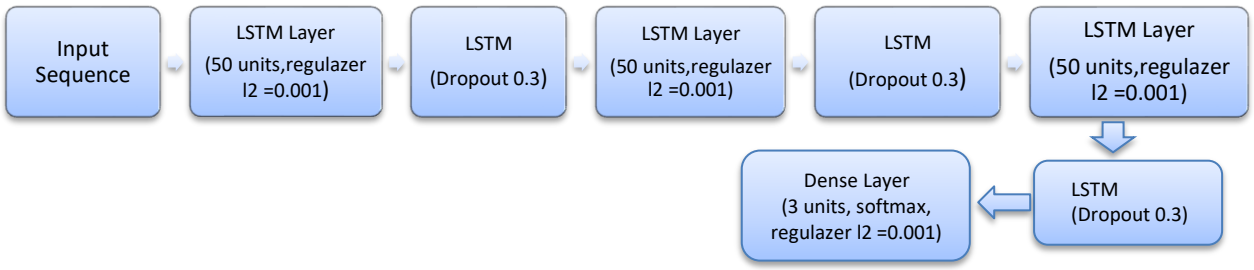



**Figure 5. The Architecture of the Third Model LSTM used in this Research**

(d) The fourth model of LSTM is shown in **Figure 6**.

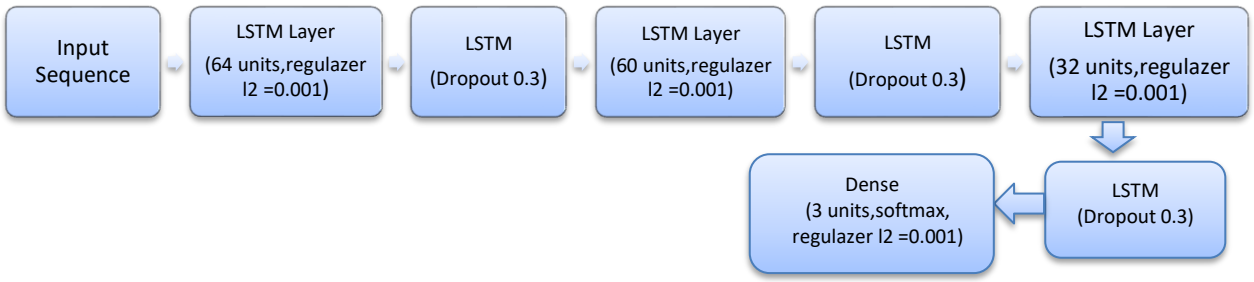



**Figure 6. The Architecture of the Fourth Model LSTM used in this Research**

### 2.8 Modelling and Evaluation

After the extraction of sound spectral features in matrix form, the dataset was then split into training and testing subsets. Thereafter, an instance of the LSTM model was trained on the former. After the end of the training phase, the trained model was deployed on the testing data to test its efficiency in identifying social behaviors through monkey vocalizations. Performance measures such as accuracy, precision, recall, F1-score, and support were computed with a confusion matrix.

### 2.9 Logistic Regression and Misclassification

Logistic Regression is a simple and widely used statistical model for binary classification. It models the probability of a data point belonging to a particular class using a sigmoid function. It is relatively easy to interpret and can be efficient for smaller datasets. Misclassification occurs when a model incorrectly predicts the class of a data point. There are 2 types of misclassifications:

a. False Positives: Predicting a positive class when the actual class is negative.
b. False Negatives: Predicting a negative class when the actual class is positive.

The impact of misclassification depends on the specific application. For example, in medical diagnosis, a false negative can have serious consequences. The choice of techniques to handle overfitting depends on the specific dataset and the complexity of the problem.

It is important to carefully evaluate the mode's performance on a held-out test set to ensure that it generalizes well to unseen data. Understanding the types of misclassifications and their potential impact is crucial for selecting appropriate evaluation metrics and making informed decisions. Understanding the types of misclassifications and their potential impact is crucial for selecting appropriate evaluation metrics and making informed decisions.

## 3. RESULTS AND DISCUSSION

The data collected were subsequently analyzed by the methods outlined in the chapter on methodology. The vocalizations from apes were processed and recognized using Python. If the training loss curve is always going down and the validation loss curve shows a plateau or rising, overfitting has happened. Increasing trends in both curves confirm overfitting. Monitoring the loss curves of a model during training goes a long way in preventing it from merely memorizing training data and, hence, in developing an ability to generalize to unseen data. Strategies to avoid overfitting include dropout to reduce model complexity, acquiring more training data if possible, tuning dropout rate, number of layers, batch size, or other hyperparameters. Model architecture designs and a set of hyperparameters are experimented with to avoid overfitting and improve the model's performance on validation data. Note that the 3 classes are ordered as: Squirrel, Tamarin, and Macaque and are named classes 1, 2, and 3, respectively. The voices in the given address are processed to obtain the MFCC from the sound data of the three apes which is done with the help of a Python program. The examples of visualizations are shown in **Figure 7**.

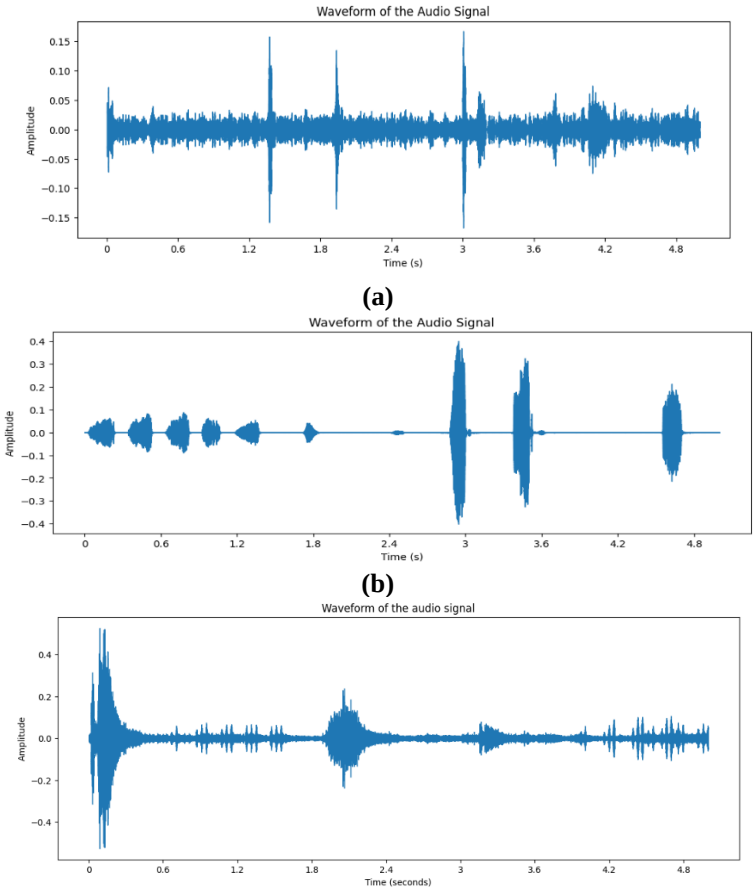



**Figure 7. Examples of Sound Waves from (a) Squirrel Monkey, (b) Golden Lion Tamarin, (c) Long-tailed Macaque**

Throughout this study, the number of MFCC coefficients to be calculated for each audio frame is 10 at the beginning of the study, with a sound size of 54 x 10. In the context of MFCC (Mel-Frequency Cepstral Coefficients), "sound size" could refer to the dimensions of the feature matrix (e.g., 54 frames × 10 coefficients). Note that the test data is 30% of the data set, i.e., 37 is the number of training data, and 17 is the number of data tests. The results for the MFCC method are not included here because the main purpose was to test and compare feature extraction methods specifically designed for swiftlet nest classification. While MFCC features are widely used in audio signal processing, they are less relevant within the context of visual and structural properties of swiftlet nests. This also goes with the objective of the study, which focuses on the methods of feature extraction that best represent the physical characteristics.

Regarding the performance metrics-accuracy, precision, recall, and F-score-of multinomial logistic regression, these have not been presented, as this method was more exploratory in nature in this work. Logistic regression was used as a baseline for classification tasks. This will be rectified in future work by explicitly stating these metrics for a better comparison of its performance relative to other methods. The other major limitation was the absence of these metrics, too, which we would like to include in future research endeavors.

### 3.1 Results of the First Model of LSTM

The LSTM model used is a simple LSTM model with one layer of LSTM and one layer of dense, which is illustrated in **Figure 3.** The sound data is reshaped to meet LSTM inputs that require three dimensions (number of samples, number of features, and time). The model was trained with 50 epochs (rounds through the entire dataset) and a batch size of 32. The model evaluation is carried out using test data, and the results are displayed as accuracy and loss. The result is shown in **Figure 8.**

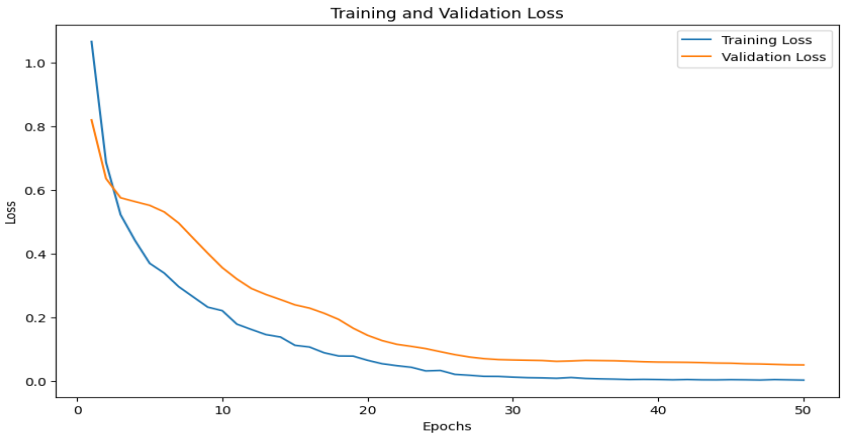



**Figure 8. The Result of the Loss Function Curve of the Approach with the LSTM model: one layer of LSTM with 100 units, Dropout of 0.2 to Prevent Overfitting, Dense Layer with 3 units, and Softmax Activation for Classification.**

The training and validation loss curves of an LSTM model over 50 epochs are depicted in **Figure 8**. Both curves exhibit a downward trend, indicating that the model is learning from the data. However, the validation loss curve begins to diverge from the training loss curve around the 10th epoch, suggesting potential overfitting, where the model becomes too specialized to the training data and performs poorly on new data.

### 3.2 Results of the 2nd LSTM Model

In this subsection, we try to use another model of LSTM. This is shown in **Figure 4.** The training loss curve shows the loss on the training dataset at each epoch, as shown in **Figure 9**, generally decreasing as the model learns from the data. The validation loss curve shows the loss of a separate validation dataset at each epoch, helping to assess how well the model generalizes to unseen data. In the early epochs, both the training and validation loss decreased rapidly, indicating that the model is learning effectively. However, in the middle epochs, the training loss continues to decrease while the validation loss plateaus or starts to increase slightly, suggesting that the model might be starting to overfit the training data. In the later epochs, the training loss continues to decrease while the validation loss increases more significantly, indicating that the model is memorizing the training data instead of learning generalizable patterns. Overall, the model shows good performance in the early stages of training, but it eventually starts to overfit the training data.

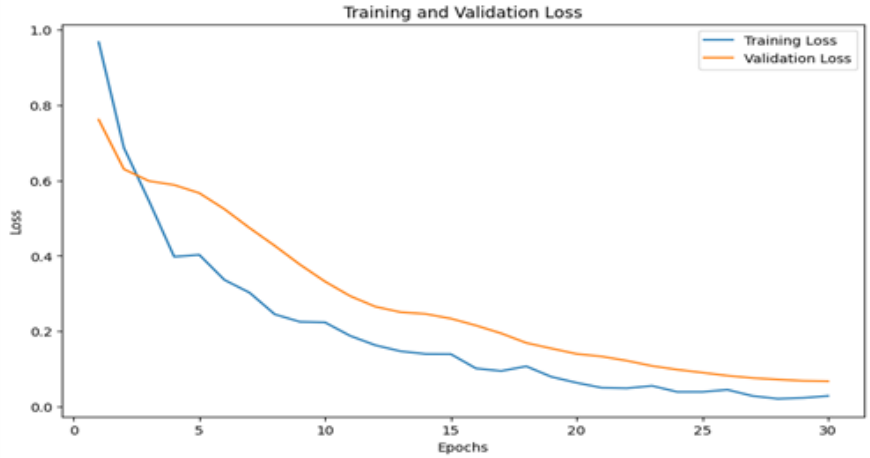



**Figure 9. The Result of the Loss Function Curve of the Approach with the 2nd LSTM Model: LSTM Model With 100 Units (Neurons); Added A Dropout with 50% off Each Iteration and Added a Dense Layer With 3 Neuron Outputs. The Activation Function is Softmax.**

### 3.3 Results of the 3rd LSTM Model

An LSTM model with L2 regularization and dropout was constructed, incorporating three LSTM layers and a final dense layer for classification, which is illustrated in **Figure 5**. The model was trained on the provided data for 20 epochs, and its performance was evaluated using training and validation loss curves, as depicted in **Figure 10**. Initially, both training and validation losses decreased, indicating effective learning. However, a subsequent divergence emerged, with training loss continuing to decline while validation loss plateaued and eventually increased, signaling overfitting. The model's tendency to memorize training data rather than generalize to new data was evident. Thus, the result is generally bad. While the model initially shows promise by learning effectively, the subsequent overfitting is a significant issue. Overfitting means the model has learned the training data too well, but it struggles to generalize to new, unseen data. This is evident in the increasing validation loss while the training loss continues to decrease.

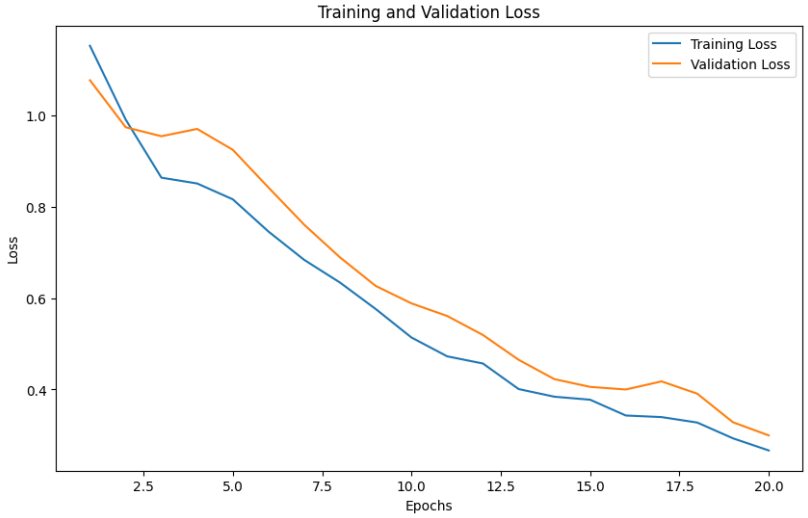



**Figure 10. The Result of the 3rd LSTM Model**

**Figure 10** is obtained by adding L2 regularization to the LSTM and dense layers to help reduce overfitting. In addition, the dropout rate is also updated. The model demonstrates effective learning in the early stages of training, as evidenced by the decreasing validation loss. However, the subsequent increase in validation loss while the training loss continues to drop is a strong indicator of overfitting. This means the model is becoming overly specialized to the training data and may struggle to generalize to unseen data.

### 3.4 Results of the 4th LSTM Model

The 4$^{th}$ Model is introduced to study for handling overfitting that may occur. **Figure 6** depicts the illustration of the 4$^{th}$ LSTM Model, and the result is shown in **Figure 11**.

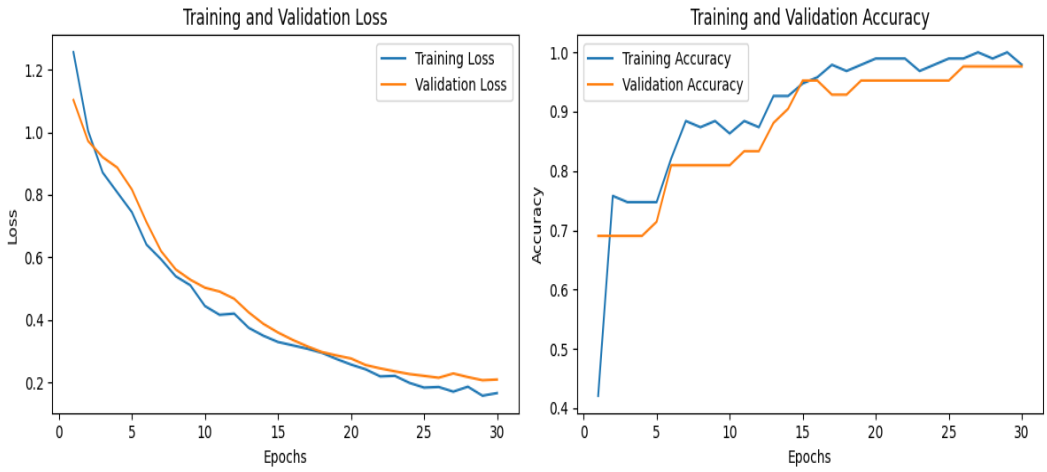



**Figure 11. Results of the 4th LSTM**

**Figure 11** displays training and validation loss and accuracy curves over 30 epochs. The training loss steadily decreases, indicating effective learning from the training data (37 of the dataset), while the validation loss initially decreases but subsequently plateaus and increases, suggesting potential overfitting. A similar trend is observed in the accuracy curves, with training accuracy improving consistently and validation accuracy leveling off at a lower value. These patterns collectively indicate that the model is overfitting the training data. The classification parameters are listed in **Table 1** to give more numerical evidence of the results.

**Table 1. Classification Report of the 4th LSTM Model**

| Class | Precision | Recall | F1-Score | Support | Confusion Matrix |
|---|---|---|---|---|---|
| 1 | 1.00 | 0.62 | 0.77 | 8 | |
| 2 | 0.62 | 1.00 | 0.77 | 5 | [[ 5 3 0] |
| 3 | 1.00 | 1.00 | 1.00 | 29 | [ 0 5 0] |
| Accuracy | 0.93 | | | 42 | [ 0 0 29]] |
| AVG Macro | 0.88 | 0.88 | 0.85 | 42 | |
| Weighted avg | 0.96 | 0.93 | 0.93 | 42 | |

Overall, the 4th LSTM model demonstrates promising performance with a high overall accuracy of 0.93, indicating its ability to correctly classify most instances. The model excels in identifying class 3, achieving perfect precision and recall, suggesting accurate classification without errors. Additionally, the F1-score of 0.85 reflects a reasonable balance between precision and recall across all classes. However, the model's performance is imbalanced, with lower precision and recall for classes 1 and 2 compared to class 3, potentially due to class distribution disparities. Furthermore, the confusion matrix reveals some misclassifications between classes 1 and 2, raising concerns about potential overfitting, especially if the model exhibits significantly better performance on training data compared to unseen data.

### 3.5 Result of Logistic Regression

Since misclassifications persisted in the LSTM model, logistic regression was employed. The confusion matrix in **Figure 12** demonstrates that logistic regression achieved zero misclassifications.

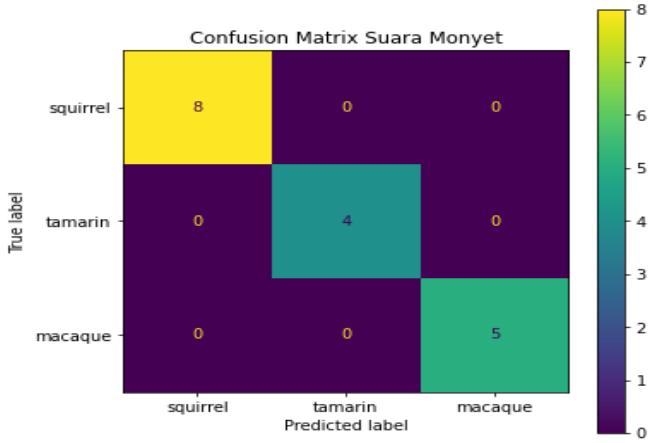



**Figure 12. The Matrix Shows Accuracy Based on the Confusion Matrix for Data Test Where No Sounds Are Misclassified Using Logistic Regression.**

**Figure 12** indicates that the model did a good job in three classes: squirrel, tamarin, and macaque since there are no nonzero elements in the off-diagonal. Each cell of the matrix includes the number of instances classified in a particular class. The actual label line gives the actual categories, namely squirrel, tamarin, and macaque, from the test data, while the predicted label column gives the categories that the model has given. The main diagonal of the matrix represents the number of correctly classified instances, or true positives (TP), with 8 instances correctly identified as squirrels, 4 as tamarins, and 5 as macaques. Since all values outside

the main diagonal are zero, there are no misclassifications, giving a model accuracy of 100%. This proves the capability of logistic regression in providing an optimal classification model for the dataset. One study on multispecies distribution models investigated accounting for heterogeneity across various classification processes, considering different classification probabilities; this can increase precision demonstrably. This improves the estimation of model parameters and enhances predictive performance, especially where the number of misclassified samples is considerably high **[21]**.

## 4. CONCLUSION

This study aimed to classify monkey vocalizations into three categories: squirrel, tamarin, and macaque, utilizing Mel-Frequency Cepstral Coefficients (MFCCs) as feature representations. MFCCs, a widely employed technique in audio processing, were chosen for their ability to capture perceptually relevant information about the spectral content of sounds. The extracted MFCC features were subsequently fed into both LSTM and logistic regression models. While the LSTM model suffered from overfitting despite hyperparameter tuning, logistic regression demonstrated superior performance, achieving perfect classification accuracy. These findings suggest that for this specific dataset and classification task, the simplicity of logistic regression outperformed the complexity of the LSTM model. However, it's essential to consider that the effectiveness of different models can vary across datasets and problem domains. Future research could explore ensemble methods combining MFCCs with other feature representations or more sophisticated deep learning architectures to potentially enhance classification performance.

## ACKNOWLEDGMENT

This article is the result of an internal research project conducted at UKSW in 2023. The research entitled "Mel-Frequency Cepstral Coefficients (MFCC) and Long Short-Term Memory (LSTM) with Internet of Things (IoT) for Voice Classification" was funded under contract number 123/SPK-PT/RIK/9/2023.

## REFERENCES

[1] A. Berdasco, G. López, I. Diaz, L. Quesada, and L. A. Guerrero, "USER EXPERIENCE COMPARISON OF INTELLIGENT PERSONAL ASSISTANTS: ALEXA, GOOGLE ASSISTANT, SIRI AND CORTANA," 2019, p. 51. doi: 10.3390/proceedings2019031051.

[2] P. K. Murali, M. Kaboli, and R. Dahiya, "INTELLIGENT IN-VEHICLE INTERACTION TECHNOLOGIES," *Adv. Intell. Syst.*, vol. 4, no. 2, p. 2100122, 2022, doi: 10.1002/aisy.202100122.

[3] Y. Iliev and G. Ilieva, "A FRAMEWORK FOR SMART HOME SYSTEM WITH VOICE CONTROL USING NLP METHODS," *Electron.*, vol. 12, no. 1, pp. 1–13, 2023, doi: 10.3390/electronics12010116.

[4] N. K. Manaswi, DEEP LEARNING WITH APPLICATIONS USING PYTHON: CHATBOTS AND FACE, OBJECT, AND SPEECH RECOGNITION WITH TENSORFLOW AND KERAS. Bangalore, Karnataka, India, 2018. [Online]. Available: https://www.hlevkin.com/hlevkin/45MachineDeepLearning/DL/Deep Learning with Applications Using Python.pdf

[5] B. Fernandes and K. Mannepalli, "SPEECH EMOTION RECOGNITION USING DEEP LEARNING LSTM FOR TAMIL LANGUAGE," *Pertanika J. Sci. Technol.*, vol. 29, no. 3, pp. 1915–1936, 2021, doi: 10.47836/pjst.29.3.33.

[6] A. Mahmood and U. Kose, "SPEECH RECOGNITION BASED ON CONVOLUTIONAL NEURAL NETWORKS AND MFCC ALGORITHM," *Adv. Artif. Intell. Res.*, vol. 1, no. 1, pp. 6–12, 2021, [Online]. Available: https://dergipark.org.tr/en/pub/aair/issue/59650/768432

[7] dan T. D. D. S. U. Bhandari, H. S. Kumbhar, V. K. Harpale, "ON THE EVALUATION AND IMPLEMENTATION OF LSTM MODEL FOR SPEECH EMOTION RECOGNITION USING MFCC," in *Proceedings of International Conference on Computational Intelligence and Data Engineering*, 2022, pp. 421–434.

[8] M. S. Beauchamp, "FACE AND VOICE PERCEPTION : MONKEY SEE , MONKEY HEAR," vol. 31, no. 9, pp. 1–7, 2021, [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960982221003043

[9] K. J. Devi, A. A. Devi, and K. Thongam, "AUTOMATIC SPEAKER RECOGNITION USING MFCC AND ARTIFICIAL NEURAL NETWORK," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 1, pp. 39–42, 2019, doi: 10.35940/ijitee.A1010.1191S19.

[10] Z. K. Abdul and A. K. Al-Talabani, "MEL FREQUENCY CEPSTRAL COEFFICIENT AND ITS APPLICATIONS: A REVIEW," *IEEE Access*, vol. 10, pp. 122136–122158, 2022, doi: 10.1109/ACCESS.2022.3223444.

[11] S. M. Widodo, E. Siswanto, and O. Sudjana, "PENERAPAN METODE MEL FREQUENCY CEPTRAL COEFFICIENT DAN LEARNING VECTOR QUANTIZATION UNTUK TEXT-DEPENDENT SPEAKER IDENTIFICATION," *J. Telemat.*, vol. 11, no. 1, pp. 15–20, 2016, [Online]. Available: https://journal.ithb.ac.id/telematika/article/view/147/pdf

[12] A. Abdo *et al.*, "PARTIAL PRE-EMPHASIS FOR PLUGGABLE 400 G SHORT-REACH COHERENT SYSTEMS," *Futur. Internet*, vol. 11, no. 12, pp. 1–10, 2019, doi: 10.3390/FI11120256.

[13] M. Labied, A. Belangour, M. Banane, and A. Erraissi, "AN OVERVIEW OF AUTOMATIC SPEECH RECOGNITION PREPROCESSING TECHNIQUES," *2022 Int. Conf. Decis. Aid Sci. Appl. DASA 2022*, pp. 804–809, 2022, doi: 10.1109/DASA54658.2022.9765043.

[14] H. Manus, "AN ULTRA-PRECISE FAST FOURIER TRANSFORM," *Sci. Talks*, vol. 4, no. December, pp. 1–26, 2022, [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772569322000974

[15] D. D. Pramesti, D. C. R. Novitasari, F. Setiawan, and H. Khaulasari, "LONG-SHORT TERM MEMORY (LSTM) FOR PREDICTING VELOCITY AND DIRECTION SEA SURFACE CURRENT ON BALI STRAIT," *BAREKENG J. Ilmu Mat. dan Terap.*, vol. 16, no. 2, pp. 451–462, 2022, doi: 10.30598/barekengvol16iss2pp451-462.

[16] I. M. Nur, R. Nugrahanto, and F. Fauzi, "CRYPTOCURRENCY PRICE PREDICTION: A HYBRID LONG SHORT-TERM MEMORY MODEL WITH GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY," *BAREKENG J. Ilmu Mat. dan Terap.*, vol. 17, no. 3, pp. 1575–1584, 2023, doi: 10.30598/barekengvol17iss3pp1575-1584.

[17] T. Xayasouk, H. M. Lee, and G. Lee, "AIR POLLUTION PREDICTION USING LONG SHORT-TERM MEMORY (LSTM) AND DEEP AUTOENCODER (DAE) MODELS," *Sustain.*, vol. 12, no. 6, 2020, doi: 10.3390/su12062570.

[18] M. Kowsher *et al.*, "LSTM-ANN & BiLSTM-ANN: HYBRID DEEP LEARNING MODELS FOR ENHANCED CLASSIFICATION ACCURACY," *Procedia Comput. Sci.*, vol. 193, pp. 131–140, 2021, doi: 10.1016/j.procs.2021.10.013.

[19] K. S. Mohamed, "BATCH GRADIENT LEARNING ALGORITHM WITH SMOOTHING L1 REGULARIZATION FOR FEEDFORWARD NEURAL NETWORKS," *Computers*, vol. 12, no. 1, pp. 1–15, 2023, doi: 10.3390/computers12010004.

[20] C. Tallec and Y. Ollivier, "CAN RECURRENT NEURAL NETWORKS WARP TIME?," in *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 2018, pp. 1–13. [Online]. Available: https://typeset.io/pdf/can-recurrent-neural-networks-warp-time-s4vftv8ksl.pdf

[21] K. P. Adjei, A. G. Finstad, W. Koch, and R. B. O'Hara, "MODELLING HETEROGENEITY IN THE CLASSIFICATION PROCESS IN MULTI-SPECIES DISTRIBUTION MODELS CAN IMPROVE PREDICTIVE PERFORMANCE," *Ecol. Evol.*, vol. 14, no. 3, 2024, doi: https://doi.org/10.1002/ece3.11092.