JOIV : Int. J. Inform. Visualization, 9(2) - March 2025 789-795

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

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

Indonesian Word Sound Recognition Using Convolutional Neural Network Method

Mandahadi Kusuma [a,*], Fayyadh Aunilbarr [a]

[a] *Informatics, Faculty of Sains and Technology, Universitas Islam Negeri Sunan Kalijaga, Yogyakarta, Indonesia*

Corresponding author: [*]mandahadi.kusuma@uin-suka.ac.id

***Abstract*—Access to education, particularly in a university environment, is essential for deaf and hard-of-hearing students as more of them pursue higher education. At UIN Sunan Kalijaga the current challenges are a limited number of sign language interpreters and translating technical terminology in lectures. Many methods are available for speech recognition, but research on how well this method performs in Indonesian has not been published, especially in education-level recognizers. This experimental study aims to investigate if Indonesian words can be recognized through Convolutional Neural Networks (CNN) and to find out the Data Ratio for Training, Validation, and Testing set to get the best performance. The study used a dataset of 4 Indonesian words with the total voice sample, each with 50 voice samples from young adults aged 19-23. Audio data is preprocessed into spectrograms, inputs to the CNN model using TensorFlow. The CNN Model had a 90% accuracy with a 60:20:20 ratio between training, validation, and test data. The other ratios (70:15:15 and 80:10:10) provided accuracy ranges of between 80% to 90%. This study shows that CNNs are the best for Indonesian word recognition and that the data ratio of 60:20:20 is optimal. This result has valuable benefits, such as using voice-to-text over lectures to enhance the ease of learning and education in Indonesia. Further studies should be conducted using different neural network approaches; the denoise approach is also necessary to increase accuracy.**

***Keywords*—CNN; sound; Indonesian; classification; accuracy; waveform; spectrogram; speech; TensorFlow.**

Manuscript received 19 April 2024; revised 26 Oct. 2024; accepted 23 Nov. 2024. Date of publication: 31 Mar. 2025.

I. INTRODUCTION

Deaf or hard-of-hearing students at UIN Sunan Kalijaga face significant challenges during lectures due to the limited availability of translators and difficulties in translating foreign words [1]. To improve accessibility, a voice-to-text feature is proposed, which could assist these students by converting spoken words into text in real-time[2]. This research aims to identify the best model for recognizing Indonesian words, focusing on the application of Convolutional Neural Networks (CNNs) due to their robustness and efficiency in pattern recognition tasks. The study explores the effectiveness of CNNs in accurately identifying specific Indonesian words, thereby enhancing the utility of voice recognition technology in educational settings for the deaf community.

Several voice recognition methods can be used: CNN, MFCC, ANN, and RNN. However, the current study proved that the Madaline model of artificial neural networks is not recommended for voice identification[3]. Meanwhile, the usage of Convolutional and Recurrent Neural Networks could outperform the accuracy if compared to Convolutional Neural Networks alone[4]. Previous research utilizing the MFCC and SVM methods with Indonesian words achieved an F1 score accuracy ranging from 44% to 100% for word classification. [5]. However, the CNN method is widely used in several research, an audio event classification method using convolutional neural networks (CNNs), achieving an accuracy of 81.5% for classifying thirty audio events across multiple datasets[6]. In the current research, the Convolutional Neural Network (CNN) method was chosen because it is considered stronger and faster[7].

A previous study that aims to improve early literacy in Indonesian children using a Convolutional Neural Network (CNN) approach for alphabet speech recognition reached a high accuracy result for this speech recognition system by 84%[8], however for Indonesian considerable vocabulary speech superiority of deep learning technologies, particularly convolutional neural networks (CNN), over traditional hidden Markov models (HMM) for speech recognition. The study proposes a discriminatively trained CNN for Indonesian large vocabulary continuous speech recognition (LVCSR),

achieving significant error reduction rates that show a 7.26% and 9.01% error reduction[9]

Studies on using Convolutional Neural Networks (CNNs) for processing voice data. Liang et al. focus on security by identifying spoofed voices with results of 95% accuracy for detection of fake voices[10], Franti et al. achieve an average accuracy of 71.33%, aiming to improve human-robot interaction by recognizing emotions[11]. Both studies demonstrate the versatility of CNNs in audio analysis and their potential impact on technology that interacts with or mimics human behavior. Another study suggests that the combination of CNN and RNN outperforms CNN alone, achieving 96.66% accuracy for 20 labels[4]. The convergence of this research underscores the broad applicability of CNNs in both safeguarding and humanizing technology.

A study for deaf support applications utilizes deep learning techniques for sound analysis using the Mel-Spectrogram representation of sounds. Real-life sounds using the Korean language are recorded via an app, identified based on learned data and associated with predefined alarms and vibrations. The experiments showed promising results, with an average classification rate of 85% for real-life sounds [12].

Convolutional Neural Networks (CNN) are a type of artificial neural network used in pattern recognition, especially in image processing [13]. CNN has an architecture with several variations, but in general, they consist of convolutional layers and pooling layers, which will be grouped into models. Followed by a fully connected layer, as is standard in JST[14].Convolution layers and pooling layers are internal structures, and fully connected layers handle generating class probabilities[15]. Convolutional layers are part of a CNN that consists of neurons connected to the receptive fields of the previous layer. Filters in convolutional layers are applied explicitly in speech recognition, where sound can be converted into images and then analyzed using CNN[16].

The Dropout and Confusion Matrix techniques are applied so that the effectiveness of CNN in recognizing objects in images becomes the basis for detecting patterns or features in images [17]. Each neuron in a convolutional layer uses the same weight as a particular filter. Meanwhile, the Max pooling layer is the grouping layer most commonly used in CNNs. This layer functions to reduce the dimensions of the spatial representation effectively without adding parameters that can be learned[18]. Dropout is a regularization technique used in artificial neural networks to reduce overfitting by avoiding dependencies between neurons[19]. Meanwhile, the Confusion Matrix is used to evaluate the performance of the classification model by presenting prediction results on test data in matrix form, enabling analysis of the quality of recognition for each class or word[20].

This study aims to evaluate the effectiveness of a Convolutional Neural Network (CNN) for Indonesian word recognition and to determine the accuracy using a dataset of four specific words. Research limitations include using Indonesian and a specific 4-word dataset; the words used are Inklusif, pecah, coba, and miring. The development using the Python programming language and Google Collab [21]. The benefits of this research include demonstrating the ability of CNN to use Indonesian voice recognition and evaluating the accuracy of the dataset used, with novelty in the context of Indonesian language use and different word class voice datasets. Because a person's voice differs in tone, pitch, and volume, it is adequate to make it uniquely distinguishable. By using the CNN process of identifying and classifying a person based on their voice, a high level of accuracy[22]. The study provides information on how CNNs can be used to recognize voices in the language, evaluates the effectiveness of this approach, and supports the advancement of tools to aid the deaf community. Additionally, it emphasizes the benefits of CNNs in enhancing and making technology more user-friendly.

II. Materials and Method

The details of each research stage consist of data collection, data preprocessing, CNN implementation, and analysis and evaluation. Each step is explained as follows:

A. Data Collection

Voice data was collected from volunteers at the Disabled Services Center (PLD) UIN Sunan Kalijaga for four words with 50 data per word, taken between February and March 2021, with permission from PLD. The voice dataset was collected from PLD volunteers with an age range of 19-23 years, consisting of the words "inklusif"*, pecah", coba", miring"*, each with fifty data. The average duration of a sound recording is 1-2 seconds in a controlled environment.

This research uses voice data obtained from male and female PLD volunteers at UIN Sunan Kalijaga, aged 19-23 years. Each word has 50 data from 10 different volunteers, five men, and five women, with four groups of words representing vowels and consonants. Volunteers were asked to say the words in a normal emotional atmosphere and in a calm environment, without distractions from sounds or other activities. Sound collection is done through recording using the application built into the volunteer's mobile device in WAV format. Sound collection criteria are set following Tabel I.

TABLE I
SOUND COLLECTION CRITERIA

Categories	Criteria
Volunteers	Volunteers from PLD UIN Suka who are not disabled students
Age	19-23 years
Words	'Inklusif' (5 times), 'pecah'(5 times), 'coba'(5 times), 'miring'(5 times)
Duration	1-2 seconds
Application	Own recording app and device
Environment	Quiet, not whispering, without other distractions, uttering with ordinary emotion
Data Format	Waveform Audio File Format (WAV)

Each volunteer recorded voices 5 times per word, so the total voice is 20 times per volunteer. The collected data is then stored on Google Drive and accessed using Google Collab to compute the program code. Before preprocessing the data, the data is labeled by dividing the folders according to each word label, namely in one folder there are 4 folders with the labels *'inklusif'*, *'pecah'* , *'coba'*, and *'miring'* with 50 data in each folder.

B. Data Preprocessing

Data Preprocessing is conducted using TensorFlow. TensorFlow is used because it allows building large-scale neural network models with many layers and can be used for various purposes such as classification, perception, and prediction[23]. The sound data is processed with the tf.audio function to convert wav type sound files into waveforms which are later converted into spectrograms so they can be input to the CNN[24].

A spectrogram is an image representation of a waveform signal that shows its frequency intensity range over time[25]. Although spectrograms only display frequency over time, while waveforms display changes in amplitude over time, spectrograms can lose information about changes in amplitude from audio data or waveforms[26].

C. CNN Implementation

This stage implements the CNN method in program code via Google Collaboration using TensorFlow. Starting with determining the model to be created and then compiling the model. In the Convolutional Neural Network (CNN) architecture, the input layer is used with an image-resizing process to speed up model training[27]. The hidden layer consists of 2 convolutional layers with a maxPooling layer, followed by a dropout layer and a dense layer for image classification. However, in audio spectrogram conversion, Tensorflow does not have a feature to return to audio[28].

D. Analysis and Evaluating

The confusion matrix serves as an analytical tool by utilizing the `tf.math.confusion_matrix` function in TensorFlow, complementing the CNN's classification process1. This study compared data ratios (training:validation:test) of 60:20:20, 70:15:15, and 80:10:10 to determine the optimal result. the CNN model analysis can be done by looking at the loss and accuracy curves of the model that has been trained and checking what percentage of accuracy the model runs on test data[29]. This study finds that choosing a ratio that produces at least 80% accuracy is the optimum result.

III. Results and Discussion

An example of the output from this process can be seen in Figure 1, which shows the sound waveform in WAV format. This waveform is a graph of the time function of each displacement in sound, with the x-axis showing the amplitude and the y-axis showing the period [30]. Each word's sound amplitude differs, as seen in the inclusive word with a high amplitude at time 50000. In the case of consonants, it can be seen that the words '*pecah'* and *'coba'* have a silent sound in the middle of the spoken word.

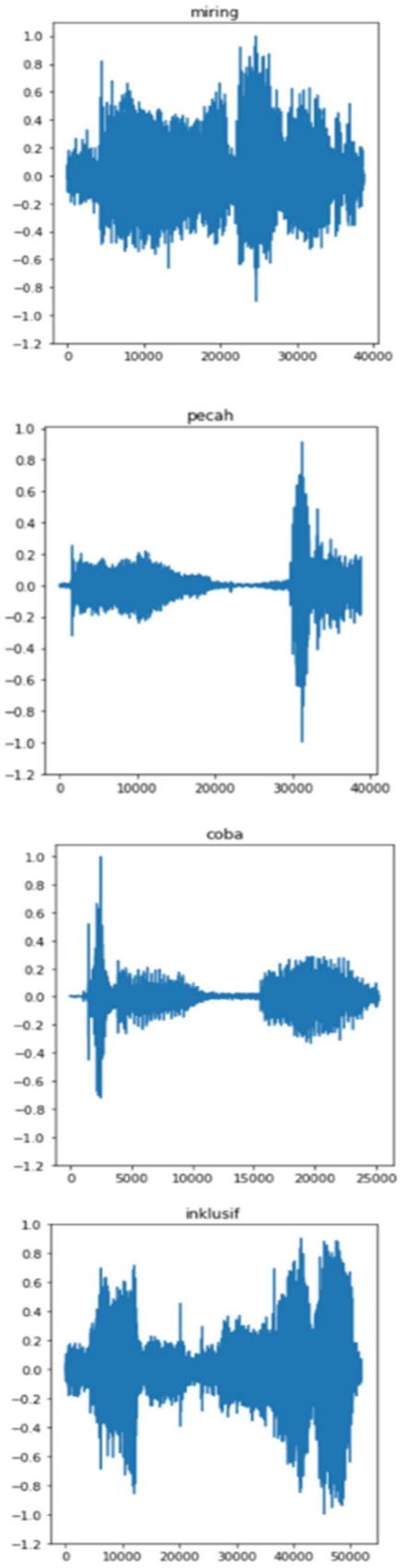

Fig. 1 Sample of sound data in waveform

A. Data Processing

At this stage, sound files are processed to be inserted into a Convolutional Neural Network (CNN). The first stage involves converting sound files in WAV format into waveforms, which are then converted into spectrograms so they can be input into the CNN. The next stage is converting the waveform into a spectrogram, which is shown in Figure 2, which shows a sample of conversion results from several sound files.

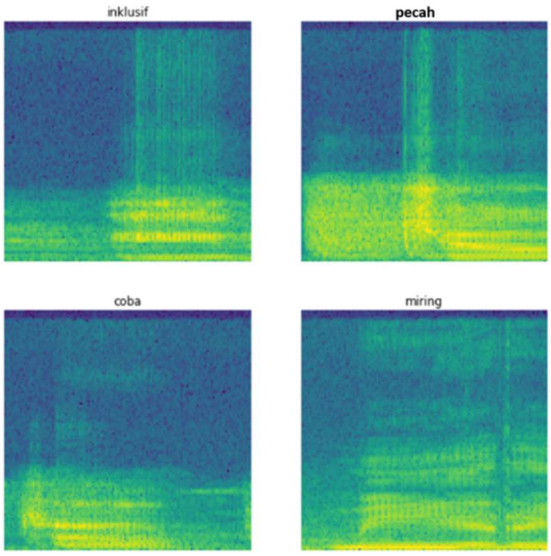

Fig. 2 Sample conversion result from waveform to spectrogram

Figure 2 demonstrates the conversion of waveforms into spectrograms for the four tested Indonesian words, which are; 'inklusif','pecah','coba', and 'miring' . Spectrograms display frequency content over time, with color intensity indicating energy levels. Brighter areas represent higher energy concentrations at specific frequencies, while darker areas indicate lower energy levels. This conversion is crucial for CNN processing as it transforms audio data into a format suitable for image-based analysis.

B. CNN Implementation

This stage involves implementing the CNN method in program code via Google Collaboration using Tensorflow. It starts by determining the model to be created and then compiling it. Next, the model is compiled using the Adam optimizer, which is inserted into the line of code as in Figure 3.

```
model.compile(
    optimizer=tf.keras.optimizers.Adam(),
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
    metrics=['accuracy'],
)
```

Fig. 3 Compile model using Adam optimizer

At this stage, model training is carried out using the prepared dataset. Model training is a process in which a machine learning algorithm is learned using an appropriate data set. The data is divided into three groups, namely training data, validation data, and test data. Training data is used specifically to train the model, while validation data is used to test the model during training. On the other hand, test data is hidden data used to test predictions after the model has been trained. Before training begins, the ratio between the three types of data is determined, with different ratio options. In this research, model training was carried out using 50 epochs. The line code as in Figure 4.

```
EPOCHS = 50
history = model.fit(
    train_ds,
    validation_data=val_ds,
    epochs=EPOCHS,
    callbacks=tf.keras.callbacks.EarlyStopping(verbose=3, patience=10),
)
```

Fig. 4 Train Model using 50 epoch

C. Results and Analysis

1) Loss and Accuracy. Model analysis is carried out by examining the loss and accuracy curves as well as the level of accuracy on the test data. Figure 5 shows at the 60:20:20 ratio, there is a significant decrease in loss after around the 30th epoch, with peak accuracy reached between the 20th and 30th epoch.

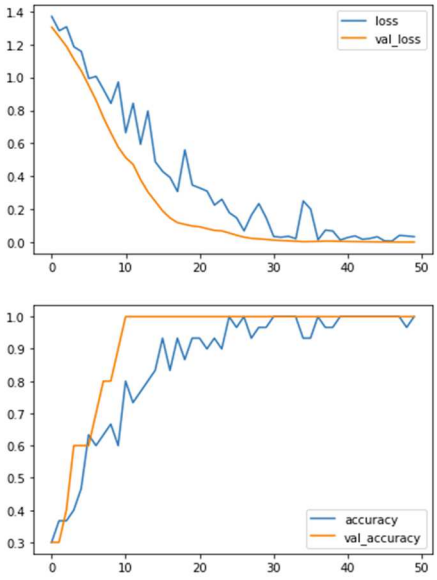

Fig. 5 Loss and accuracy analysis results with a ratio of 60:20:20

Meanwhile, Figure 6 shows that in the 70:15:15 ratio, there was a significant decrease in loss around the 40th iteration, with peak accuracy occurring between the 20th and 30th epochs.

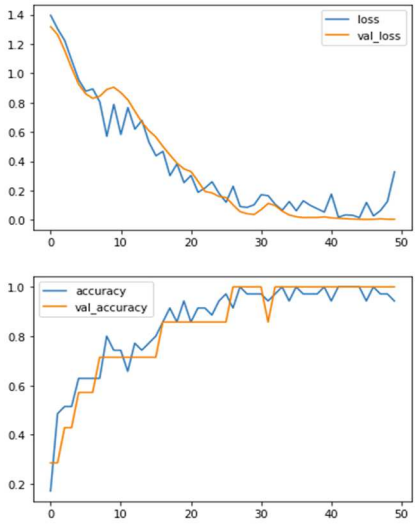

Fig. 6 Loss and accuracy analysis results with a ratio of 70:15:15

Figure 7 shows that at a ratio of 80:10:10, the lowest loss value occurs in the 30th iteration onwards, while the highest accuracy value occurs between the 30th and 40th iterations.

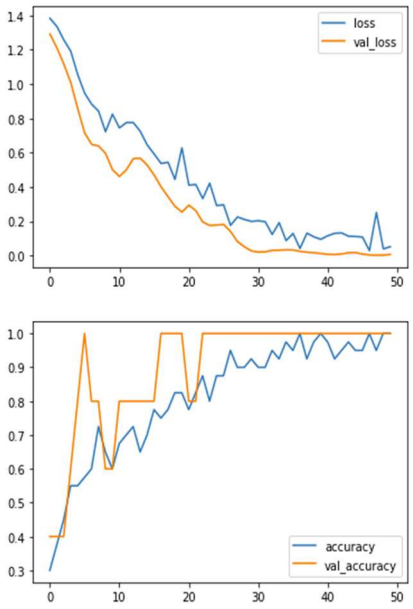

Fig. 7 Loss and accuracy analysis results with a ratio of 80:10:10

Table 2 compares three different ratios of training, validation, and test data. The 60:20:20 ratio achieved the best accuracy test result of 90%. The 70:15:15 ratio resulted in 86% accuracy, and the 80:10:10 ratio had the lowest accuracy at 80%. Table II suggests that a balanced distribution of data contributes to higher accuracy in the CNN model's performance.

TABLE II
ACCURACY RESULTS BETWEEN 3 DIFFERENT RATIOS

Ratio	Training Data	Validation Data	Test Data	Accuracy
60:20:20	30 data	10 data	10 data	90%
70:15:15	35 data	7 data	7 data	86%
80:10:10	40 data	5 data	5 data	80%

2) The Evaluation Performance of Confusion Matrix. The confusion matrix of predictions and labels is evaluated using tf.math.confusion_matrix in TensorFlow. It can be seen in Figure 8, Figure 9, and Figure 10. However, there is no confusion matrix display feature with true positive, true negative, false positive, false negative, precision, recall, and F1-score. Independent calculations were carried out in this study.

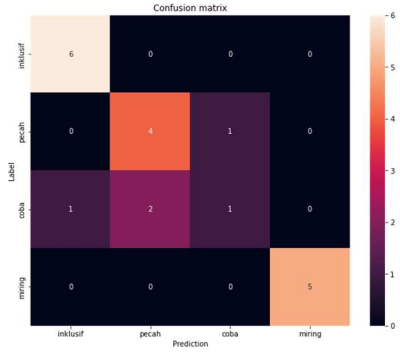

Fig. 8 Confusion matrix results with a ratio of 60:20:20

Figure 8 show that by using this model, the matrix shows the results for the words *'inklusif'*, *'pecah'*, *'coba'*, and *'miring'*, with *'miring'* achieving a perfect score in precision, recall, and F1-score.

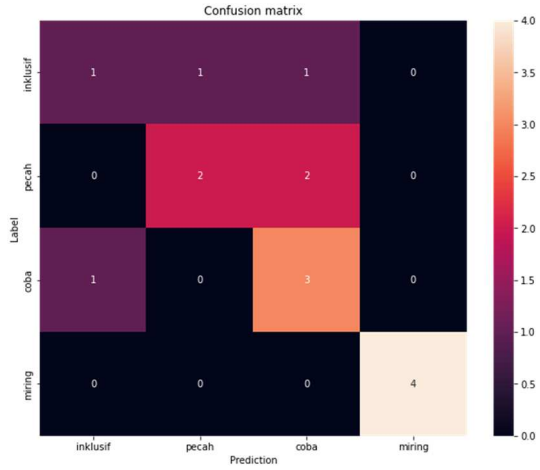

Fig. 9 Confusion matrix results with a ratio of 70:15:15

Figure 9, the matrix highlights the number of correct and incorrect predictions made by the model, with the word *'miring'* once again showing superior performance compared to the other classes.

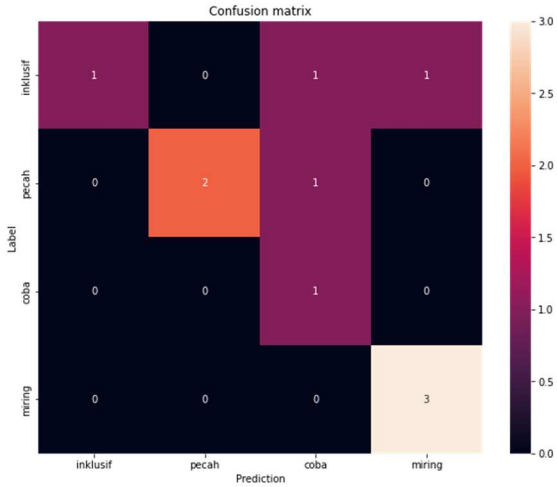

Fig. 10 Confusion matrix results with a ratio of 80:10:10

Figure 10 indicates that while the overall accuracy is lower with this ratio, the word *'miring'* and *'coba'* both achieve a precision of 100%, and the recall values for *'inklusif'* and *'pecah'* reach 100% as well. The F1-score for *'pecah'* is the highest among the words, at 80%.

The evaluation performance of a voice recognition system for Indonesian words using a Convolutional Neural Network (CNN)1. It focuses on four words: 'inklusif', 'pecah', 'coba', and 'miring'. Precision, recall, and F1-score are used as metrics. Precision measures the accuracy of positive predictions, recall measures the coverage of actual positive cases, and the F1-score is the harmonic mean of precision and recall. The evaluation results were presented on Table III, Table IV, and Table V.

TABLE III
CONFUSION MATRIX RATIO 60:20:20

Classes	*inklusif*	*pecah*	*coba*	*miring*
True positive	6	4	1	5
True negative	13	13	15	15
False positive	0	1	3	0
False negative	1	2	1	0
Precision	1	0.8	0.25	1
Recall	0.86	0.67	0.5	1
F1-score	0.91	0.73	0.33	1

Table III shows that the word 'miring ' in both precision and recall classes achieved perfect scores, which is the harmonic mean of precision and recall, showing a balanced accuracy. It also has the highest F1-score of 1, which is the harmonic means of precision and recall, showing a balanced accuracy. Compared to other words, *'miring'* had the best overall performance, with *'inklusif'* and *'pecah'* having lower precision and recall rates. The high scores for *'miring'* suggest that the CNN model was particularly effective at recognizing this word within the dataset.

TABLE IV
CONFUSION MATRIX RATIO 70:15:15

Classes	*inklusif*	*pecah*	*coba*	*miring*
True positive	1	2	3	4
True negative	11	11	8	11
False positive	2	2	1	0
False negative	1	0	3	0
Precision	0.33	0.5	0.75	1
Recall	0.5	0.5	0.5	1
F1-score	0.39	0.5	0.6	1

Table IV result show that the word *'miring'* have score 1 at precision, recall, and F1-score, indicating perfect classification for this word. The words *'inklusif'* and *'pecah'* have lower precision and recall values compared to *'miring'*, suggesting some misclassifications occurred. The F1 scores for 'inclusive' and 'pecah' are 0.39 and 0.5, respectively, which are measures of the test's accuracy.

The high performance for 'miring' suggests that the CNN model is particularly effective at recognizing this word within the dataset. The lower scores, in other words, indicate areas where the model's recognition capabilities could be improved. While this ratio is highly accurate for some words, there is room for improvement in its overall word recognition accuracy, especially for words with lower precision and recall values.

TABLE V
CONFUSION MATRIX RASIO 80:10:10

Classes	*inklusif*	*pecah*	*coba*	*miring*
True positive	1	2	1	3
True negative	7	7	7	6
False positive	2	1	0	0
False negative	0	0	2	1
Precision	0.33	0.67	1	1
Recall	1	1	0.33	0.75
F1-score	0.49	0.8	0.49	0.21

Table V shows that the words *'miring'* and *'coba'* have a score of 1 or 100% precision, indicating that when the model predicted these words, it was always correct. The recall for *'inklusif'* and *'pecah'* was also 100%, meaning all instances of these words were correctly identified by the model. The F1-score for *'pecah'* was the highest at 0.8, suggesting a balance between precision and recall. However, the F1-score for *'miring'* was only 0.21, indicating a potential issue with the balance between precision and recall for this word.

With some words like *'miring'* and *'coba'* performing exceptionally well in precision, while others like *'inklusif'* excelled in recall. This variation indicates that the model's ability to recognize words depends on the specific characteristics of each word.

This study found that the 60:20:20 ratio yielded the best results due to a balanced distribution of data, which is crucial for practical training, validation, and testing phases, along with several key points.

1) Balanced Training: The 60:20:20 ratio gives the optimal amount of information from which the model can learn (train), validate its learnings (validate), and test its performance (test). Striking this balance contributes to improved accuracy and enhanced generalization.

2) Loss and Accuracy: Although the peak accuracy was reached between the 20th and 30th epochs, the loss was significantly lower after around the 30th epoch. This means that the model can learn well and make reasonable guesses based on unseen data.

3) Confusion Matrix: The confusion matrix of the 60:20:20 ratio had high precision, recall, and F1-scores for each word as well as it did for the word 'miring', where it got perfect scores. So, it means that the model is well enough to make true predictions in general but also true predictions as positive.

4) Precision and recall: The high precision and recall for most words indicate that the model is reliable when predicting the correct class and effectively covers actual positive cases.

These are the implications of achieving 90% accuracy using the 60:20:20 ratio to improve Indonesian universities' educational access. This recognition could be a huge step toward improving the classes of deaf and hard-of-hearing students on campus, especially the deaf students at UIN Sunan Kalijaga. Currently, most of these students encounter significant obstacles with lecture material, for instance, through a lack of available sign language interpreters or having trouble translating technical or foreign terminology. The CNN-based system we developed is a good alternative since it provides real-time voice-to-text conversion.

IV. CONCLUSION

From our analysis of different ratios, the performance of the CNN model varied in crucial ways about its configuration during training. The 60:20:20 combination was the most accurate setup, with 90%, while being quite balanced for all test words. Among them, 'miring' performed perfectly in all metrics, but 'inclusive' was good, with a high F1-score of 0.91. Some interesting patterns came in with the 70:15:15 ratio. Whereas 'miring' kept its perfect recognition rate; other words, it did not. The most noticeable case was 'inklusif', whose precision degraded to 0.33 and recalled to 0.5. Its decline indicates that a reduction of validation impairs the model's fine-tuning ability.

The most surprising finding, however, is that despite having the most extensive training set, the 80:10:10 ratio provided the most unreliable results. Although terms such as 'miring' and 'coba' had a perfect precision of 1, their F1 Scores were low, with 'miring' yielding an F1 Score of only 0.21. This is a very important lesson: more training data is not always better, especially when it comes to more minor validation and testing sets.

There is still much work to be done in the future: We will increase our dataset even more by adding words and testing the system in real classroom environments. Noise reduction techniques can be included to enhance performance in practice. Another interesting investigation could be the study of other neural network architectures, like RNNs or hybrid models, which may improve the results even further.

More importantly, the research points out how artificial intelligence can be used to solve real-world accessibility challenges. Since its development focused on practical fields and was done with a high level of accuracy, we are in a position to assist in building an inclusive educational system for all students in Indonesia.

REFERENCES

[1] U. Rahma, Z. Hikmiah, and T. H. Firmanda, "Pemetaan Kebutuhan Pendampingan Konseling: Study of Psychological Wellbeing on Students with Disabilities," *INKLUSI*, vol. 9, no. 1, pp. 21–44, Aug. 2022, doi: 10.14421/ijds.090102.

[2] D. Cazzani, "Audio processing in TensorFlow. An implementation of the Short Time...," *Towards Data Science*, May 06, 2024. [Online]. Available: https://towardsdatascience.com/audio-processing-in-tensorflow-208f1a4103aa. Accessed: May 06, 2024.

[3] S. M. Al Sasongko, S. Tsaury, S. Ariessaputra, and S. Ch, "Mel Frequency Cepstral Coefficients (MFCC) Method and Multiple Adaline Neural Network Model for Speaker Identification," *JOIV : International Journal on Informatics Visualization*, vol. 7, no. 4, pp. 2306–2312, 2023, doi: 10.30630/joiv.7.4.01376.

[4] S. Poudel and D. R. Anuradha, "Speech Command Recognition using Artificial Neural Networks," *JOIV : International Journal on Informatics Visualization*, vol. 4, no. 2, pp. 73–75, 2020, doi:10.30630/joiv.4.2.358.

[5] W. Mustikarini, R. Hidayat, and A. Bejo, "Real-Time Indonesian Language Speech Recognition with MFCC Algorithms and Python-Based SVM," *IJITEE (International Journal of Information Technology and Electrical Engineering)*, vol. 3, no. 2, pp. 55–60, 2019, doi: 10.22146/ijitee.49426.

[6] "Convolutional Neural Network based Audio Event Classification," *KSII Transactions on Internet and Information Systems*, vol. 12, no. 6, 2018, doi: 10.3837/tiis.2018.06.017.

[7] A. A. Khamees, H. D. Hejazi, M. Alshurideh, and S. A. Salloum, "Classifying Audio Music Genres Using CNN and RNN," *Advances in Intelligent Systems and Computing*, vol. 1339, pp. 315–323, 2021, doi: 10.1007/978-3-030-69717-4_31.

[8] D. C. Khrisne and T. Hendrawati, "Indonesian Alphabet Speech Recognition for Early Literacy using Convolutional Neural Network Approach," *Journal of Electrical, Electronics and Informatics*, vol. 4, no. 1, pp. 34–37, Feb. 2020, doi: 10.24843/JEEI.2020.V04.I01.P06.

[9] H. F. Pardede, P. Adhi, V. Zilvan, A. Ramdan, and D. Krisnandi, "Deep convolutional neural networks-based features for Indonesian large vocabulary speech recognition," IAES International Journal of Artificial Intelligence (IJ-AI), vol. 12, no. 2, p. 610, Jun. 2023, doi:10.11591/ijai.v12.i2.pp610-617.

[10] H. Liang, X. Lin, Q. Zhang, and X. Kang, "Recognition of spoofed voice using convolutional neural networks," in *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2017, pp. 293–297. doi: 10.1109/GlobalSIP.2017.8308651.

[11] E. Franti *et al.*, "Voice Based Emotion Recognition with Convolutional Neural Networks for Companion Robots," 2018.

[12] J.-H. An, N.-K. Koo, J.-H. Son, H.-M. Joo, and S. Jeong, "Development on Deaf Support Application Based on Daily Sound Classification Using Image-based Deep Learning," *JOIV : International Journal on Informatics Visualization*, vol. 6, no. 1–2, pp. 250–255, 2022, doi: 10.30630/joiv.6.1-2.936.

[13] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights Imaging*, vol. 9, no. 4, pp. 611–629, Aug. 2018, do: 10.1007/s13244-018-0639-9.

[14] W. Rawat and Z. Wang, "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review," *Neural Comput*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017, doi: 10.1162/neco_a_00990.

[15] M. Kubanek, J. Bobulski, and J. Kulawik, "A Method of Speech Coding for Speech Recognition Using a Convolutional Neural Network," *Symmetry (Basel)*, vol. 11, no. 9, p. 1185, 2019, doi:10.3390/sym11091185.

[16] S. Skansi, "Convolutional Neural Networks," 2018, pp. 121–133. doi:10.1007/978-3-319-73004-2_6.

[17] E. Beauxis-Aussalet and L. Hardman, "Simplifying the visualization of confusion matrix", *Proc. BNAIC*, pp. 1-2, Nov. 2014.

[18] H. Phan, L. Hertel, M. Maass, and A. Mertins, "Robust Audio Event Recognition with 1-Max Pooling Convolutional Neural Networks," *arXiv preprint arXiv:1604.06338*, 2016. [Online]. Available: https://arxiv.org/abs/1604.06338

[19] O. Kembuan, G. Caren Rorimpandey, and S. Milian Tompunu Tengker, "Convolutional Neural Network (CNN) for Image Classification of Indonesia Sign Language Using Tensorflow," in *2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS)*, 2020, pp. 1–5. doi:10.1109/icoris50180.2020.9320810.

[20] J. Xu, Y. Zhang, and D. Miao, "Three-way confusion matrix for classification: A measure driven view," *Inf Sci (N Y)*, vol. 507, pp. 772–794, Jan. 2020, doi: 10.1016/J.INS.2019.06.064.

[21] T. Carneiro, R. V. Medeiros Da Nobrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. R. Filho, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. 6, pp. 61677–61685, 2018, doi:10.1109/access.2018.2874767.

[22] E. A. W. Hachim, M. T. Gaata, and T. Abbas, "Voice-Authentication Model Based on Deep Learning for Cloud Environment," *JOIV : International Journal on Informatics Visualization*, vol. 7, no. 3, pp. 864–870, 2023, doi: 10.30630/joiv.7.3.1303.

[23] X. Zheng, M. Wang, and J. Ordieres-Meré, "Comparison of Data Preprocessing Approaches for Applying Deep Learning to Human Activity Recognition in the Context of Industry 4.0," *Sensors*, vol. 18, no. 7, p. 2146, Jul. 2018, doi: 10.3390/s18072146.

[24] Y. Gong and C. Poellabauer, "How do deep convolutional neural networks learn from raw audio waveforms?," 2018. [Online]. Available: https://openreview.net/forum?id=S1Ow_e-Rb.

[25] D. D. Oliveira, M. Rampinelli, G. Z. Tozatto, R. V. Andreão, and S. M. T. Müller, "Forecasting vehicular traffic flow using MLP and LSTM," *Neural Comput Appl*, vol. 33, no. 24, pp. 17245–17256, Dec. 2021, doi: 10.1007/s00521-021-06315-w.

[26] R. A. Solovyev *et al.*, "Deep Learning Approaches for Understanding Simple Speech Commands," in *2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO)*, IEEE, Apr. 2020, pp. 688–693. doi: 10.1109/elnano50318.2020.9088863.

[27] G. Parmar, R. Zhang, and J.-Y. Zhu, "On Aliased Resizing and Surprising Subtleties in GAN Evaluation," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2022, pp. 11400–11410. doi:10.1109/CVPR52688.2022.01112.

[28] P. Alonso-Jimenez, D. Bogdanov, J. Pons, and X. Serra, "Tensorflow Audio Models in Essentia," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, May 2020, pp. 266–270. doi:10.1109/icassp40776.2020.9054688.

[29] K. Palanisamy, D. Singhania, and A. Yao, "Rethinking CNN Models for Audio Classification," *arXiv preprint arXiv:2007.11154*, 2020. [Online]. Available: https://arxiv.org/abs/2007.11154.

[30] J. J. Benedetto, I. Konstantinidis, and M. Rangaswamy, "Phase-Coded Waveforms and Their Design," *IEEE Signal Process Mag*, vol. 26, no. 1, pp. 22–31, 2009, doi: 10.1109/MSP.2008.930416.