Human Age Estimation Through Audio Utilizing MFCC and RNN

Ken Ken[1], Osfredo Quinn[2], Irpan Adiputra Pardosi[3], Wenripin Chandra[4]
[1,2,3]Program Studi Teknik Informatika, Universitas Mikroskil, Medan, Indonesia
[4]Program Studi Teknik Informatika, Universitas Mikroskil, Medan, Indonesia; Program Studi Informatika, Universitas Pelita Harapan, Medan, Indonesia
[1]181111048@students.mikroskil.ac.id, [2]181110166@students.mikroskil.ac.id, [3]irpan@mikroskil.ac.id, [4]wripin@gmail.com

Submitted : July 6, 2023 | **Accepted** : July 26, 2023 | **Published** : July 30, 2023

Abstract: Age, one of human main attributes, is an important factor to improve communication experiences. In order to enrich communication experiences, one has to know the opposite's age to adjust the way of communication. To utilize age effectively, age estimation must be as accurate as possible. Age estimation has been used in several applications to improve user experience. Therefore, an approach is needed to estimate the user age, one of which is through audio. In this study, Mel Frequency Cepstral Coefficients (MFCC) and Recurrent Neural Network (RNN) will be used to estimate age through audio. MFCC is used to get features from audio data, while RNN is used to estimate age. Dataset used here was taken from corpus of user speech data on the Common Voice website. This study shows that RNN model Long Short-Term Memory (LSTM) is superior in estimating human age via audio with highest accuracy score 0.7087 while the similar score of SimpleRNN is 0.5647.

Keywords: Classification, Age Estimation, Audio, MFCC, RNN

INTRODUCTION

Age is one of human's main attributes. In communication, age is an important factor used in adjusting to the opposite (Mahmoodi et al., 2011). By aware to the opposite's age, a party can better understand the opposite and improve communication experience by having suitability in diction, context as well as communication way (Alwi, Adikara, & Indriati, 2020).

As well as human, various applications also need human age to adapt to their users. Those applications such as application with age-based access system, security access control application, age adaptive targeted marketing application, electronic customer, relationship management, entertainment recommender system, biometric, *precision advertising*, *intelligent surveillance*, Internet access control, social media analysis, health, psychologic, as well as human computer interaction (HCI) (Zaghbani, Boujneh, & Bouhlel, 2018). The large number of users makes it difficult to adapt the application to the user so that the need to know the user's age automatically increases rapidly. Most applications find age by using estimates because getting exact age is not easy. Therefore, some applications do more age estimation based on age ranges because it is easier to see and manage differences in thinking, behavior, and habits in age groups (Sánchez-Hevia, Gil-Pita, Utrilla-Manso, & Rosa-Zurera, 2019). One mean to do age estimation is voice recognition through audio signals where audio has acoustic characteristics that distinguish the age of speaker such as pitch, loudness, pressure, vocal vibrations, breath, speed of speech, and pauses between sounds (Spiegl et al., 2009).

Mel-Frequency Ceptrum Coefficients (MFCC) is an effective and robust characteristic extraction method for audio signals. (Singh & Rani, 2014). MFCC is able to provide higher level of accuracy compared to other feature extraction methods such as linear prediction cepstral coefficient (LPCC) and zero crossing rate (ZCR) in performing speaker recognition (Chauhan, Isshiki, & Li, 2019). Research on MFCC and machine learning in estimating age through audio has been carried out such as combining the Support Vector Machine (SVM) and MFCC methods with error rate of 5.89%.(Mahmoodi et al., 2011), and the combination of the K-Nearest Neighbor (K-NN) and MFCC methods produces 44% accuracy (Abdulsatar, Davydov, Yushkova, Glinushkin, & Rud, 2019). However, methods such as SVM and K-NN still have drawbacks because of longer training time and requires manual feature engineering resulting in increased time for preprocessing and data mining. To overcome this problem, deep learning method is used to perform feature engineering layer by layer automatically. This can provide better results compared to manual feature engineering (Wu, Han, Song, & Wang, 2019).

*name of corresponding author

One of the deep learning methods that widely used is the recurrent neural network (RNN) method because of its ability to recognize information with many time variants (Tridarma & Endah, 2020). Sequential data, such as audio signals obtained from human voices, can be processed properly using the RNN method (Wu et al., 2019). SimpleRNN is one of the RNN methods with basic structure that can process sequential data and has a memory that functions to store information on data that has been processed so that this information can be channeled to the next process to produce more optimal results (Apaydin et al., 2020). In previous research, the SimpleRNN method also showed results in solving grouping problems well (Dixon, 2018). Additionally, one of the other frequently used RNN methods is the Long Short-Term Memory (LSTM) where LSTM can carry out learning process for long-term dependencies because it has a memory block that can determine which value will be selected as output relevant to the input provided (Wiranda & Sadikin, 2019). Previous research on LSTM also shows that classification of audio signals provides very good accuracy (Raza et al., 2019). In this research we built an application implementing MFCC and RNN and then test the performance of the models built to find which one, SimpleRNN or LSTM, is superior in estimating human age via audio.

LITERATURE REVIEW

2.1. Mel Frequency Cepstral Coefficients

Mel Frequency Cepstral Coefficients (MFCC) is a feature extraction method by representing audio signals based on the peripheral human hearing system and mimicking the way of human auditory system responds to low frequency sounds linearly and high frequency sounds logarithmically. The process sequence to obtain MFCC can be seen in Figure 1.

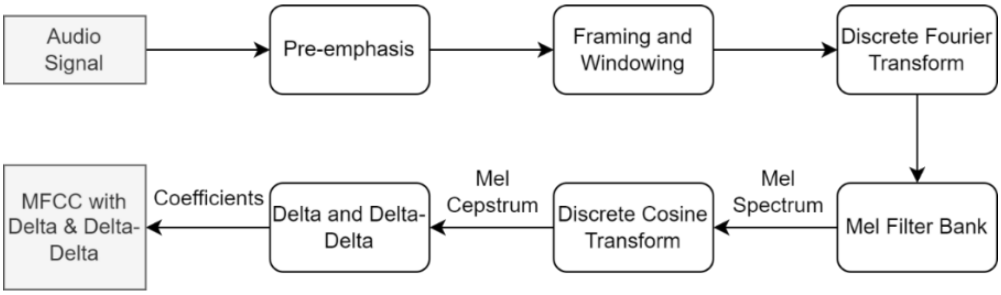

Fig. 1 Block Diagram for Obtaining Mel Frequency Cepstrum Coefficients

MFCC processes are:

1. Pre-emphasis
   Pre-emphasis is carried out to amplify high frequencies in the audio signal so that the energy in signal increases (Martinez, Perez, Escamilla, & Suzuki, 2012). Pre-emphasis can be expressed by the following equation:
   $$y_n = x_n - ax_{n-1} \quad , a \approx 0.95 - 0.97 \tag{1}$$

2. Frame Blocking and Windowing
   This process aims to divide the signal into several short duration blocks or frames, as well as to avoid non-stationary or changeable signals. Then, the frame collections are windowed to reduce signal discontinuity in the frame blocking process or loss of information between frames. The method that is often used is the hamming window which is expressed in following equation:
   $$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \tag{2}$$

3. Discrete Fourier Transform
   Discrete Fourier Transform (DFT) is process to convert audio signal from time domain to frequency domain. The DFT process can be defined by:
   $$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{\frac{-i2\pi kn}{N}} \tag{3}$$
   After obtaining the results, the energy spectral calculation (periodogram) is carried out with following equation:
   $$P_k = \frac{1}{N}|X_k|^2 \tag{4}$$

4. Mel Filter Bank

*name of corresponding author

The distance between frequencies obtained from results of Fourier transform is very wide, making it difficult to map on linear scale (Martinez et al., 2012). Therefore, these frequencies must be mapped onto the mel scale in order to determine amount of energy present with the help of a triangular filter. With mel scale, mapping results will have linear distance when the frequency is below 1 kHz, and logarithmic when it is above 1 kHz. In each filter bank, the process of calculating lowest and highest frequencies in mel frequency is carried out with the following equation:

$$m = 2595\ \log_{10}\left(1 + \frac{f}{700}\right) \tag{5}$$

Each filter bank has result eith value of 1 at the midpoint of the frequency and decreases linearly towards 0 until the frequency reaches the midpoint between two adjacent frequencies whose value is 0. Mel Filter Bank can be determined by following equation:

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

The results of the Fourier transform will be filtered using formed filter bank formed. The filtering process can be done using the following equation. The result of the Fourier transform will be followed by filtering using a formed filter bank. The filtering process can be carried out using the following equation..

$$D'_m = \ln\left(\sum_{k=0}^{N-1} |D_k| \cdot H_m(k)\right) \tag{7}$$

5. Discrete Cosine Transform

The next process in MFCC feature extraction is performing Discrete Cosine Transform (DCT). Discrete Cosine Transform is used to get the correlation value of the mel spectrum, namely cepstrum. Cepstrum is the DCT result obtained from mel frequency in the time domain. Discrete Cosine Transform is expressed through the following equation:

$$c_{\mathrm{n}} = 2 * \sum_{k=0}^{K-1} D'_k \cdot \cos\left[n(k - 0.5)\frac{\pi}{K}\right] \tag{8}$$

Where:

c_{n} = DCT result at nth index
D'_k= results of kth spectrum
K = number of signals to convert

The results of cepstrum are then taken by several coefficients from the second coefficient as the cepstrum MFCC coefficient.

6. Delta and Deltas

The process of obtaining delta and deltas is carried out after obtaining the cepstrum coefficient. Delta (differential) and deltas (acceleration) are coefficients that function to get to recognize audio better such as dynamics of the spectrum strength or changes that occur in MFCC from time to time, as well as providing information about changes and language styles to the speaker (Oliveira et. all, 2018). Calculation of delta and deltas can be formulated by the following equation:

$$d_t = \frac{\sum_{n=1}^{N} n(c_{t+n} - c_{t-n})}{2 \cdot \sum_{n=1}^{N} n^2} \tag{9}$$

2.2 SimpleRNN

SimpleRNN is a simple RNN using process that resembles the previous process on each neuron sequentially and refers to previously stored information. SimpleRNN works by changing the hidden state h, which is a vector with one dimension at time-t. The next hidden state h_t is calculated using the previous hidden state h_{t-1} with next input x_t. Then, the next output y_t is calculated using h_t. SimpleRNN can be described by the following equation:

$$\begin{gathered} h_t = tanh(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \\ y_t = W_{hy}h_t + b_y \end{gathered} \tag{10}$$

2.3. Long Short-Term Memory

Long Short-Term Memory or LSTM is one of RNN architectures with memory cells so that the network can solve problems that depend on it for a long time. LSTM architecture can be described as follows:

1. Forget Gates
   Forget gates are gates for forgetting information that is irrelevant or no longer needed by the LSTM. Forget gates can be expressed by the following equation:
   $$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \quad (11)$$
2. Input Gates
   Input gates are gates to enter useful information to support data accuracy. Input gates can be expressed by the following equation:
   $$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (12)$$
3. Cells
   Cells work as information storage at time t. The equation for cells can be expressed by the following equation:
   $$c_t = f_t c_{t-1} + i_t tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (13)$$
4. Output Gates
   Output gates work in determining the value of memory cell at a later time. Output gates can be written into the following equation:
   $$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \quad (14)$$

The results of the output gates will then be calculated to get value to be passed on to the next cell. The calculation of these results can be formulated as follows:

$$h_t = o_t\, tanh(c_t) \quad (15)$$

2.4. Confusion Matrix

One of the ways to evaluate performance of a multi-class classification such as age estimation is confusion matrix. The confusion matrix is method of evaluating performance of a classifier based on testing data. Based on confusion matrix, there are several things obtained, namely:

1. Accuracy
   Accuracy is comparison between the number of main diagonals of the confusion matrix and total number of predictions. Calculation of accuracy can be seen in the equation:
   $$accuracy = \frac{\sum_i^m C_{ii}}{\sum_i^m \sum_j^m C_{ij}} \quad (16)$$
2. Precision
   Precision is ratio of the number of correct predictions by the system to the number of incorrect predictions in predicting the correct results. Precision can be calculated by the equation:
   $$precision_i = \frac{C_{ii}}{\sum_{j=1}^m C_{ji}} \quad (17)$$
3. Recall
   Recall is ratio of the number of samples with correct predictions to the total samples with predictions of correct results and predictions of incorrect results. Recall can be formulated by:
   $$recall_i = \frac{C_{ii}}{\sum_{j=1}^m C_{ij}} \quad (18)$$
4. F_1-score
   F_1-score is measurement of the harmonic mean system performance of precision and recall. F_1-score can be calculated by the following equation:
   $$F_1 - score_i = 2 \times \frac{precision_i \times recall_i}{precision_i + recall_i} \quad (19)$$

METHOD

3.1. Research Object

Object used in this research is the Common Voice dataset which is a corpus of user speech data on the Common Voice website created by Mozilla and obtained via Kaggle. The dataset used consists of 29,555 training datasets and 618 testing datasets which contain male and female voices and have United States English voice accent. Each audio in dataset are in MPEG-1 Audio Layer 3 (MP3) format and contain information about age range of the speakers which are teens, twenties, thirties, forties, fifties, sixties, seventies and eighties. Figure 2 shows statistics of the dataset.

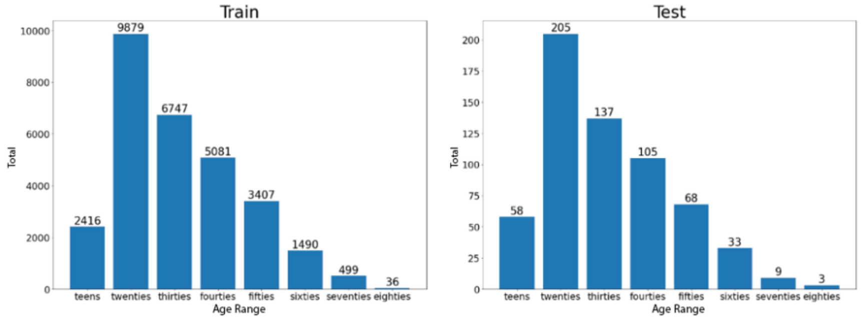

Fig. 2 Distribution of age range and the number of datasets in each class

3.2. Research Stages

Data preparation is carried out to tidy up data, improve data quality, speed up data processing, and reduce errors when processing data (Hameed & Naumann, 2020). In carrying out data preparation, each audio data will be converted into a single channel or monoaural (abbreviated as mono). The process is continued by resampling each audio data with sample rate of 22050 to reduce memory consumption and noise or excessive low spectrum. Then, for each audio, silent removal process is carried out on audio parts that have a silent duration of more than 0.1 seconds and decibel size below -50 dB.

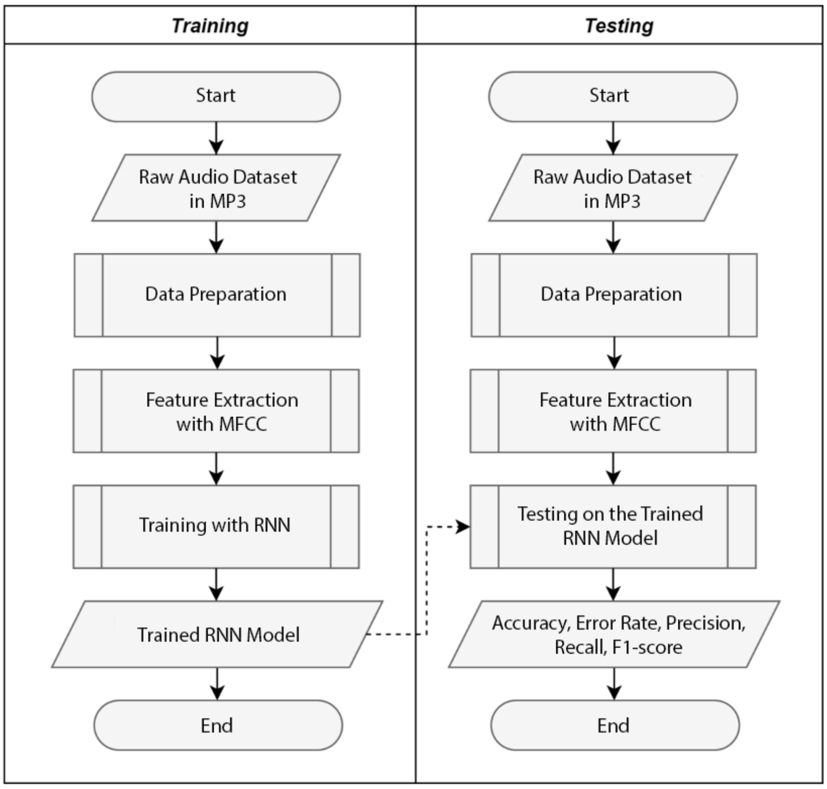

Fig. 3 Training and testing flowchart

To do age estimation through audio, training and testing process needs to be done on the model before it can be used. The implementation scheme based on flowchart in Figure 3 can be seen in Figure 4.

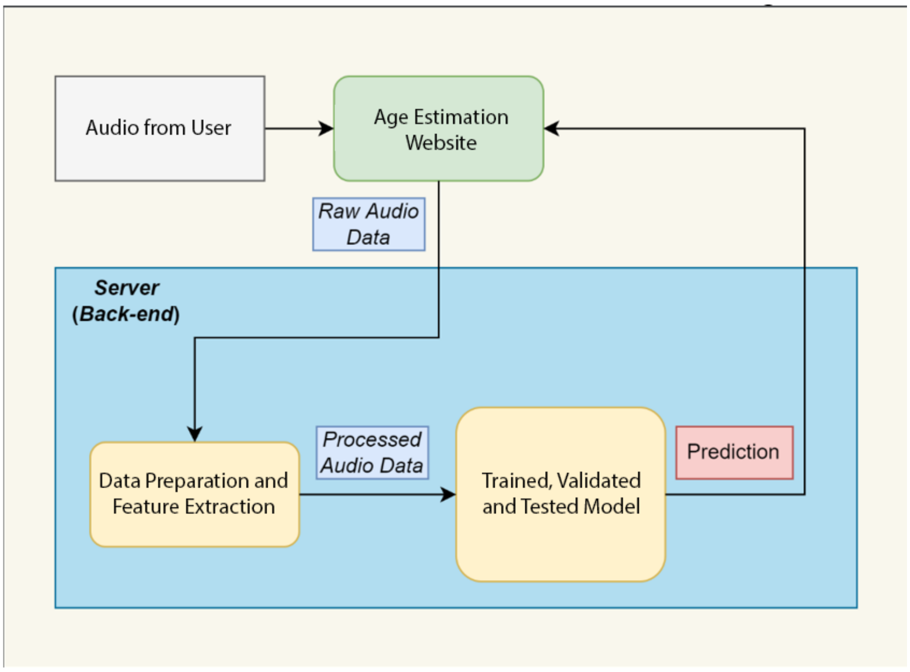

Fig. 4 Implementation scheme of age estimation via audio

Based on the scheme in Figure 4, the neural network model that has been formed will be stored in a file with the Hierarchical Data Format version 5 (HDF5) extension so it can be reused without having to go through the training and validation process again. HDF5 is a file format that supports large, complex, and heterogeneous data. Using the stored data, the model can make predictions by making calls to the back-end via created website. This website uses HTML and CSS as well as Javascript as the main display. As back-end, we use Flask framework. Flask is a micro web framework written in Python. Flask is used to enable calls to neural network results via REST API in website. By going through Flask framework, audio data entered by user will be sent to the back-end in binary form, then the back-end system will carry out data preparation and feature extraction with MFCC. The processed data will then be predicted by recurrent neural network. The prediction results will be sent back via REST API to the website to be displayed to the user in form of estimated age based on the entered audio data.

RESULT

This testing was conducted using Python and TensorFlow library. Experiments were carried out in system with CPU Intel Core i7-1165G7 @ 2.80 GHz and RAM DDR4 16 GB 3200 MHz. Performance of the methods is evaluated using accuracy, precision, recall, and f1-score. We conduct experiments on model with MFCC (Mel Frequency Cepstral Coefficients) and RNN (Recurrent Neural Network) using 29.555 training datasets and 618 testing datasets. In the experiment, MFCC feature extraction are used with 13 cepstral coefficients, 13 delta, and 13 deltas as input, batch size of 256, and ratio 90:10 for training validation. The two RNN models in this experiment are SimpleRNN and LSTM.

The training process on each model is carried out on two hyperparameters namely learning rate and epoch to determine the model with best accuracy. The tests are carried out using architectural model that has an input layer, output layer and 5 hidden layers each with 128 neurons, one with 2 dense layers with 64 units and dropout value of 0.1, while the other has a dense layer with 32 units and dropout value of 0.1.

SimpleRNN Model Testing

Based on the testing architecture, SimpleRNN model gives following test results:

1. Learning rate
   The training scenario carried out was to compare nine learning rates, namely 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, and 1; and number of epochs used is 10.

*name of corresponding author

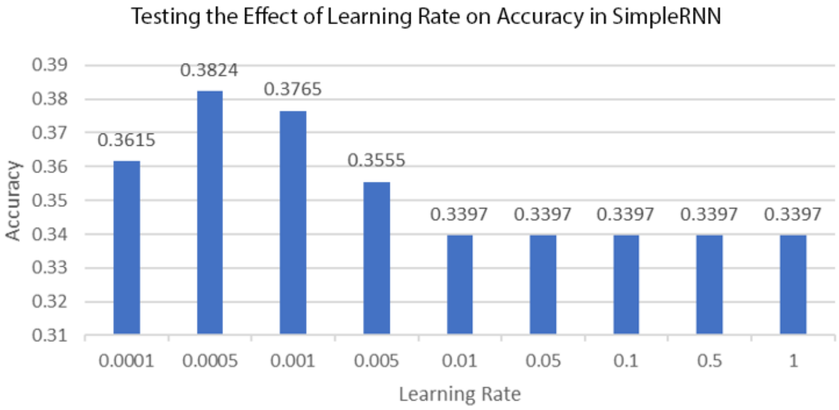

Fig. 5 Results of testing the effect of learning rate on accuracy in SimpleRNN

Based on learning rate test results, changes to the learning rate value affect the validation accuracy results. When the learning rate is above 0.0005, the accuracy results are lower. This proves that in SimpleRNN model, the learning process is too fast resulting in model being unable to estimate age optimally [2]. The results of the learning rate test show that the learning rate 0.0005 in SimpleRNN model with the highest accuracy value of 0.3824.

2. Epoch

Epoch were tested up to the 1000th epoch with learning rate hyperparameter 0.0005 which has the highest accuracy when tested. The results of test can be seen in Figure 6.

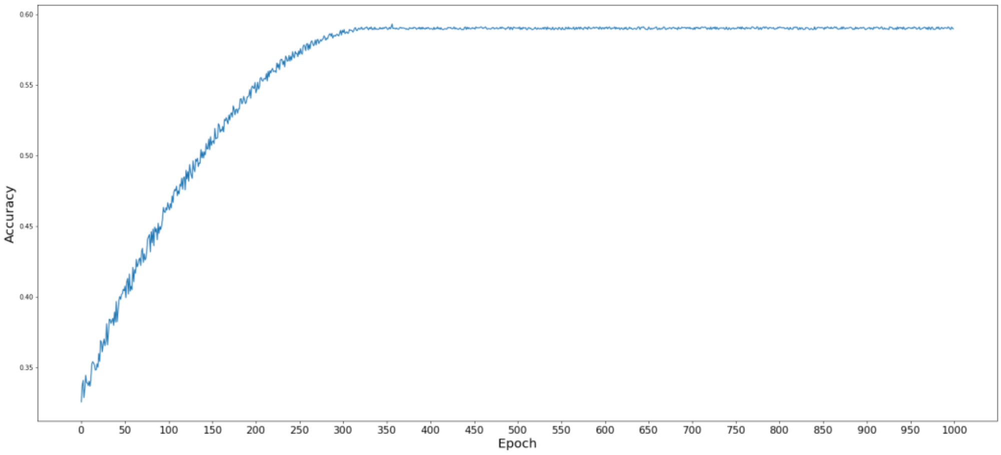

Fig. 6 Results of testing the effect epoch on accuracy in SimpleRNN

Results of testing effect of epoch on accuracy in SimpleRNN shown in Figure 6 indicate validation accuracy value obtained is fluctuating where the next epoch can increase or decrease accuracy value. Even so, when the epoch is higher, the accuracy becomes more stable until it reaches convergent state which is state where the learning process in the model no longer increases. The convergent state occurs after the epoch is above 400 where change in accuracy is very small between 0.001 to 0.002. Based on the epoch test, it was found that SimpleRNN model got the highest accuracy value of 0.5930 at epoch 356.

From training tests conducted on SimpleRNN model, it was found that SimpleRNN with learning rate 0.0005 and epoch 356 got the best accuracy value. Furthermore, the best training model will be tested to determine performance of this model. From testing carried out on 618 audio data, results of confusion matrix can be seen in Figure 7.

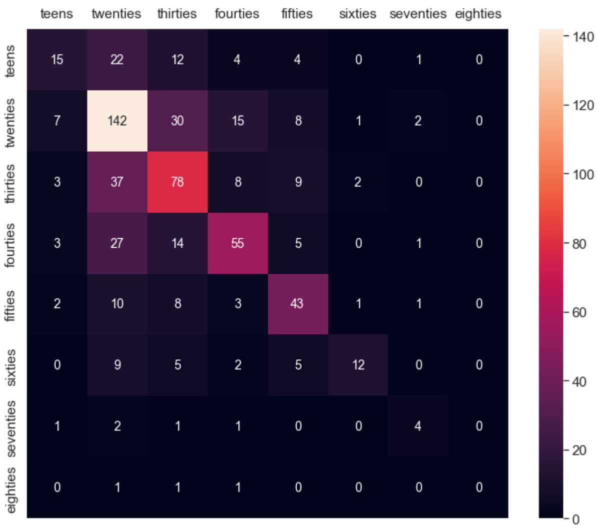

Fig. 7 Confusion matrix when tested with the best SimpleRNN model

Figure 7 shows the accuracy value is (15+142+78+55+43+12+4+0)/618 = 0.5647. Precision, recall, and f1-score for each age range shown in Table 1.

Table 1. Precision, recall, and f1-score for each age range using the best SimpleRNN model

Age Range	Precision	Recall	F1-score
Teens	0.4839	0.2586	0.3371
Twenties	0.568	0.6927	0.6242
Thirties	0.5235	0.5693	0.5455
Fourties	0.618	0.5238	0.567
Fifties	0.5811	0.6324	0.6056
Sixties	0.75	0.3636	0.4898
Seventies	0.4444	0.4444	0.4444
Eighties	0	0	0

The test shows that the model is unable to estimate age range of eighties (value 0 in precision, recall, and f1-score) due to lack of data for training as shown in Figure 2. That age range only has 36 datasets out of total of 29555 datasets. Test results show that number of datasets has positive impact on performance where the more datasets in age range, the better results of precision, recall and f1-score can be obtained. Testing on SimpleRNN model produces accuracy of 0.5647, average precision of 0.4961, average recall of 0.4356, and average f1-score of 0.4517.

LSTM Model Testing

Based on the testing architecture, LSTM model gives following test results:

1. Learning rate
   The training scenario carried out was to compare nine learning rates, namely 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, and 1; and number of epochs used is 10.

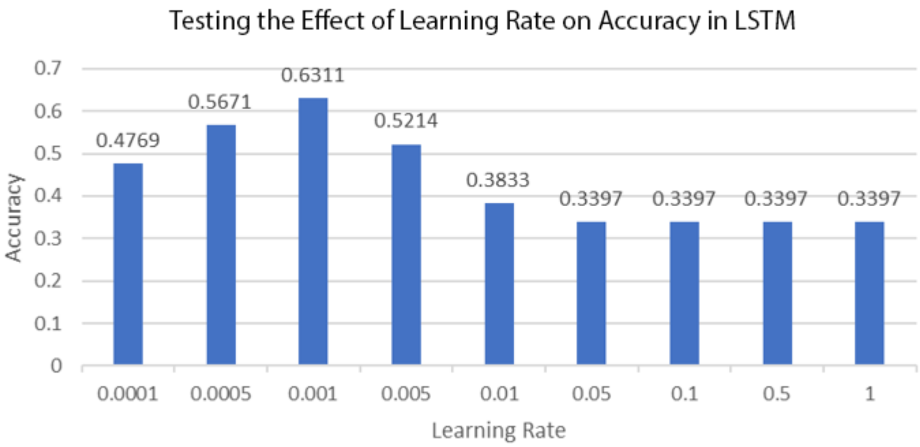

Fig. 8 Results of testing the effect of learning rate on accuracy in LSTM

2. Epoch

Epoch were tested up to the 1000th epoch with learning rate hyperparameter 0.001 which has the highest accuracy when tested. The results of test can be seen in Figure 9.

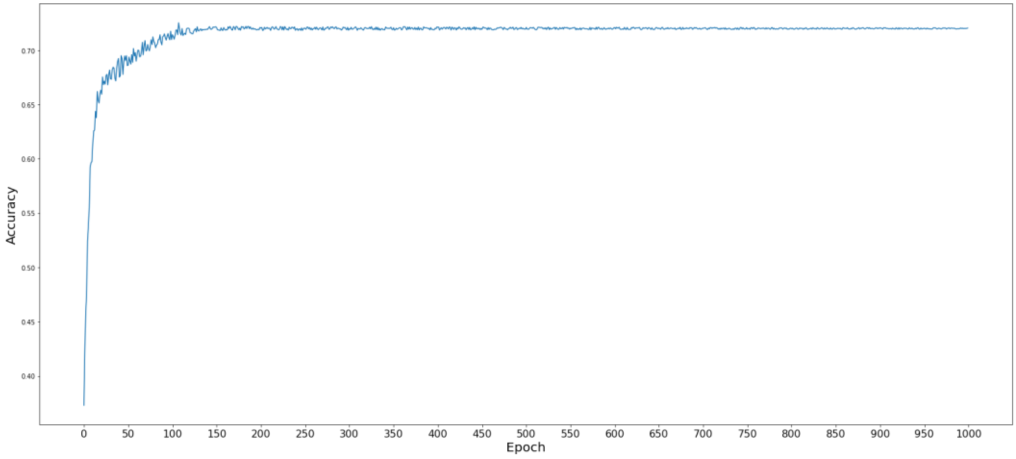

Fig. 9 Results of testing the effect epoch on accuracy in LSTM

Results of testing effect of epoch on accuracy in LSTM shown in Figure 9 indicate accuracy value obtained is fluctuating initially. Even so, when the epoch is higher, the accuracy becomes more stable until it reaches convergent state above epoch 200 where change in accuracy is very small between 0.001 to 0.002. In the epoch test, it was found that LSTM model got the highest accuracy value 0.7252 at epoch 107.

From training tests conducted on LSTM model, it was found that LSTM with learning rate of 0.001 and epoch 107 got the best accuracy value. Furthermore, the best training model will be tested to determine performance of this model. From testing carried out on 618 audio data, results of confusion matrix can be seen in Figure 10.

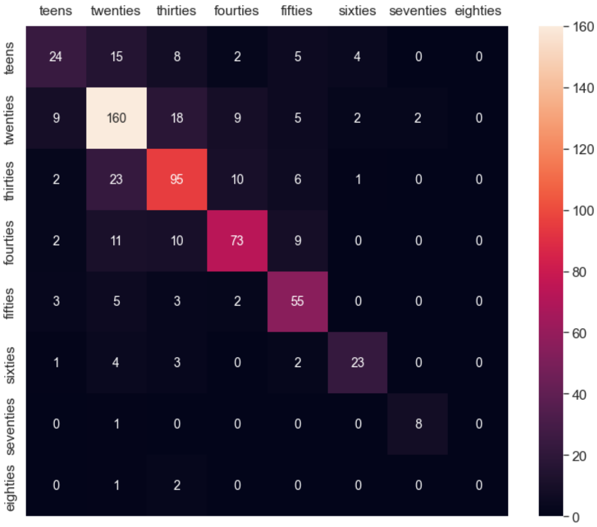

Fig. 10 Confusion matrix when tested with the best LTSM model

Figure 10 shows the accuracy value is (24+160+95+73+55+23+8+0)/618 = 0.7087. Precision, recall, and f1-score for each age range shown in Table 2.

Table 2. Precision, recall, and f1-score for each age range using the best LTSM model

Age Range	Precision	Recall	F1-score
Teens	0.5854	0.4138	0.4848
Twenties	0.7273	0.7805	0.7529
Thirties	0.6835	0.6934	0.6884
Fourties	0.7604	0.6952	0.7264
Fifties	0.6707	0.8088	0.7333
Sixties	0.7667	0.697	0.7302
Seventies	0.8	0.8889	0.8421
Eighties	0	0	0

The test shows that this model also unable to estimate age range of eighties (value 0 in precision, recall, and f1-score) as in SimpleRNN due to lack of data for training as shown in Figure 2. Testing on model produces accuracy of 0.7087, average precision of 0.6242, average recall of 0.6222, and average f1-score of 0.6198.

DISCUSSIONS

From the test, it was found that LSTM model had better results compared to SimpleRNN. Meanwhile, these two models have difficulties in estimating age range of eighties due to the imbalanced dataset problem, so it is advisable to add datasets or use more balanced dataset. Subsequent tests can also consider to add features such as pitch, chroma features and energy to better understand the characteristics of each sound in the audio so that audio recognition process can be improved.

CONCLUSION

In this research, we learn that MFCC and RNN can be used as approach to estimate user age through audio, with LTSM superior than SimpleRNN. SimpleRNN model has the best accuracy 0.5647, average precision 0.4961, average recall 0.4356, and average f1-score 0.4517 while LSTM model has the best accuracy value of 0.7087, average precision 0.6242, average recall 0.6222, and average f1-score 0.6198. In learning rate test, it was found that learning rate value with the highest accuracy in SimpleRNN is 0.0005 while in LSTM is 0.001. This test also shows that the higher learning rate produces lower accuracy because learning process in the model is too fast. Through epoch testing, it was found that these two RNN models at some point reached convergence, with highest accuracy in SimpleRNN reached on epoch 356, and 107 in LSTM.

REFERENCES

Abdulsatar, A. A., Davydov, V. V., Yushkova, V. V., Glinushkin, A. P., & Rud, V. Y. (2019). Age and Gender Recognition From Speech Signals. *Journal of Physics: Conference Series*, *1410*(1), 0–7.

https://doi.org/10.1088/1742-6596/1410/1/012073

Alwi, A. A., Adikara, P. P., & Indriati. (2020). Pengenalan Jenis Kelamin dan Rentang Umur berdasarkan Suara menggunakan Metode Backpropagation Neural Network. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *4*(7), 2083–2093.

Apaydin, H., Feizi, H., Sattari, M. T., Colak, M. S., Shamshirband, S., & Chau, K. W. (2020). Comparative analysis of recurrent neural network architectures for reservoir inflow forecasting. *Water (Switzerland)*, *12*(5), 1–18. https://doi.org/10.3390/w12051500

Chauhan, N., Isshiki, T., & Li, D. (2019). Speaker Recognition Using LPC, MFCC, ZCR Features With ANN and SVM Classifier for Large Input Database. *2019 IEEE 4th International Conference on Computer and Communication Systems, ICCCS 2019*, 130–133. https://doi.org/10.1109/CCOMS.2019.8821751

Dixon, M. F. (2018). Sequence Classification of the Limit Order Book Using Recurrent Neural Networks. *SSRN Electronic Journal*, 1–20. https://doi.org/10.2139/ssrn.3002814

Hameed, M., & Naumann, F. (2020). Data Preparation: A Survey of Commercial Tools. *ACM SIGMOD Record*, *49*(3), 18–29. https://doi.org/10.1145/3444831.3444835

Mahmoodi, D., Marvi, H., Taghizadeh, M., Soleimani, A., Razzazi, F., & Mahmoodi, M. (2011). Age estimation based on speech features and support vector machine. *2011 3rd Computer Science and Electronic Engineering Conference, CEEC'11*, (May 2014), 60–64. https://doi.org/10.1109/CEEC.2011.5995826

Martinez, J., Perez, H., Escamilla, E., & Suzuki, M. M. (2012). Speaker recognition using Mel frequency Cepstral Coefficients (MFCC) and Vector quantization (VQ) techniques. In *CONIELECOMP 2012, 22nd International Conference on Electrical Communications and Computers* (pp. 248–251). IEEE. https://doi.org/10.1109/CONIELECOMP.2012.6189918

Raza, A., Mehmood, A., Ullah, S., Ahmad, M., Choi, G. S., & On, B. W. (2019). Heartbeat Sound Signal Classification Using Deep Learning. *Sensors (Switzerland)*, *19*(21), 1–15. https://doi.org/10.3390/s19214819

Sánchez-Hevia, H. A., Gil-Pita, R., Utrilla-Manso, M., & Rosa-Zurera, M. (2019). Convolutional-recurrent Neural Network for Age and Gender Prediction From Speech. *2019 Signal Processing Symposium, SPSympo 2019*, 242–245. https://doi.org/https://doi.org/10.1109/SPS.2019.8881961

Singh, P. P., & Rani, P. (2014). An Approach to Extract Feature Using MFCC. *IOSR Journal of Engineering*, *4*(8), 21–25. https://doi.org/10.2307/j.ctt46nrzt.12

Spiegl, W., Stemmer, G., Lasarcyk, E., Kolhatkar, V., Cassidy, A., Potard, B., … Nöth, E. (2009). Analyzing Features for Automatic Age Estimation on Cross-sectional Data. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2923–2926. https://doi.org/10.21437/interspeech.2009-740

Tridarma, P., & Endah, S. N. (2020). Pengenalan Ucapan Bahasa Indonesia Menggunakan MFCC dan Recurrent Neural Network. *Jurnal Masyarakat Informatika*, *11*(2), 36–44.

Wiranda, L., & Sadikin, M. (2019). Penerapan Long Short Term Memory Pada Data Time Series Untuk Memprediksi Penjualan Produk Pt. Metiska Farma. *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, *8*(3), 184–196.

Wu, W., Han, F., Song, G., & Wang, Z. (2019). Music Genre Classification Using Independent Recurrent Neural Network. *Proceedings 2018 Chinese Automation Congress, CAC 2018*, 192–195. https://doi.org/10.1109/CAC.2018.8623623

Zaghbani, S., Boujneh, N., & Bouhlel, M. S. (2018). Age Estimation Using Deep Learning. *Computers and Electrical Engineering*, *68*(October 2017), 337–347. https://doi.org/10.1016/j.compeleceng.2018.04.012