p-ISSN 2301–4156 | e-ISSN 2460–5719

Teachable Machine: East Sumba Dialect (Kambera) Detection Using Open Source Services

Edwin Ariesto Umbu Malahina

D3 Informatics Engineering Study Program, STIKOM Uyelindo Kupang, Nusa Tenggara Timur 85111 INDONESIA (tel.: 0380 – 8430672; fax.: 0380-8430672; email: edwinariesto@gmail.com)

[Received: 29 May 2023, Revised: 10 October 2023]
Corresponding Author: Edwin Ariesto Umbu Malahina

ABSTRACT — This research seeks to develop a phonetic detection system for the Kambera dialect, the East Sumba local language, based on the TensorFlow framework that will be implemented in mobile applications. As part of this initiative, this research compiled a representative dataset of Kambera dialect phonetic samples. The main objective is to improve precision in phonetic recognition. Using the Kambera dialect as a case study, the data were extracted and trained using the open-source Teachable Machine service. This research adopted a convolutional neural network (CNN)-based approach combined with the Mel-frequency cepstral coefficients (MFCC) method for more accurate feature extraction. After data collection, model training, testing, and implementation, the model was integrated into the Android platform to benefit the public who wished to understand the Kambera dialect of East Sumba. The development and testing of this system were designed to detect and interpret the phonetics of the local language of East Sumba with the Kambera dialect, making a significant contribution to optimizing phonetic recognition and providing a dataset for ongoing research interests. It also serves as an accessible linguistics educational tool and supports linguistic inclusion and diversification in digital technology. Empirical evaluation showed that the overall average dialect detection precision rate reached 98.3% to 99.6%, with the user satisfaction rate reaching 99.33%. These results confirm that the developed system has a very efficient and good detection capability.

KEYWORDS — Convolutional Neural Network, Mel Frequency Cepstral Coefficients, Teachable Machine, Voice Detection, East Sumba, Open Source, TensorFlow, Kambera Dialect.

I. INTRODUCTION

Open source-based services are a type of service or software that uses open source. Its primary purpose is to allow developers and communities to contribute to fixing bugs and adding and utilizing features or functions [1], [2]. The service can also collaborate with other different services [3]–[5]. Teachable Machine is an open-source platform that facilitates the development of classification models in the context of machine learning by leveraging the TensorFlow library. It adopts deep learning techniques in artificial neural networks to recognize speech patterns. In this study, the researcher focused on simulating phonetic recognition of East Sumbanese regional languages, paying particular attention to its basic terminology. Previous studies have validated the efficacy of the Teachable Machine, noting impressive levels of detection accuracy, precision, and sensitivity, ranging from 90% to 100% [6]–[8]. This excellent level of accuracy makes the Teachable Machine a recommended service in this research.

This research identifies a critical issue that becomes the focus of this research topic, namely the urgency of preserving local languages as an integral part of the nation's cultural heritage. In Indonesia, 718 documented local languages constitute cultural heritage that must be preserved and protected. A number of local languages, however, are currently facing a deep crisis. They are experiencing a significant decline in the number of native speakers and are no longer being passed down from generation to generation. These matters pose a risk of losing valuable linguistic heritage, which could erode the nation's cultural richness and diversity. Therefore, there is an urgent need to take proactive measures in ensuring the transmission of local languages to the next generation in order to maintain linguistic diversity and promote the preservation of local languages [9], especially East Sumba language which consists of several dialects: Rindi, Lumbu Manggit, Kambera, Melolo, Uma Ratu Nggai (Umbu Ratu Nggai), Lewa, Kanatang, Mangili-Waijelo (Wai Jilu, Waidjelu, Rindi, Waijelo), and South Sumba [10]. These local languages and dialects are presently facing the crisis of losing native speakers and transmission to the next generation. As a solution, this research proposes the utilization of artificial intelligence through the Teachable Machine service, allowing the development of a machine learning-based speech detection system by utilizing the TensorFlow library. The research creates the potential to develop regional language voice datasets that can facilitate further research in local language preservation efforts and optimize speech recognition by combining the Mel-frequency cepstral coefficients (MFCC) method within the TensorFlow framework.

Unlike the previous research, which focuses on the concept of dialect mapping [11]–[13], politeness in language [14], language history [15], and Anthropolinguistics lingual forms [16], the current research focuses on optimizing dialect detection capabilities that combine the convolutional neural network (CNN) algorithm in the TensorFlow framework with the MFCC method approach, which is tested for comparison and accuracy. In an effort to preserve East Sumba regional languages and prevent language extinction by leveraging intelligent systems that can contribute to the data collection of East Sumba regional languages, this research also focuses on the digitalization of Kambera dialects in the form of applications and the creation of sample datasets (databases). This research is also expected to be beneficial for anyone interested in learning East Sumba languages. In addition, it does not rule out the possibility of further research to utilize and further develop the offered datasets.

The discussion of problems and literature studies have assisted the researcher in developing and contributing to the process of optimizing voice detection using MFCC, which is added to the TensorFlow framework. As a result, the developed mobile-based software can detect Kambera dialects more accurately.

II. METHODOLOGY

The methodology adopted in the data collection phase followed an exploratory approach to the Kambera dialect. This process generated data in the form of conversation transcripts and audio recordings obtained from three local community resource persons with expertise in the Kambera dialect. The primary objective of this approach is to identify and document variations in vocabulary, pronunciation, grammatical structure, intonation, and other linguistic features of the Kambera dialect. This analysis seeks to distinguish and understand the unique nuances of the Kambera dialect, as well as lay the groundwork for future linguistic research.

In software development and testing, method techniques utilizing the services of the TensorFlow framework were used. This framework provides the necessary infrastructure and has the advantage of efficient development of machine learning code and models [17], [18]. The dataset training process in TensorFlow was controlled by CNN, which was a data classification method [19]; in this context, dialect voice sample data were extracted, and their features were classified separately, thus forming a learning-based detection system [20]. In this research, Kambera linguistic data were trained using the Teachable Machine. Teachable Machine provided a CNN algorithm that could be used to classify Kambera dialect voices to obtain the accuracy rate in percentage form. Adding the MFCC method helped in better feature extraction from the voice data, which can improve the accuracy of speech recognition.

A. TRANSFER LEARNING MODEL

The TensorFlow.js framework is an open-source library for machine learning that operates in a JavaScript environment. It allows developers to incorporate machine learning elements into web and mobile applications and is compatible with a wide variety of JavaScript-supporting devices [21]. Transfer learning is a prior knowledge approach from machine learning models, such as CNNs, that is reused to understand new and similar tasks. In this context, models that have been trained in speech recognition can be reused for other speech classifications, saving training time and resources.

CNN has been applied to research in various fields with different case studies of voice detections. This research includes voice detection to classify voices [22], [23], detect depressive or emotional voices [24], [25], detect music [26], realize voice-based security systems [27], [28], detect language phonemes [29], [30], detect disease by sounds [31]–[34], and identify animal voices [35], [36]. There are many other research case studies that can be found and used as ongoing future research.

Figure 1 shows the CNN architecture starting on the left (red box), with each layer comprising a number of neurons (white circles). A multi-layer perceptron (MLP) receives one-dimensional input data and sends it through the network to produce an output. Each connection between neurons in two adjacent layers has a one-dimensional weight parameter that affects the quality of the model. At each layer, the input data undergoes a linear operation by taking into account the existing weight values. Then, the computational results are transformed using a nonlinear activation function [37].

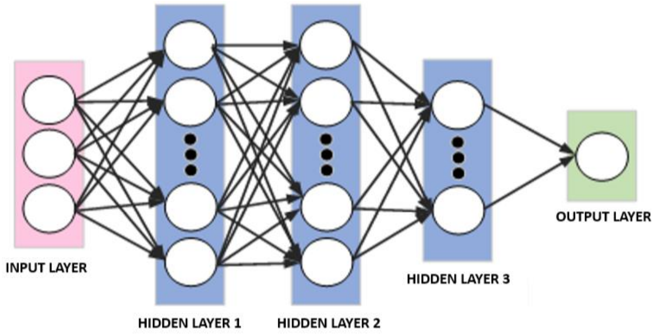

Figure 1. Convolutional neural network architecture.

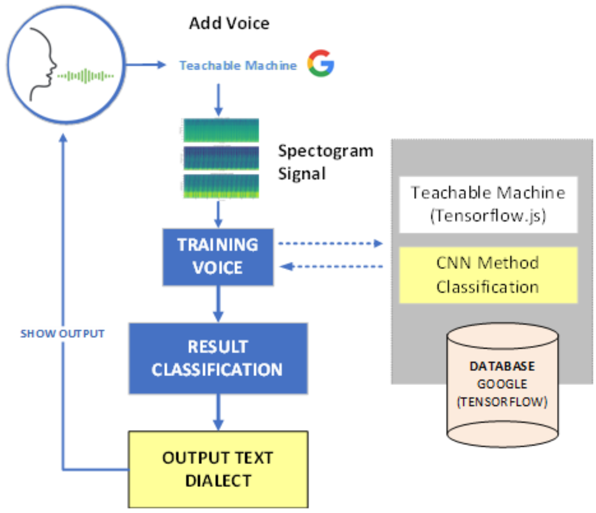

Figure 2. Flowchart of managing voice dialect data into Teachable Machine website.

B. SYSTEM WORKFLOW

In this research, the audio recognition process of the Kambera dialect in the East Sumba language used the TensorFlow framework from Teachable Machine service as a data processing tool. The concept used in this tool is based on the principle of machine learning, where a system can learn and analyze data without having to be explicitly programmed. It aims to help various groups understand the concept of machine learning workflow by creating and using the developer's classification experiment model independently [38].

This work system started with preparing East Sumba language voice samples of the Kambera dialect for voice classification data. The step was continued by providing audio noise samples to eliminate noise during testing. Then, voice samples and audio noises were imported into the Teachable Machine and then converted into spectrograms. Automatic data training in the Teachable Machine was then conducted, followed by testing the accuracy. If successful, the data were exported to JavaScript to be developed into an application.

C. MODEL TRAINING

A program flowchart serves as a visual representation of a computer program through flowchart symbols. This type of flowchart describes the process steps of a program in detail and shows the steps of solving problem units that are interrelated and interact with each other to achieve certain goals. This flowchart shows the flow of the system, or the functions performed by the system.

Process flowchart is depicted in Figure 2. The process began with the user inputting the voice of the East Sumba language Kambera dialect to the Teachable Machine

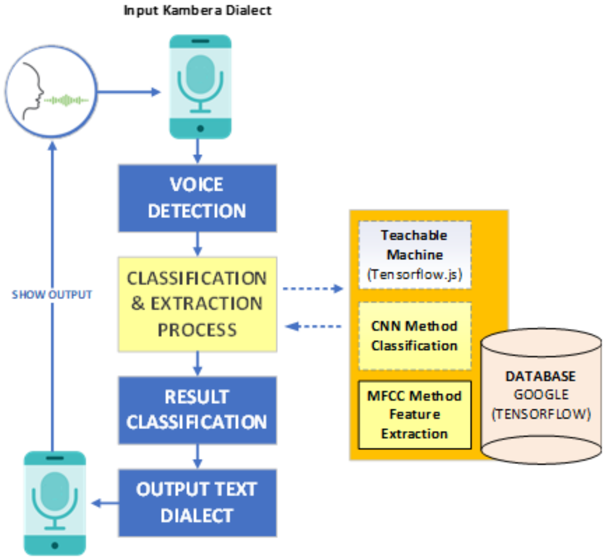

Figure 3. Flowchart of users performing the Kambera dialect detection with the developed mobile web application.

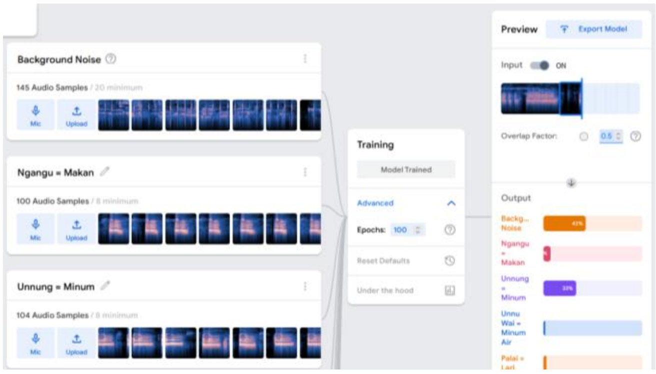

Figure 4. Voice dialect data is pre-trained on the Teachable Machine service.

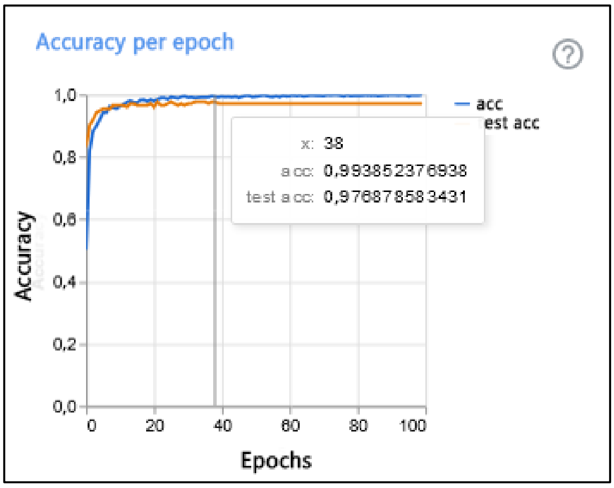

Figure 5. Accuracy per epoch.

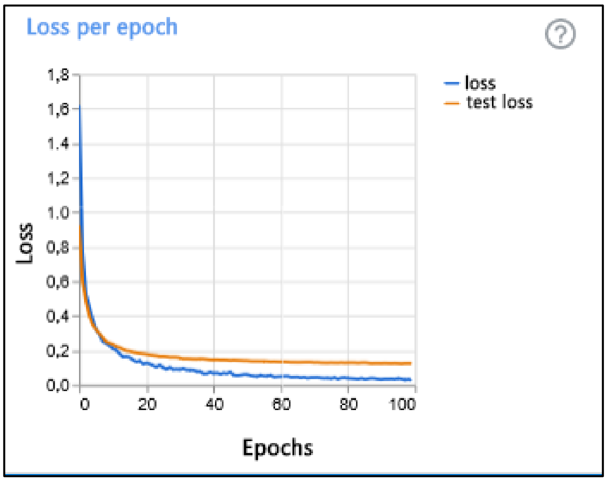

Figure 6. Loss per epoch.

application to be stored as an audio signal and audio noise. The signal was then converted into a spectrogram and noise signal to detect and remove voice noise later. Next, the voice was trained for speech recognition. The training was managed by the CNN algorithm for classification (without the MFCC method for audio feature extraction). The result data were stored in the Google database, and the system generated voice classification. Finally, the classification results were displayed as text according to the user's voice input.

The results of the process, as in Figure 2, were then exported into a TensorFlow.js file. They were manually added to the MFCC method function for the audio feature extraction.

Based on Figure 3, the process commenced when the user entered the voice into the mobile application. Next, the voice was processed processing and identified using the CNN algorithm, supported by voice feature extraction using the previous Mel MFCC method in the TensorFlow.js file. The addition of the MFCC method aims to improve the quality of existing voice data, thereby improving the accuracy of voice recognition. If the identification result was successful, the mobile application displayed the text matched the accuracy of the speech recognition presentation to the text articulation of the Kambera language entered.

III. RESULT AND DISCUSSION

A. TESTING THE VOICE DIALECT MODEL

Testing the Kambera voice dialect model in Teachable Machine involved using a separate dataset containing 100 samples per vocabulary to test the accuracy and performance of the voice model. Then, the voice samples were inputted into the trained Teachable Machine, and the model classified the voice into the predetermined label (audio class name description). The purpose of this test is to evaluate the model's ability to recognize and classify sounds with high accuracy, and to provide information about the reliability and effectiveness of the model.

In Figure 4, in the context of the framework provided by Teachable Machine, the research involved comprehensive testing of 100 spectrograms representing each pertinent dialect label. Voice sampling was conducted following a standard recording duration of 2 s per dialect sample, with the option to re-record if necessary. Next, the trained model using the Teachable Machine platform was tested using predefined dataset classes. The model's performance was evaluated by analyzing several metrics including accuracy, precision, and sensitivity. It aims to measure the extent to which the model can accurately recognize and classify various dialect variants, ensuring the validity and effectiveness of the developed solution in the context for the Kambera dialect speech recognition.

In the context of speech recognition using Teachable Machine, "accuracy per epoch" refers to a metric that measures the percentage of the amount of data correctly classified by the model during one training cycle (epoch). In contrast, "loss per epoch" measures the magnitude of the model's prediction error on the training data during one epoch. The goal in model

TABLE I
DIALECT DETECTION TESTING USING CNN ALGORITHM OF THE TENSORFLOW AND MFCC METHOD

No.	Meaning of Dialect (Indonesian Language)	Kambera Dialect (East Sumba)	Dataset	CNN - TensorFlow		MFCC	
				Min Accuracy (%)	*Max Accuracy (%)*	*Min Accuracy (%)*	*Max Accuracy (%)*
1	Makan	Ngangu	100	99	100	99	100
2	Minum	Unnung	100	90	100	99	100
3	Minum Air	Unnu Wai	100	79	89	98	98
4	Lari	Palai	100	91	98	97	100
5	Seperti	Tuna Kadi	100	96	100	98	100
6	Pergi	Laku	100	85	99	97	99
7	Kesana	Laku Nolua	100	98	100	98	99
8	Kemari	Mai Yohu	100	94	99	99	100
9	Nama Saya	Na Tamu Ngu	100	98	98	99	100
10	Kamu	Nyummu	100	93	99	99	100
Average Accuracy			1,000	92.3 %	98.2 %	98.3%	99.6%

training is to increase the accuracy per epoch while reducing the loss per epoch, which can be achieved through adjusting model parameters such as the learning rate or changes to the model architecture to ensure that the model can learn and classify sounds better according to the desired application goals. The following results can be seen in Figure 5 and Figure 6.

In Figure 5, a screenshot of the accuracy per epoch graph on the Teachable Machine website depicts the “acc” metric, which measures the training model’s accuracy and indicates how well the model predicts the training data. The value of “acc” ranged from 0.97 to 1, meaning that the model had an accuracy between 97% and 100% against the training data. Meanwhile, “test acc” is the validation accuracy that measures how well the model predicts test data that are not used in training. The “test acc” value ranged from 0.97 to 0.99, indicating an accuracy between 97% and 99% against the validation data.

Figure 6 shows the screenshot of a graph of loss per epoch in the Teachable Machine site, illustrating the ability of the model to make predictions on the entire dataset during each training iteration. The “loss” is a metric that measures how far the model’s prediction is from the true value and the lower the loss value, the better the model is at making predictions with a range of loss values between 0.07 to 0.03 (or 7% to 3%).

In addition, test loss measures the model error on test or each epoch’s validation data. This test loss is important to evaluate the extent to which the model can apply learning from training data to data that has not been used before. The test loss results showed a consistent and good decrease in the model error, with a range of values between 0.66 to 0.35 (or 66% to 35%) on the validation data.

B. VOICE DIALECT LIVE TESTING

Ten samples of the Kambera dialect were used to conduct an empirical evaluation of the developed application. Initially, tests were conducted using the TensorFlow framework with the existing CNN algorithm to measure the performance of dialect speech recognition. Although initial results showed an acceptable level of performance, with some samples still recording below accuracy levels, namely “Unnu Wai” with a minimum accuracy of 79% and “Laku” with 85%, there was room for significant improvement. In response to these findings, the MFCC method was applied in an integrated manner to transform the voice signal into a set of coefficients that better represent the characteristics of the voice.

Implementing this strategy by adding the MFCC method resulted in a marked improvement in speech recognition accuracy, reaching a level of at least 97%. Through an in-depth exploration of the combination of CNN architecture and MFCC feature extraction, this research successfully created a reliable and accurate speech recognition model. The findings confirmed that the collaboration between CNN and MFCC could effectively improve the recognition of Kambera speech dialects, increasing the stability and accuracy of the phonetic identification process. Further analysis showed that the integration of MFCC helped reduce phonetic ambiguity and intraspecies variability in the voice data, facilitating more precise dialect identification.

The main advantage of this approach lies in how CNN can take advantage of the spatial feature hierarchy of the voice data, while MFCC, with its ability to characterize the frequency properties of the voice, provides a good and informative representation of the inherent phonetic characteristics. The synergistic combination of CNN and MFCC creates a new paradigm in the field of speech recognition, where these two methods complement each other to create a robust framework for dialect variety identification with a high degree of accuracy. In this context, the MFCC method acts as an efficient tool to reduce the dimensionality of speech data, while the CNN architecture takes advantage of the local and global structure of the data to perform effective pattern recognition, thus demonstrating how the strategic integration between CNN and MFCC can serve as a strong foundation for futuristic developments in speech recognition technology, especially in multilingual and multidialectal contexts.

This research also considered how variety in the training data could affect the model’s performance. The research found that the resulting model could better understand and generalize the various dialects of Kambera voices by having a diversified training dataset, thus significantly improving speech recognition performance. This research has important implications for future research in speech recognition, especially in the effort to reduce the technological gap among different dialects and languages.

Table I shows ten sample datasets of Kambera vocabulary dialects tested using an average dataset of 100 spectrogram samples. There was a minimum accuracy of each dialect and a minimum average dialect accuracy correction of 92.3% (less accurate). Using default or standard TensorFlow, the maximum average accuracy of the tested dialects was 98.2%; however,

JURNAL NASIONAL TEKNIK ELEKTRO DAN TEKNOLOGI INFORMASI
p-ISSN 2301–4156 | e-ISSN 2460–5719

TABLE II
USER TESTING

No.	Question	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
		5	*4*	*3*	*2*	*1*
1	Is the user interface in the app easy to use and understand the menus?	10	5	5	0	0
2	Is the input voice detection easily recognized by the application system?	6	12	1	1	0
3	Is the application very fast in the process of detecting the dialect sounds entered?	16	4	0	0	0
4	Is the developed application useful for the general public?	2	6	10	2	0
5	Does the accuracy level affect the results of user voice detection?	16	4	0	0	0
6	Can the utilization of intelligent repository systems help in the preservation of local languages, especially the Kambera dialect from East Sumba?	2	18	0	0	0
7	Were you helped by the system that was developed?	7	12	1	0	0
Total		59	61	17	3	0
Likert formula = $T \times Pn$ (Number of respondents × Likert score series)		295	244	51	6	0

Figure 7. Application interface when first opened.

adding the MFCC Method contributed to extracting improved features from the voice data, thereby increasing the voice accuracy to 98.3%–99.6%. It is more improved than testing using the default standard TensorFlow using the CNN algorithm, so that the combination of these two methods is optimal for enhancing the accuracy of voice detection.

Several essential points are obtained during this research.

1. The same dialect voice is only recorded for 2 s for the division of spectrogram samples (cannot be more).
2. The dialect voice must be clear and loud to produce accurate audio waves.
3. Noise is very influential with the results of percentage accuracy.
4. Vocabulary that is similar to the existing dataset affects accuracy.
5. The number of class samples will affect the processing time of training data results.

Table II presents data on user testing. This testing was conducted using questionnaires distributed to two groups of respondents: fifteen people from randomly selected general users and five people who are expertise in the Kambera dialect. The purpose of this test was to assess the accuracy of the system's recognition of the Kambera dialect. The data from this questionnaire were analyzed using the Likert scale method, involving certain calculation steps according to the formula described in (1).

$$T \times Pn \tag{1}$$

where T is the total number of respondents and Pn is the number of Likert score options.

Strongly agree	= 59 × 5 = 295
Agree	= 61 × 4 = 244
Neutral	= 17 × 3 = 51
Disagree	= 3 × 2 = 6
Strongly disagree	= 0 × 0 = 0

If all values are totaled, they result in a value of 596.

The test results through questionnaire data using Likert scale calculations resulted in a value of 596. Furthermore, this value was divided by the total number of questions totaling six questions (as listed in Table II). The results of this calculation resulted in a percentage of 99.33%, falling into the "Strongly Agree" category for the level of agreement from respondents to the questions asked in the questionnaire.

C. SYSTEM TESTING ANDA IMPLEMENTATION

The following is the result of the development of a voice dialect recognition system that has been integrated into a mobile web application. This application is designed to facilitate people in learning and understanding the Kambera dialect, which is one of the dialects of the East Sumba language.

Figure 7 shows the initial interface of the Kambera dialect recognition application when first accessed by the user. This page contains illustrations representing the island of East Sumba, as well as a "MULAI" button that, when clicked, directs the user to the East Sumba language's Kambera dialect recognition detection page. That page is the main process of application, where the user can directly say Kambera sentence and later the system responds and recognizes their speech in order to match it to the Google TensorFlow database, regardless of whether it is recognized. The application is designed with an intuitive and easy-to-use user interface, allowing users with varying levels of technological proficiency to interact with the system effectively. By integrating machine learning-based speech recognition technology, the application

Figure 8. Voice to text conversion and accuracy based on the Kambera.

offers an innovative solution to the challenge of dialect recognition in multilingual regions such as East Sumba. The use of Google's TensorFlow database in the system's infrastructure demonstrates the utilization of cloud technology and machine learning to address local problems with global resources.

Figure 8 displays the page for the Kambera dialect recognition. The user can start the speech recognition process by clicking the "UCAP" button and then speaking vocabulary from the Kambera dialect into the application. The application, which is equipped with automatic speech recognition technology, will analyze the voice input and, if recognized, will display the Indonesian translation along with the percentage accuracy of the recognition. It facilitates the user's understanding of the vocabulary of the Kambera dialect of East Sumba by providing percentage-based accuracy feedback.

IV. CONSLUSION

Based on the results of the research conducted on speech recognition of the Kambera dialect of the East Sumba language, several critical conclusions and recommendations have been identified. Analysis using the TensorFlow framework with the CNN algorithm showed that the minimum achievable accuracy was 92.3%, while the maximum accuracy reached 98.2%. When this method was combined and optimized with the MFCC approach, there was a significant increase in the average accuracy, being in the range of 98.3% to 99.6%. This improvement suggests that the integration of MFCC can facilitate more effective feature extraction, thereby improving the speech recognition accuracy of the initial model. Furthermore, user testing resulted in a positive result with an accuracy rate of 99.33%, confirming that the system has an excellent voice detection rate and is ready for implementation.

In developing this learning model, there are several obstacles and limitations encountered: the duration of dialect voice recording is limited to two seconds for spectrogram sample distribution, clear and loud dialect voice input are required during data training to produce accurate audio waves, noise has a significant effect on accuracy results, similar vocabulary will affect the existing dataset on the accuracy level, increase in data training process time with increasing number of class samples (>100), adequate computer specifications to support the process. Recommended specifications include a multicore or multithreading server processor with a high clock speed, such as an Intel Xeon or AMD EPYC, at least 16GB of memory, at least 20GB of storage capacity, and a stable internet connection. These recommendations aim to optimize system performance, especially in the face of increased load due to adding test samples.

CONFLICT OF INTEREST

During the implementation of the research study "Teachable Machine: Voice Detection of East Sumba Regional Dialect (Kambera) Using Open-Source Services," the author personally has no conflict of interest with any parties.

AUTHOR CONTRIBUTION

Conceptualization, Edwin Ariesto Umbu Malahina; methodology, Edwin Ariesto Umbu Malahina; formal analysis, formal analysis; testing and system development, Edwin Ariesto Umbu Malahina.

REFERENCES

[1] E.A.U. Malahina, R.P. Hadjon, and F.Y. Bisilisin, "Teachable Machine: Real-Time Attendance of Students Based on Open Source System," *The IJICS (Int. J. Inform., Comput. Sci.)*, Vol. 6, No. 3, pp. 140–146, Nov. 2022, doi: 10.30865/ijics.v6i3.4928.

[2] A.T. Murray and J. Baik, "Opensource Spatial Optimization in GIScience for Strategic Positioning," *Trans. GIS*, Vol. 27, No. 3, pp. 646–662, May 2023, doi: 10.1111/tgis.13033.

[3] J.M.D.S. Dos Santos, C.A.A.P. Abar, and M.V. de Almeida, "Automatic Feedback GeoGebra Tasks – Searching and Opensource and Collaborative Intelligent Interactive Tutor," *Proc. 26th World Multi-Conf. Systemics Cybern., Inform.: WMSCI 2022*, 2022, pp. 77–82, doi: 10.54808/WMSCI2022.03.77.

[4] S. Majumdar and N.D. Pan, "Combining Opensource GIS and Meta-Analysis to Link Rainfall Trend and Human Activity: Case Study on Gumti and Khowai Drainage Systems, Tripura, India," *Spat. Inf. Res.*, Vol. 28, No. 3, pp. 287–298, Jun. 2020, doi: 10.1007/s41324-019-00288-8.

[5] H.D. Trung, N.T. Hung, and N.H. Trung, "Opensource Based IoT Platform and LoRa Communications with Edge Device Calibration for Real-Time Monitoring Systems," in *Advanced Computational Methods for Knowledge Engineering. ICCSAMA 2019. Advances in Intelligent Systems and Computing*, H.A.L. Thi, H.M. Le, T.P. Dinh, and N. Nguyen, Eds., Cham, Switzerland: Springer, 2019, pp. 412–423, doi: 10.1007/978-3-030-38364-0_37.

[6] M. Jebbar, A. Maizate, and R.A. Abdelouahid, "Moroccan's Arabic Speech Training and Deploying Machine Learning Models with Teachable Machine," *Procedia Comput. Sci.*, Vol. 203, pp. 801–806, 2022, doi: 10.1016/j.procs.2022.07.120.

[7] P.Y. Prasad *et al*., "Implementation of Machine Learning Based Google Teachable Machine in Early Childhood Education," *Int. J. Early Child. Special Educ.*, Vol. 14, No. 3, pp. 4132–4138, May 2022, doi: 10.9756/INT-JECSE/V14I3.527.

[8] D. Agustian, P.P.G.P. Pertama, P.N. Crisnapati, and P.D. Novayanti, "Implementation of Machine Learning Using Google's Teachable Machine Based on Android," *2021 3rd Int. Conf. Cybern., Intell. Syst. (ICORIS)*, 2021, pp. 1–7, doi: 10.1109/ICORIS52787.2021.9649528.

[9] Pengelola Web Kemdikbud (2022) "Pelestarian Bahasa Daerah Menjaga Warisan Bangsa," [Online], https://www.kemdikbud.go.id/main/blog/2022/02/pelestarian-bahasa-daerah-menjaga-warisan-bangsa, access date: 13 May 2023.

[10] (2023) "Bahasa Sumba Timur," [Online], https://labbineka.kemdikbud.go.id/bahasa/databahasa/3644a684f98ea8fe223c713b77189a77, access date: 13-May-2023.

[11] R.M.I. Malo, "Preliminary Study of the Dialects of Kambera," *KULTURISTIK: J. Bhs., Budaya*, Vol. 5, No. 2, pp. 1–6, Jul. 2021, doi: 10.22225/kulturistik.5.2.3654.

[12] N.W. Kasni, "Strategy to Combine Clauses in Waijewa Dialect a Sumbanese Language," *e-J. Linguist.*, Vol. 6, No. 2, pp. 93–107, Jul. 2012.

[13] I.G. Budasi, "Bukti-Bukti Leksikal Pembeda Bahasa Wanokaka dan Anakalang di Sumba NTT," *Mabasan*, Vol. 4, No. 1, pp. 24–42, Jan.–Jun. 2010, doi: 10.26499/mab.v4i1.184.

[14] I.W. Simpen, A.M. Mbete, I.M. Suastra, and I.W. Pastika, "Kesantunan Berbahasa pada Penutur Bahasa Kambera di Sumba Timur," *e-J. Linguist.*, Vol. 2, No. 1, pp. 1–15, May 2008.

[15] J.S. Lansing *et al*., "Coevolution of languages and Genes on the Island of Sumba, Eastern Indonesia," *Proc. Nat. Acad. Sci. U. S. A.*, Vol. 104, No. 41, pp. 16022–16026, Oct. 2007, doi: 10.1073/pnas.0704451104.

[16] I. Iswanto, V.J. Arnold, J. Kabnani, and T. Salau, "Kajian Antropolinguistik Bentuk Lingual Umbu dalam Nyanyian Tidur "Ille Le" pada Masyarakat Melolo, Kabupaten Sumba Timur, Nusa Tenggara Timur (Antropholinguistic Study Word "Umbu" in Sleeping Song 'Ille Le' at the Melolo Community, East Sumba, East Nusa Tenggara)," *Jalabahasa*, Vol. 17, No. 2, pp. 179–191, Nov. 2021, doi: 10.36567/jalabahasa.v17i2.768.

[17] M. Hoijtink and A. Planqué-van Hardeveld, "Machine Learning and the Platformization of the Military: A Study of Google's Machine Learning Platform TensorFlow," *Int. Political Sociol.*, Vol. 16, No. 2, pp. 1–19, Jun. 2022, doi: 10.1093/ips/olab036.

[18] Y. Xie, M. He, T. Ma, and W. Tian, "Optimal Distributed Parallel Algorithms for Deep Learning Framework TensorFlow," *Appl. Intell.*, Vol. 52, No. 4, pp. 3880–3900, Mar. 2022, doi: 10.1007/s10489-021-02588-9.

[19] T. He *et al*., "Bag of Tricks for Image Classification with Convolutional Neural Networks," *2019 IEEE/CVF Conf. Comput. Vis., Pattern Recognit. (CVPR)*, 2019, pp. 558–567, doi: 10.1109/CVPR.2019.00065.

[20] G. Jiang, H. He, J. Yan, and P. Xie, "Multiscale Convolutional Neural Networks for Fault Diagnosis of Wind Turbine Gearbox," *IEEE Trans. Ind. Electron.*, Vol. 66, No. 4, pp. 3196–3207, Apr. 2019, doi: 10.1109/TIE.2018.2844805.

[21] (2022) "What is TensorFlow.js?," [Online], https://codelabs.developers.google.com/TensorFlowjs-transfer-learning-teachable-machine#1, access date: 3-Oct-2022.

[22] Z. Tariq, S.K. Shah, and Y. Lee, "Feature-Based Fusion Using CNN for Lung and Heart Sound Classification," *Sens.*, Vol. 22, No. 4, pp. 1–28, Feb. 2022, doi: 10.3390/s22041521.

[23] U.A. Vishniakou and B.H. Shaya, "Voice Detection Using Convolutional Neural Network," *Doklady BGUIR*, Vol. 21, No. 2, pp. 114–120, Apr. 2023, doi: 10.35596/1729-7648-2023-21-2-114-120.

[24] A.Y. Kim *et al*., "Automatic Depression Detection Using Smartphone-Based Text-Dependent Speech Signals: Deep Convolutional Neural Network Approach," J. *Med. Internet Res.*, Vol. 25, pp. 1–17, Jan. 2023, doi: 10.2196/34474.

[25] P.R. Prakash *et al*., "A Novel Convolutional Neural Network with Gated Recurrent Unit for Automated Speech Emotion Recognition and Classification," *J. Control, Decis.*, Vol. 10, No. 1, pp. 54–63, Jan. 2023, doi: 10.1080/23307706.2022.2085198.

[26] Y. Yu, C. Peng, Q. Tang, and X. Wang, "Monaural Music Source Separation Using Deep Convolutional Neural Network Embedded with Feature Extraction Module," *2022 Asia Conf. Algorithms, Comput., Mach. Learn. (CACML)*, 2022, pp. 546–551, doi: 10.1109/CACML55074.2022.00098.

[27] P.H. Chandankhede, A.S. Titarmare, and S. Chauhvan, "Voice Recognition Based Security System Using Convolutional Neural Network," *2021 Int. Conf. Comput. Commun., Intell. Syst. (ICCCIS)*, 2021, pp. 738–743, doi: 10.1109/ICCCIS51004.2021.9397151.

[28] N. Li *et al*., "Robust Voice Activity Detection Using a Masked Auditory Encoder Based Convolutional Neural Network," *ICASSP 2021 - 2021 IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2021, pp. 6828–6832, doi: 10.1109/ICASSP39728.2021.9415045.

[29] M. Soundarya, P.R. Karthikeyan, and G. Thangarasu, "Automatic Speech Recognition Trained with Convolutional Neural Network and Predicted with Recurrent Neural Network," *2023 9th Int. Conf. Elect. Energy Syst. (ICEES)*, pp. 41–45, doi: 10.1109/ICEES57979.2023.10110224.

[30] M.M. Kamruzzaman, "Arabic Sign Language Recognition and Generating Arabic Speech Using Convolutional Neural Network," *Wirel. Commun., Mob. Comput.*, Vol. 2020, pp. 1–9, May 2020, doi: 10.1155/2020/3685614.

[31] M. Hireš *et al*., "Convolutional Neural Network Ensemble for Parkinson's Disease Detection from Voice Recordings," *Comput. Biol.*, Med., Vol. 141, pp. 1–9, Feb. 2022, doi: 10.1016/j.compbiomed.2021.105021.

[32] K.K. Lella and A. Pja, "Automatic Diagnosis of COVID-19 Disease Using Deep Convolutional Neural Network with Multi-Feature Channel from Respiratory Sound Data: Cough, Voice, and Breath," *Alex. Eng. J.*, Vol. 61, No. 2, pp. 1319–1334, Feb. 2022, doi: 10.1016/j.aej.2021.06.024.

[33] I. Kwon *et al*., "Diagnosis of Early Glottic Cancer Using Laryngeal Image and Voice Based on Ensemble Learning of Convolutional Neural Network Classifiers," *J. Voice*, to be published.

[34] M. Hireš, M. Gazda, L. Vavrek, and P. Drotár, "Voice-Specific Augmentations for Parkinson's Disease Detection Using Deep Convolutional Neural Network," *2022 IEEE 20th Jubil. World Symp. Appl. Mach. Intell., Inform. (SAMI)*, 2022, pp. 000213–000218, doi: 10.1109/SAMI54271.2022.9780856.

[35] T.T. Leonid and R. Jayaparvathy, "Classification of Elephant Sounds Using Parallel Convolutional Neural Network," *Intell. Automat., Soft Comput.*, Vol. 32, No. 3, pp. 1415–1426, Jun. 2022, doi: 10.32604/iasc.2022.021939.

[36] S.D.H. Permana *et al*, "Classification of Bird Sounds as an Early Warning Method of Forest Fires Using Convolutional Neural Network (CNN) Algorithm," *J. King Saud Univ. – Comput., Inf. Sci.*, Vol. 34, No. 7, pp. 4345–4357, Jul. 2022, doi: 10.1016/j.jksuci.2021.04.013.

[37] W.S.E. Putra, "Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101," *J. Tek. ITS*, Vol. 5, No. 1, pp. A65–A69, Mar. 2016, doi: 10.12962/j23373539.v5i1.15696.

[38] M. Carney *et al*., "Teachable Machine: Approachable Web-Based Tool for Exploring Machine Learning Classification," *Ext. Abstr. 2020 CHI Conf. Hum. Factors Comput. Syst.*, 2020, pp. 1–8, doi: 10.1145/3334480.3382839.