JOIV : Int. J. Inform. Visualization, 8(2) - May 2024 735-741

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

CNN-LSTM for Heartbeat Sound Classification

Nurseno Bayu Aji [a,*], Kurnianingsih [a], Naoki Masuyama [b], Yusuke Nojima [b]

[a] *Department Electronic Engineering, Politeknik Negeri Semarang Tembalang, Semarang, 50275, Indonesia*
[b] *Osaka Metropolitan University, Osaka, Japan*

Corresponding author: [*]bayu.nurseno@polines.ac.id

***Abstract*—Cardiovascular disorders are among the primary causes of death. Regularly monitoring the heart is of paramount importance in preventing fatalities arising from heart diseases. Heart disease monitoring encompasses various approaches, including the analysis of heartbeat sounds. The auditory patterns of a heartbeat can serve as indicators of heart health. This study aims to build a new model for categorizing heartbeat sounds based on associated ailments. The Phonocardiogram (PCG) method digitizes and records heartbeat sounds. By converting heartbeat sounds into digital data, researchers are empowered to develop a deep learning model capable of discerning heart defects based on distinct cardiac rhythms. This study proposes the utilization of Mel-frequency cepstral coefficients for feature extraction, leveraging their application in voice data analysis. These extracted features are subsequently employed in a multi-step classification process. The classification process merges a convolutional neural network (CNN) with a long short-term memory network (LSTM), forming a comprehensive deep learning architecture. This architecture is further enhanced through optimization utilizing the Adagrad optimizer. To examine the effectiveness of the proposed method, its classification performance is evaluated using the "Heartbeat Sounds" dataset sourced from Kaggle. Experimental results underscore the effectiveness of the proposed method by comparing it with simple CNN, CNN with vanilla LSTM, and traditional machine learning methods (MLP, SVM, Random Forest, and *k*-NN).**

***Keywords*— Cardiovascular; phonocardiogram; CNN-LSTM; heartbeat.**

Manuscript received 9 Sep. 2023; revised 15 Oct. 2023; accepted 11 Nov. 2023. Date of publication 31 May 2024.

I. INTRODUCTION

According to the World Health Organization (WHO), cardiovascular disease is the most significant cause of mortality worldwide, accounting for around 17.9 million deaths every year [1], [2]. Additionally, acute myocardial infarction is responsible for 85% of cardiovascular events. Cardiovascular conditions encompass various ailments involving blood vessels and the heart, such as rheumatic heart disease, cerebrovascular disorders, and coronary heart disease, among others. Individuals afflicted with hyperlipidemia, diabetes, hypertension, etc., face elevated risks of cardiovascular issues [3], [4]. This disorder gradually becomes a more integral aspect of our lives as we age. In the elderly, cardiovascular disease can have more severe repercussions compared to younger individuals due to diminished recovery rates [5]. Thus, the early identification of cardiac abnormalities [6] plays a crucial role in patient care.

Clinical practice employs numerous tools for diagnosing cardiovascular disease. Auscultation, a fundamental diagnostic technique, involves listening to heart sounds through a stethoscope placed on the patient's chest to facilitate diagnosis [7], [8]. While auscultation remains precise, diagnosing cardiovascular and heart-related disorders, especially for non-clinical and inexperienced individuals, proves challenging. Despite its accuracy, auscultation demands extensive experience and prolonged training to diagnose cardiovascular disease effectively [9].

Phonocardiogram (PCG) is a particularly suitable general heart disease screening method. PCG involves digitally recording and storing heart sounds [10]. This can be achieved through microphones connected to the patient's chest or digital stethoscopes, enabling signal analysis and processing via computer-based methods. The aim is to aid doctors in diagnosing heart disease through computer-assisted heart sound analysis.

There is a wealth of medical data and advanced artificial intelligence technology, with an increasing emphasis on developing deep learning approaches for heart sound classification. Most research has predominantly focused on distinguishing between regular and irregular heartbeat sounds. Notably, a study utilizing an artificial neural network (ANN)

model has attained a precision rate of 90% [11]. Furthermore, the fusion of Variational Mode Decomposition (VMD) with CNN-LSTM has yielded an impressive accuracy of 98.65% [12]. In addition, a one-dimensional convolutional neural network has achieved an accuracy of 93% [9], and MFCC-CNN-RNN has showcased a remarkable precision of 98.63% [13]. While the accuracy is commendable, prior studies have limited themselves to the classification of only two classes. However, given the requirement for more specific information, a model capable of classifying data based on its inherent characteristics or original labels is necessary.

The traditional method for computer-based heart sound analysis consists of three stages: (1) pre-processing (filtering and segmentation), (2) feature extraction, and (3) classifier design [13]. In stages 1 and 2, Deng et al. [13] employ a Butterworth bandpass filter of fifth order (25-400 Hz) for pre-processing and Mel-frequency cepstral coefficients (MFCC) for feature extraction in detecting heart sound abnormalities. In stage 3, classification primarily relies on deep learning, as demonstrated by the work of Yazan Al Issa et al. [14], who employ a CNN-LSTM classifier for multiclass cardiovascular detection.

This study pioneers the integration of two state-of-the-art technologies in the field of cardiovascular anomaly detection: a hybrid CNN-LSTM framework for classification tasks, and MFCC for precise feature extraction. MFCC helps the network focus on essential features while reducing noise and irrelevant information in the raw audio, CNN can significantly reduce the dimensionality of the input data while preserving important information, and LSTM is good for time series data. These combined innovations substantially elevate the model's accuracy and effectiveness.

The subsequent sections of the paper are organized as follows: Section II describes the approach, which includes data pretreatment, feature extraction, and the classification algorithm. Section III presents the experimental setup, datasets, evaluation metrics, results, and discussions. Section IV concludes the work by suggesting future research alternatives.

II. Material and Method

This study develops a method to classify heart disease based on heartbeat sounds. The proposed method utilizes MFCC for feature extraction and combines CNN and LSTM for the classifier. The overall method is illustrated in Fig. 1.

A. Dataset

The dataset was initially generated for a machine learning competition focused on categorizing heartbeats and was obtained from Kaggle (https://www.kaggle.com/datasets/kinguistics/heartbeat-sounds) [15]. The information originated from two distinct origins: (A) the broader public through the iStethoscope app and (B) a medical study conducted in hospitals using the digital stethoscope DigiScope.

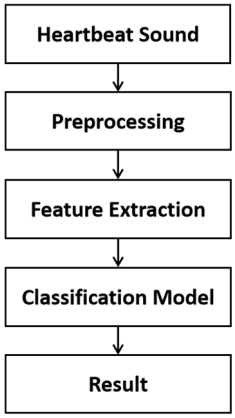

Fig. 1 Architecture of the proposed methods.

The dataset comprises sound files in *.wav format and label data in *.csv format. The data is categorized into five classes, totaling 767 pieces, with the percentages shown in Fig. 2. These classes encompass normal, murmur, artifact, extrasystole, and extrahls, as illustrated in Fig. 2.

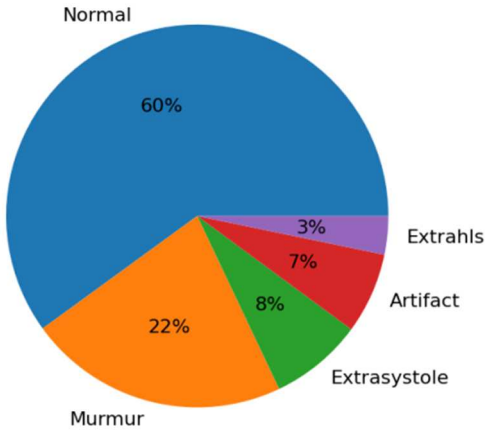

Fig. 2 Class distribution of the Heartbeat sound dataset.

This dataset has been employed in numerous studies. Some studies utilize two classes, categorizing data as normal and abnormal [9], [12], [13]. In this particular study, we focus on three labels/classes. The labels or classes utilized in this study include typical, murmur, and artifact and remove extrasystole and extrahls.

B. Preprocessing

First, the heartbeat dataset in *.wav format undergoes processing to convert it into signal data. Sound transmits air pressure waves to our ears. A digital audio file is generated by detecting these sound waves and transforming them into electrical signals through a sound sensor. This process provides insights into wave displacement and its temporal evolution.

Fig. 3 illustrates a heartbeat sound signal. The x-axis represents the time duration of the heartbeat, and the y-axis represents the displacement of air molecules. Amplitude quantifies the extent of this displacement from the molecule's equilibrium position.

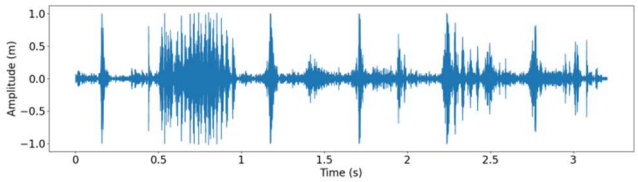

Fig. 3 Heartbeat sound signal.

Secondly, the heartbeat signal is transformed into a spectrum. The spectrum concisely depicts a sound by showcasing vibration intensity at various discrete frequencies. Typically presented as a graph, it illustrates power or pressure against frequency. Power or pressure is often quantified in decibels, while frequency is measured in hertz (Hz) or kilohertz (kHz).

The spectrum, obtained through sound analysis, represents the frequency composition of the sound. A sound spectrum is commonly visualized on a coordinate plane, where the frequency (f) is shown horizontally (abscissa), and the amplitude (A), or intensity, of a specific frequency's harmonic component is plotted along the vertical axis (ordinates). The spectrum is illustrated in Fig. 4.

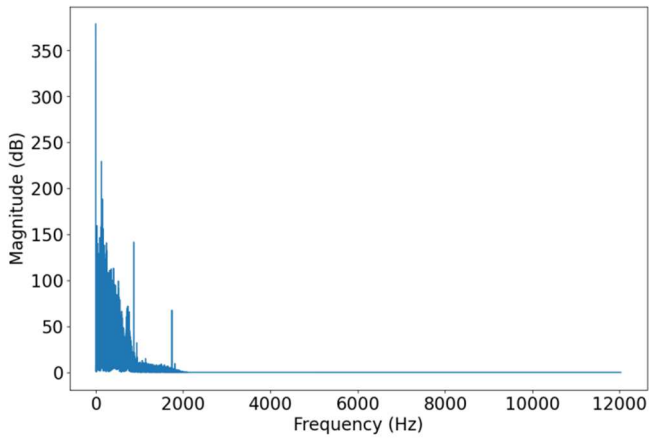

Fig. 4 Spectrum from the heartbeat.

Human perception of sound is influenced not only by its instantaneous intensity but also by its pitch, which correlates with frequency. A higher pitch corresponds to a higher frequency, and vice versa. To create a representation more in line with human cognition, we introduce a spectrogram, as depicted in Fig. 5. A spectrogram serves as a visual representation of how a signal's frequency spectrum changes over time [16]. In the context of audio signals, spectrograms are also commonly known as sonographs, voiceprints, or voicegrams.

C. Feature Extraction

Machine learning relies heavily on feature extraction and selection. Audio signal axes include frequency, amplitude, and time. An audio signal's spatial and temporal characteristics give clear information [17], [18]. The audio signal, a time-varying entity, can be digitally quantified. It comprises numerous frequency sound waves with varying amplitudes.

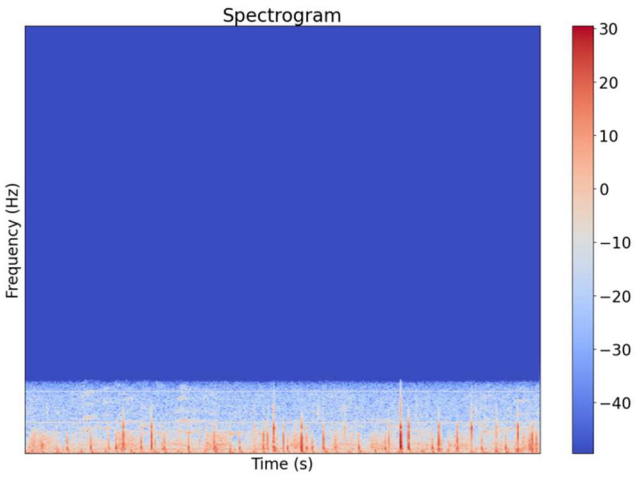

Fig. 5 Spectrogram from the heartbeat.

Each sample may be broken into sine and cosine signals utilizing the Fourier Transform, resulting in a spectrum. A frequency content spectrogram is a time-based representation of frequency content. Over time, sound content reveals abnormalities and variations. To address this, multiple Fast Fourier Transform (FFT) operations are performed on distinct windows of the sound source. For spectrum analysis, these windowed signals are subjected to the FFT technique. Mel-filtering is a human perception-inspired process that merges frequency components from Mel-filter bands into a single energy intensity. The logarithm of all Mel-filter band intensities is used in the non-linear transformation. The Modified Discrete Cosine Transform (DCT) is then applied to convert the altered intensities into MFCC [19]. The overall process of MFCC is illustrated in Fig. 6.

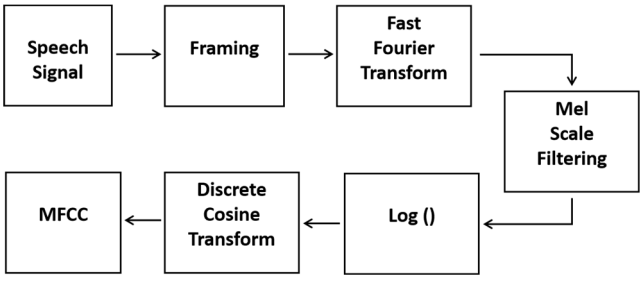

Fig. 6 MFCC Architecture.

The computation of MFCC involves constructing Mel-Scale filter banks according to the formula [1]:

$$m = 1127 \log_e\left(\frac{f}{100} + 1\right) \quad (1)$$

where, f represents the frequency in the linear scale, while m signifies the frequency in the Mel-Scale. The Spectrum from the signal is transformed using power spectral density (PSD) onto the Mel-Scale through multiplication with the generated filter banks, and the logarithm of the energy output from each filter is calculated as outlined below:

$$s(m) = \log_e\left(\sum_{k=10}^{N-1} |X(k)|^2 H_m(k)\right) \quad (2)$$

where m the number of filter banks and $H_m(k)$ is the filter banks. The MFCC is estimated utilizing the spectrum's discrete cosine transform (DCT):

$$c(n) = \sum_{m=0}^{N-1} S(m)cos(\frac{\pi n}{M}(m - \frac{1}{2})), n = 0,1,2,\ldots,M \quad (3)$$

where M is the total number of filter banks.

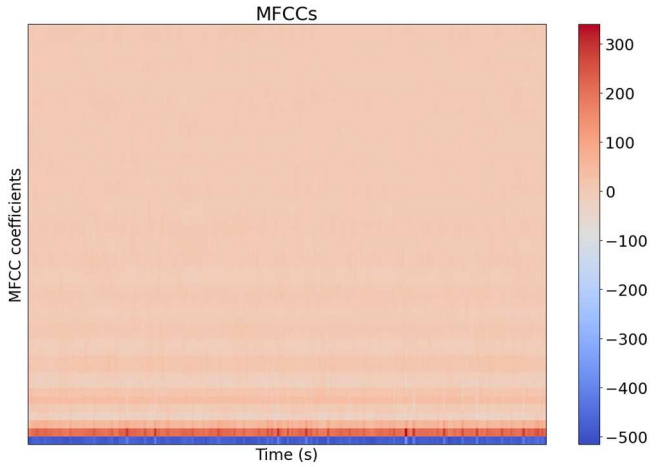

Fig. 7 MFCC Feature.

There are several ways to extract characteristics from audio. We focus on the main qualities that accurately describe an audio clip in our study. Turning an audio stream into a spectrogram depicted on the Mel scale filtering is known as spectrogram imaging. These photos are saved and sent into the algorithm. On the Mel scale filtering, MFCC gives a concise depiction of the spectral envelope [20]. Higher-order coefficients gather pitch and tone, whereas the first 13 coefficients define spectral structure. We used 52 MFCC features in this investigation. Fig. 7 depicts the MFCC Feature.

D. CNN

CNN is a deep learning network capable of identifying and classifying image attributes within computer vision. The configuration and functioning of the CNN are influenced by the organization of the brain's visual cortex, which aims to mimic the neural connections found in the human brain [21]. Each neuron within CNN assesses information in its respective receptive zone. The sequential CNN layers are designed to detect elementary features like lines and curves before progressively moving on to more complex patterns such as faces and objects. This implies that integrating a CNN could potentially enhance computational capabilities [22]. The convolutional layer lies at the heart of the CNN architecture and plays a critical role, as illustrated in Fig. 8.

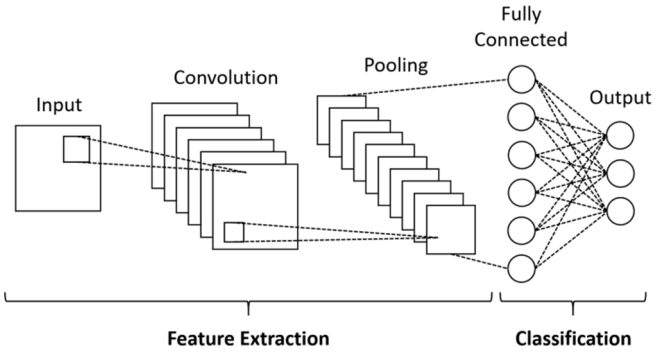

Fig. 8 Convolution Layer [23]

E. LSTM

LSTM emerged as a network model to tackle the persistent issues of gradient expansion and gradient vanishing that plagued RNNs [24]. With its inherent memory and capacity for accurate predictions, it has been widely adopted in applications such as speech recognition, sentiment analysis, and text analysis [25]. Moreover, it has gained recent popularity in the realm of stock market forecasting [26]. In contrast to the traditional single recurrent module structure of an RNN, typically using a tanh layer, LSTM boasts four distinct interacting modules [27]Fig. 9 depicts the LSTM memory cell's three components: the forget gate, the input gate, and the output gate.

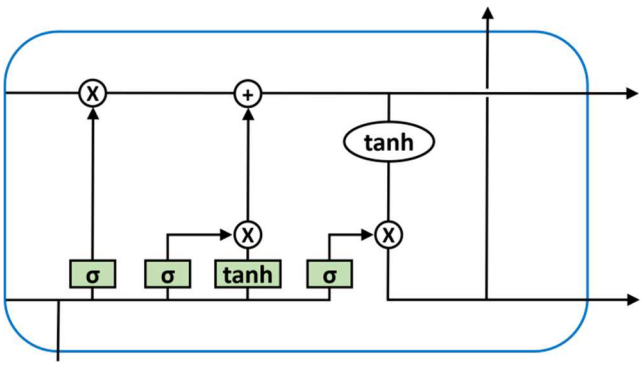

Fig. 9 Architecture of LSTM memory cell [28]

The process of LSTM computation progresses in the following manner:

1) The forget gate utilizes the output value from the previous time step and the input value from the current time step. Subsequently, through calculation, it produces the forget gate output value using this formula [28]:

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad (4)$$

where f_t lies in the range (0, 1), W_f represents the weight of the forget gate, b_f is the forget gate's bias, x_t denotes the input at the current time, and h_{t-1} is the output from the previous moment.

2) The input gate obtains the output from the preceding time step and the input from the present time step. It computes the output value and the candidate cell state for the input gate using these equations:

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i), \quad (5)$$

$$\widetilde{C_t} = tanh(W_c.[h_{t-1}, x_t] + b_c), \quad (6)$$

where, i_t lies in the range (0, 1), W_i represents the weight of the input gate, b_i is the bias of the input gate, W_c stands for the weight of the candidate input gate, and b_c is the bias of the candidate input gate.

3) Apply the subsequent modifications to the current cell state:

$$C_t = f_t * C_{t-1} + i_t * C_t \quad (7)$$

where C_t ranges between (0, 1).

4) At time t, the output h_{t-1} and input x_t are utilized as input values for the output gate, and the output o_t of the output gate is computed as follows:

$$o_t = \sigma(W_0 . [h_{t-1}, x_t] + b_0), \quad (8)$$

where o_t ranges between (0, 1), W_0 represents the weight of the output gate, and b_0 is the bias of the output gate.

5) The LSTM's output value is determined by multiplying the output of the output gate by the cell state, as depicted in the subsequent formula:

$$h_t = o_t * tanh(C_t) \quad (9)$$

F. Proposed CNN-LSTM

Feature extraction is a pivotal stage in numerous machine learning applications, as it generates valuable insights for prediction models, enhancing their accuracy [25][29]. Time-series problems are no exception in this context. They encompass diverse dynamics that necessitate comprehension and adaptation for regression/classification models, often with expert guidance. Moreover, feature extraction not only demands significant time but also exhibits considerable variability across applications [30].

Recently, convolutional techniques have gained prominence as an automated alternative to human-driven feature extraction, mainly through CNN architectures, which have achieved groundbreaking results in addressing computer vision challenges [31]. However, this approach is not limited to image data and can be adapted for time-series data. For this purpose, let *Y* and *X* represent the output and input of the convolution operation, respectively. *W* symbolizes the matrix of learnable parameters, and ⊛ denotes the valid cross-correlation operator. The variables *C* and *L* signify the number of features and the sequence length, respectively. An automated feature extraction framework can be established by undergoing the convolution process and instructing the model on efficient feature extraction by optimizing the weights (*W*).

In the present scenario, the *C* variable is set to 7, representing the number of features, while *L* is designated as 256, consistent with the LSTM architecture. The architecture incorporates three sequential convolutional layers with 2048-512 kernels, accompanied by kernel sizes of 256 and 128, respectively, using a stride of one.

At the culmination of each convolutional layer, both max pooling and layering are implemented. All convolutional operations are executed through a Time Distributed layer to preserve temporal dimensions. After applying the spatial feature extraction capability of the convolutional layers, we introduce the LSTM component to advance the temporal understanding of the data. The predictor was constructed using the same LSTM architecture outlined earlier, encompassing flattened and fully connected layers. This integration effectively merged the spatial feature extraction of CNN with LSTM's aptitude for capturing sequential dependencies. The final cascaded architecture is illustrated in Table 1.

TABLE I
PARAMETER SETTING PROPOSED CNN-LSTM

Layer(type)	Output Shape	Param#
Conv 1D	(None, 52, 1024)	6144
Max Pooling 1D	(None, 26, 1024)	0
Batch Normalization	(None, 26, 1024)	4096
Conv 1D	(None, 26, 512)	2621952
Max Pooling 1D	(None, 13, 512)	0
Batch Normalization	(None, 13, 512)	2048
Conv 1D	(None, 13, 256)	655616
Max Pooling 1D	(None, 7, 256)	0
Batch Normalization	(None, 7, 256)	1024
LSTM	(None, 7, 256)	525312
LSTM	(None, 128)	197120
Dense	(None, 64)	8256
Dropout	(None, 64)	0
Dense	(None, 32)	2080
Dropout	(None, 32)	0
Dense	(None, 3)	99
Total params: 4,023,747		
Trainable params: 4,020,163		
Non-trainable params: 3,584		

G. Adagrad Optimizer

Adagrad is a gradient-based optimization algorithm that employs learning rates to adjust parameters. It implements slight modifications for parameters connected to prevailing characteristics yet executes more substantial adjustments for parameters tied to distinct attributes. This approach demonstrates notable efficiency in managing sparse data, enhancing the performance of stochastic gradient descent (SGD), and finding wide applications in the training of expansive neural networks [32]. The Adagrad optimizer is a gradient-based optimization approach that is well-suited for handling sparse gradients. The learning rate governs the extent to which parameter adjustments are influenced by the inverse direction of a gradient estimate (g). This learning rate is modulated according to the characteristics of the data.

The essential formula for updating parameters is depicted in Equation (10), where θ_t stands for the parameter at a given time t, α, denotes the learning rate, indicates the estimated gradient g_t, and $\odot$ symbolizes element-wise multiplication:

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\varepsilon + \sum g_t^2}} \odot g_t \quad (10)$$

H. Fitness metrics

To assess the effectiveness of the proposed method, this study employs Mean Squared Error (MSE) and F1-score derived from the confusion matrix. MSE represents the average of squared errors, quantifying the disparity between estimated and actual parameter values. MSE indicates the predictive model's accuracy, with superior outcomes tending to be non-negative and closer to zero. Therefore, a lower MSE signifies enhanced performance of the prediction model [33].

Moreover, a reduced MSE signifies a closer alignment of the prediction model with the ideal model. The calculation of MSE is expressed as follows:

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \quad (11)$$

where $\hat{y}_i$ and y_i represent the model predictions and actual outputs, respectively. Here, $\hat{y}_i$ is the mean output, and m signifies the sample count. The data used in this research is

imbalanced, we utilized the F1 score results for model assessment. The confusion matrix is depicted in Fig. 10 for reference.

	Predicted Positive	**Predicted Negative**	**F1 Score:** $\frac{2\ x\ Precision\ x\ Recall}{Precision\ +\ Recall}$
Actual Positive	**TP** True Positive	**FN** False Negative	**Sensitivity (recall):** $\frac{TP}{(TP+FN)}$
Actual Negative	**FP** False Positive	**TN** True Negative	**Specificity:** $\frac{TN}{(TN+FP)}$
	Precision: $\frac{TP}{(TP+FP)}$	**Negative Predicted Value:** $\frac{TN}{(TN+FN)}$	**Accuracy:** $\frac{TP+TN}{(TP+TN+FP+FN)}$

Fig. 10 Confusion matrix

III. Results and Discussion

The proposed method utilizes MFCC for feature extraction and combines CNN-LSTM for classification. Table I presents the parameters of the CNN-LSTM settings for this study.

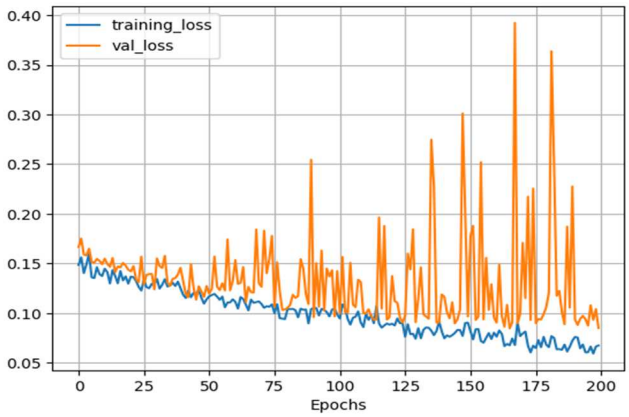

Fig. 11 MSE training error and validation error

In this study, feature extraction using MFCC produces 52 features from the input data of heartbeat sounds. To accommodate the resulting 52 features from MFCC, the first layer of CNN-LSTM is configured as (None, 52, 1). This undergoes several rounds of processing through convolutional layers, followed by the LSTM layer. Finally, a dense layer is employed and optimized for the classification stage with Adagrad. The proposed method is set with 200 epochs during the training process, where MSE serves as the error metric. The training process involves calculating MSE values for each epoch; the best result of MSE at cross-validation is shown in Fig. 11.

Fig. 11 indicates a consistent decrease in training MSE values, while validation MSE appears somewhat inconsistent. The inconsistency in results on the validation data is caused by the characteristics of Adagrad, which is a derivative of stochastic gradient descent with high variance in parameter updates, thereby affecting the results on the validation data [34]. However, despite this disparity, the outcomes do not deviate significantly from the training error. Subsequently, the trained model is evaluated using test data, and the assessment results are presented in the confusion matrix shown in Fig. 12.

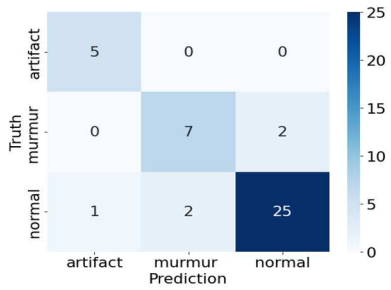

Fig. 12 Confusion matrix

The confusion matrix depicted in Figure 12 reveals that the precision rates for the artifact, murmur, and normal classes are 0.83, 0.78, and 0.93, respectively. The corresponding recall rates are artifact at 1, murmur at 0.78, and normal at 0.89. Overall, the method we propose achieves an accuracy rate of 88%.

TABLE II
PERFORMANCE EVALUATION WITH 10 K-FOLD CROSS-VALIDATION

No	Method	Average F-1 Score	Average Accuracy
1	MLP	0.81	0.66
2	SVM	0.80	0.66
3	Random Forest	0.68	0.65
4	*k*-NN	0.67	0.64
5	CNN	0.54	0.53
6	Simple CNN-LSTM	0.63	0.61
7	Proposed method	0.87	0.81

We evaluated the efficacy of our proposed technique by comparing it with several other classification methods, including multi-layer perceptron (MLP), random forest, support vector machine (SVM), k-nearest Neighbor (k-NN), Convolution Neural Network (CNN), and a simplified CNN-LSTM model that is using vanilla LSTM for the architecture. The input data for all methods except the proposed method are derived from spectrogram feature extraction. We used 10-fold cross-validation to assess the performance of each approach. The comparative results are summarized in Table II. The performance comparison from Table II demonstrates that the suggested approach surpasses traditional classification methods regarding f1 score and accuracy in the realm of heartbeat categorization based on sound. Notably, the CNN-LSTM model combined with MFCC excels in handling imbalanced datasets, as indicated by its top-ranking f1 score in the comparison table.

IV. Conclusion

This study employed a publicly available "heartbeat sound" dataset from Kaggle. We processed it using our proposed method, which combines MFCC for feature extraction and a CNN-LSTM model for classification. This approach yielded a predictive accuracy of 88%. The precision values for the individual classes were 0.93 for normal, 0.78 for murmur, and

0.83 for artifact. Additionally, we achieved an f1-score of 0.87 and an accuracy of 0.81 using 10-fold cross-validation. Future studies might further focus on enhancing preprocessing and feature extraction strategies to fine-tune the distinction between murmurs and normal heart sounds.

REFERENCES

[1] D. M. Nogueira, C. A. Ferreira, E. F. Gomes, and A. M. Jorge, "Classifying Heart Sounds Using Images of Motifs, MFCC and Temporal Features," Journal of Medical Systems, vol. 43, no. 6, May 2019, doi: 10.1007/s10916-019-1286-5.

[2] WHO, "Cardiovascular diseases (CVDs)," 2021. .

[3] F. D. Fuchs and P. K. Whelton, "High Blood Pressure and Cardiovascular Disease," Hypertension, vol. 75, no. 2, pp. 285–292, Feb. 2020, doi: 10.1161/hypertensionaha.119.14240.

[4] H. Yadav et al., "CNN and Bidirectional GRU-Based Heartbeat Sound Classification Architecture for Elderly People," Mathematics, vol. 11, no. 6, p. 1365, Mar. 2023, doi: 10.3390/math11061365.

[5] L. Ciumărnean et al., "Cardiovascular Risk Factors and Physical Activity for the Prevention of Cardiovascular Diseases in the Elderly," International Journal of Environmental Research and Public Health, vol. 19, no. 1, p. 207, Dec. 2021, doi: 10.3390/ijerph19010207.

[6] J. L. Rodgers et al., "Cardiovascular Risks Associated with Gender and Aging," Journal of Cardiovascular Development and Disease, vol. 6, no. 2, p. 19, Apr. 2019, doi: 10.3390/jcdd6020019.

[7] S. Tanwar et al., "Human Arthritis Analysis in Fog Computing Environment Using Bayesian Network Classifier and Thread Protocol," IEEE Consumer Electronics Magazine, vol. 9, no. 1, pp. 88–94, Jan. 2020, doi: 10.1109/mce.2019.2941456.

[8] Y. Kim, Y. Hyon, S. Lee, S.-D. Woo, T. Ha, and C. Chung, "The coming era of a new auscultation system for analyzing respiratory sounds," BMC Pulmonary Medicine, vol. 22, no. 1, Mar. 2022, doi:10.1186/s12890-022-01896-1.

[9] B. Xiao, Y. Xu, X. Bi, J. Zhang, and X. Ma, "Heart sounds classification using a novel 1-D convolutional neural network with extremely low parameter consumption," Neurocomputing, vol. 392, pp. 153–159, Jun. 2020, doi: 10.1016/j.neucom.2018.09.101.

[10] P. T. Krishnan, P. Balasubramanian, and S. Umapathy, "Automated heart sound classification system from unsegmented phonocardiogram (PCG) using deep neural network," Physical and Engineering Sciences in Medicine, vol. 43, no. 2, pp. 505–515, Feb. 2020, doi:10.1007/s13246-020-00851-w.

[11] P. Keikhosrokiani, A. B. Naidu A/P Anathan, S. Iryanti Fadilah, S. Manickam, and Z. Li, "Heartbeat sound classification using a hybrid adaptive neuro-fuzzy inferences system (ANFIS) and artificial bee colony," Digital Health, vol. 9, p. 205520762211507, Jan. 2023, doi:10.1177/20552076221150741.

[12] M. Fakhry and A. Gallardo-Antolín, "Variational Mode Decomposition and a Light CNN-LSTM Model for Classification of Heart Sound Signals," IEEE EUROCON 2023 - 20th International Conference on Smart Technologies, Jul. 2023, doi:10.1109/eurocon56442.2023.10199054.

[13] M. Deng, T. Meng, J. Cao, S. Wang, J. Zhang, and H. Fan, "Heart sound classification based on improved MFCC features and convolutional recurrent neural networks," Neural Networks, vol. 130, pp. 22–32, Oct. 2020, doi: 10.1016/j.neunet.2020.06.015.

[14] Y. Al-Issa and A. M. Alqudah, "A lightweight hybrid deep learning system for cardiac valvular disease classification," Scientific Reports, vol. 12, no. 1, Aug. 2022, doi: 10.1038/s41598-022-18293-7.

[15] A. Raza, A. Mehmood, S. Ullah, M. Ahmad, G. S. Choi, and B.-W. On, "Heartbeat Sound Signal Classification Using Deep Learning," Sensors, vol. 19, no. 21, p. 4819, Nov. 2019, doi: 10.3390/s19214819.

[16] A. Meghanani, A. C. S., and A. G. Ramakrishnan, "An Exploration of Log-Mel Spectrogram and MFCC Features for Alzheimer's Dementia Recognition from Spontaneous Speech," 2021 IEEE Spok. Lang. Technol. Work. SLT 2021 - Proc., pp. 670–677, 2021, doi:10.1109/SLT48900.2021.9383491.

[17] V. Bansal, G. Pahwa, and N. Kannan, "Cough Classification for COVID-19 based on audio mfcc features using Convolutional Neural Networks," 2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON), Oct. 2020, doi:10.1109/gucon48875.2020.9231094.

[18] A. Chowdhury and A. Ross, "Fusing MFCC and LPC Features Using 1D Triplet CNN for Speaker Recognition in Severely Degraded Audio Signals," IEEE Transactions on Information Forensics and Security, vol. 15, pp. 1616–1629, 2020, doi: 10.1109/tifs.2019.2941773.

[19] E. Rejaibi, A. Komaty, F. Meriaudeau, S. Agrebi, and A. Othmani, "MFCC-based Recurrent Neural Network for automatic clinical depression recognition and assessment from speech," Biomedical Signal Processing and Control, vol. 71, p. 103107, Jan. 2022, doi:10.1016/j.bspc.2021.103107.

[20] A. Hamza et al., "Deepfake Audio Detection via MFCC Features Using Machine Learning," IEEE Access, vol. 10, pp. 134018–134028, 2022, doi: 10.1109/access.2022.3231480.

[21] T. Kattenborn, J. Leitloff, F. Schiefer, and S. Hinz, "Review on Convolutional Neural Networks (CNN) in vegetation remote sensing," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 173, pp. 24–49, Mar. 2021, doi: 10.1016/j.isprsjprs.2020.12.010.

[22] R. Yan, J. Liao, J. Yang, W. Sun, M. Nong, and F. Li, "Multi-hour and multi-site air quality index forecasting in Beijing using CNN, LSTM, CNN-LSTM, and spatiotemporal clustering," Expert Systems with Applications, vol. 169, p. 114513, May 2021, doi:10.1016/j.eswa.2020.114513.

[23] D. Bhatt et al., "CNN Variants for Computer Vision: History, Architecture, Application, Challenges and Future Scope," Electronics, vol. 10, no. 20, p. 2470, Oct. 2021, doi: 10.3390/electronics10202470.

[24] M. E. H. Chowdhury, A. Khandakar, K. Alzoubi, and S. Mansoor, "Real-Time Smart-Digital Stethoscope System for Heart Diseases Monitoring," 2019.

[25] N. Gupta and A. S. Jalal, "Integration of textual cues for fine-grained image captioning using deep CNN and LSTM," Neural Computing and Applications, vol. 32, no. 24, pp. 17899–17908, Oct. 2019, doi:10.1007/s00521-019-04515-z.

[26] A. Yadav, C. K. Jha, and A. Sharan, "Optimizing LSTM for time series prediction in Indian stock market," Procedia Computer Science, vol. 167, pp. 2091–2100, 2020, doi: 10.1016/j.procs.2020.03.257.

[27] Z. Jin, Y. Yang, and Y. Liu, "Stock closing price prediction based on sentiment analysis and LSTM," Neural Computing and Applications, vol. 32, no. 13, pp. 9713–9729, Sep. 2019, doi: 10.1007/s00521-019-04504-2.

[28] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-Based Model to Forecast Stock Prices," Complexity, vol. 2020, pp. 1–10, Nov. 2020, doi: 10.1155/2020/6622927.

[29] R. Mutegeki and D. S. Han, "A CNN-LSTM Approach to Human Activity Recognition," 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Feb. 2020, doi: 10.1109/icaiic48513.2020.9065078.

[30] F. Elmaz, R. Eyckerman, W. Casteels, S. Latré, and P. Hellinckx, "CNN-LSTM architecture for predictive indoor temperature modeling," Building and Environment, vol. 206, p. 108327, Dec. 2021, doi:10.1016/j.buildenv.2021.108327.

[31] T.-Y. Kim and S.-B. Cho, "Predicting residential energy consumption using CNN-LSTM neural networks," Energy, vol. 182, pp. 72–81, Sep. 2019, doi: 10.1016/j.energy.2019.05.230.

[32] S. Venkatesh and Dr. M. Jeyakarthic, "Adagrad Optimizer with Elephant Herding Optimization based Hyper Parameter Tuned Bidirectional LSTM for Customer Churn Prediction in IoT Enabled Cloud Environment," Webology, vol. 17, no. 2, pp. 631–651, Dec. 2020, doi: 10.14704/web/v17i2/web17057.

[33] L. Qiao, X. Li, Q. Umer, and P. Guo, "Deep learning based software defect prediction," Neurocomputing, vol. 385, pp. 100–110, Apr. 2020, doi: 10.1016/j.neucom.2019.11.067.

[34] N. Zhang, D. Lei, and J. F. Zhao, "An Improved Adagrad Gradient Descent Optimization Algorithm," 2018 Chinese Automation Congress (CAC), Nov. 2018, doi: 10.1109/cac.2018.8623271.