# Spoken language identification on 4 Indonesian local languages using deep learning

**Panji Wijonarko, Amalia Zahra**
Department of Computer Science, Bina Nusantara University, Jakarta, Indonesia

**Article Info**

***Article history:***

Received May 30, 2022
Revised Aug 18, 2022
Accepted Aug 30, 2022

***Keywords:***

Artificial neural network
Convolutional neural network
Deep learning
Long-term short memory
Spoken language identification

## ABSTRACT

Language identification is at the forefront of assistance in many applications, including multilingual speech systems, spoken language translation, multilingual speech recognition, and human-machine interaction via voice. The identification of indonesian local languages using spoken language identification technology has enormous potential to advance tourism potential and digital content in Indonesia. The goal of this study is to identify four Indonesian local languages: Javanese, Sundanese, Minangkabau, and Buginese, utilizing deep learning classification techniques such as artificial neural network (ANN), convolutional neural network (CNN), and long-term short memory (LSTM). The selected extraction feature for audio data extraction employs mel-frequency cepstral coefficient (MFCC). The results showed that the LSTM model had the highest accuracy for each speech duration (3 s, 10 s, and 30 s), followed by the CNN and ANN models.

***Corresponding Author:***

Panji Wijonarko
Department of Computer Science, Bina Nusantara University
Jakarta, 11480 Indonesia
Email: panji.wijonarko@binus.ac.id

## 1. INTRODUCTION

As an archipelagic country, Indonesia is made up of many ethnic groups. Language is one of Indonesia's cultural treasures. According to language agency data, Indonesia constains around 718 languages ranging from Sabang to Merauke [1]. The diversity of languages within each tribe, often known as local languages, is an intriguing aspect to incorporate into information technology via spoken language identification.

Spoken language identification is the process of utilizing a computer system to determinate the language of a spoken utterance [2]. Language identification refers to spoken communication that can be identified by a computer system [3]. Language identification is the process of distinguishing language from spoken speech [4]. Language identification is at the forefront of assistance in many applications, including multilingual speech systems, spoken language translation, multilingual speech recognition, and human-machine interaction via voice. In the future, language identification research can be continual, particularly in support of multilingual automatic speech recognition (ASR).

One of the uses of ASR in spoken language translation applications that is currently growing rapidly is in assisting different multilingual communication by translating the speaker's native language input and then translating it by machine into the target language (example: translating English to Bahasa Indonesia), or how a content multimedia on the internet can be directly translated into the language that we want. With many regions in Indonesia that can be visited by local and foreign tourists, voice technology is very useful for establishing communication between tourists and natives. Futhermore, there is a lot of digital multimedia content that uses Indonesian local languages as the main language and needs to be automatically translated, so it is critical to identify the language first, so identifying all local languages in Indonesia becomes a challange. The identification of Indonesian local languages using spoken language identification technology has enormous potential to advance tourism potential and digital content in Indonesia.

Language identification occurs in stages. Preprocessing of the speech data is performed initially, followed by extraction of the features of the input speech and measurement of accuracy in the classification process. Many studies on language identification have been conducted, with various feature extraction and classification techniques being used. Several techniques are used to extract features from the audio data, including phone recognition followed by language modeling (PRLM) [5] and parallel phone recognition followed by language modeling (PPRLM) [5] for phonetic approach or perceptual linear prediction (PLP) [5], mel-frequency cepstral coefficient (MFCC) [6]–[8], i-vector [8], [9] and x-vector [10] for the acoustic approx neural networks [11], convolutional neural networks (CNN) [12], [13], logistic regression (LR) [8], PLDA [14], Gaussian mixture model (GMM) [15], [16], support vector machine [17], [18] are among techniques used to classify the language spoken.

Safitri *et al.* [5] used PRLM and PPRLM in their previous research on Indonesian local languages. PPRLM has an accuracy of 73.92%, while PRLM has an accuracy of 64.7%. These findings were obtained using a dataset of speech corpora in three Indonesian local languages (Javanese, Sundanese, and Minangkabau) that were independently recorded. Abdurrahman *et al.* [8] used an acoustic approach with i-vector and x-vector extraction features with probabilistic linear discriminant analysis (PLDA) and LR classifications to study three Indonesian local languages. As a result, the x-vector performs best when using PLDA, while the i-vector outperforms the x-vector when using LR. Wicaksana *et al.* [7] classified MFCC features using random forest (RF), GMM, and k-nearest neighbor (KNN) methods. According to the results, KNN achieved the highest accuracy in 30 s conditions, up to 98.88%, followed by RF at 95.55% and GMM at 82.24%.

Draghici *et al.* [19] classified images using the mel-spectrogram feature and CNN and convolutional recurrent neural network (CRNN). Rammo *et al.* [20] used MFCC and CNN achieve 99.8%, Moreno *et al.* [21] got a 45% increase in performance on energy efficiency ratio (EER) and Cavg at 3s and 10s conditions, Sarthak *et al.* [22] obtained 93.7% accuracy on 1D-ConvNet and 95.4% accuracy on 2D-ConvNet, respectively. Those named Vince *et al.* [23] with EERs of 30% and 20% for 3s and 10s utterances, respectively, Diez *et al.* [24] using convolutional deep neural networks (CDNNs) with increased EER, Krishna and Patil [25] obtaining 95.90% accuracy results in 4 s of sound conditions using ResNet-long term short memory (LSTM) multi-head self-attention (MHA), and Mukherjee *et al.* [26] obtains 94.6% accuracy using MFCC and CCN.

The author wishes to expand on the work of Safitri *et al.* [5], Abdurrahman *et al.* [8], and Wicaksana *et al.* [7]. The proposed study will employ four Indonesian dialects: Javanese, Sundanese, Minangkabau, and Buginese. The extracted feature is MFCC, and the proposed classification technique using a deep learning model is artificial neural network (ANN), CNN, and LSTM, which have not been studied in Indonesian local languages previously.

## 2. RESEARCH METHOD

### 2.1. Dataset

This study will use four Indonesian local languages consisting of Javanese, Sundanese, Minangkabau, and Buginese. Datasets from each local language will be collected from the internet. Javanese and Sundanese datasets will be collected from the openslr.org site, while Minangkabau and Buginese datasets will be collected from the YouTube site. The total duration will be as much as 200 minutes for each local language. Table 1 describes the Javanese, Sundanese, Minangkabau, and Buginese datasets used. All audio data will be saved to wav format, with a bit rate of 16 kbps and a sample rate 16,000 Hz.

Table 1. Total dataset duration

| Language | Duration (minutes) |
|---|---|
| Javanese | 200 |
| Sundanese | 200 |
| Minangkabau | 200 |
| Buginese | 200 |

### 2.2. Preprocessing

The data that has been collected for preprocessing is 200 minutes in each language. For dataset testing and validation, the dataset will be divided into three parts: speech files with a duration of 3 s, 10 s, and

30 s shown in Table 2. Data from YouTube for Minangkabau and Buginese, will be reprocessed beginning with noise removal from recorded stories. Then from the data, the part that contains the background song will be cut off. After the speech data is obtained, the speech data that has been obtained will be combined into one speech file with a duration of 200 minutes and the silent part will be removed. The Sundanese and Javanese datasets will also be combined into one speech file with a duration of 200 minutes. Each speech file will be divided into training data, validation data, and testing data with a composition of 80:10:10 for each data.

Table 2. Total dataset by duration of 3 s, 10 s, and 30 s

| Language | Duration (minutes) | 3 (s) | 10 (s) | 30 (s) |
|---|---|---|---|---|
| Javanese | 200 | 4001 | 1201 | 401 |
| Sundanese | 200 | 4001 | 1201 | 401 |
| Minangkabau | 200 | 4001 | 1201 | 401 |
| Buginese | 200 | 4001 | 1201 | 401 |

### 2.3. Model creation

This research will use the architectural model of ANN, CNN, and LSTM with feature extraction of MFCC. The feature extraction of the MFCC process will use the librosa library. The feature extraction parameters are set to determine the number of MFCCs up to 13, the sample rate at 16,000 Hz, fast fourier transform (FFT) of 2,048, and hop_legth 512.

After the extraction process is carried out, the next stage is model development. The deep learning models proposed in this research are ANN, CNN, and LSTM models. We set initial hyperparameter tuning to train data validation for each model, as shown in Table 3. For each model that is trained, the validation values for each sound time duration (3 s, 10 s, and 30 s) will be summed, then the best average accuracy value will be taken from each model.

Table 3. Initial hyperparameter tuning

| Hyperparameter | Value |
|---|---|
| EPOCH | 30,50 |
| Batch Size | 32,64 |
| Loss Function | *sparse_categorical_crossentropy* |
| Optimizer | *Adam* |
| Learning rate | 0.0001 dan 0.001 |

### 2.4. Evaluation

This study uses a confusion matrix evaluation model to measure the performance of the classification model and make predictions on the test data. The experiment will be divided into two sections: tuning the parameters for each model using validation dataset, and testing the final model using test set. After obtaining the results of the evaluation, an analysis of the results will be carried out. The results of the analysis will then answer the problem formulation. The data output consists of 4 labels. Every label will represent Javanese, Sundanese, Minangkabau, and Buginese languages so that the confusion matrix used is a type of multi-class classification.

## 3. RESULTS AND DISCUSSION

These sections present the experimental results for automatic language identification using deep learning. We trained every model with each duration (3 s, 10 s, and 30 s) with the initial hyperparameter tuning shown in Table 3. Data train and data validation will used to get average accuracy in validation data. The best accuracy for each model ANN, CNN, and LSTM will be processed further with data testing. Table 4 shows that ANN (1), ANN (2), and ANN (3) got the best result in average accuracy. Table 5 shows the results of CNN models, and got CNN (2) and CNN (4) as the models with the best average accuracy. Table 6 shows that LSTM (4) had the best average accuracy of all LSTM models.

Table 4. Average accuraty for ANN model

| Model | Hyperparameter ANN | | | | Validation data accuracy | | | Average accuracy (%) |
|---|---|---|---|---|---|---|---|---|
| | Learning rate | Batchsize | Optimizer | Epoch | 3 s (%) | 10 s (%) | 30 s (%) | |
| ANN (1) | 0.0001 | 32 | Adam | 30 | 77.6 | 81.3 | 86.1 | 82 |
| ANN (2) | 0.0001 | 32 | Adam | 50 | 80 | 80.5 | 86.7 | 82 |
| ANN (3) | 0.0001 | 64 | Adam | 30 | 78 | 80.7 | 82.3 | 80 |
| ANN (4) | 0.0001 | 64 | Adam | 50 | 78.4 | 80 | 86.1 | 82 |

Table 5. Average accuraty for ANN model

| Model | Hyperparameter ANN | | | | Validation data accuracy | | | Average accuracy (%) |
|---|---|---|---|---|---|---|---|---|
| | Learning rate | Batchsize | Optimizer | Epoch | 3 s (%) | 10 s (%) | 30 s (%) | |
| CNN (1) | 0.0001 | 32 | Adam | 30 | 86 | 87.4 | 82.3 | 85 |
| CNN (2) | 0.0001 | 32 | Adam | 50 | 86.8 | 87.8 | 90.5 | 88 |
| CNN (3) | 0.0001 | 64 | Adam | 30 | 84.6 | 85.5 | 88.6 | 86 |
| CNN (4) | 0.0001 | 64 | Adam | 50 | 85.8 | 89.5 | 88.6 | 88 |

Table 6. Average accuraty for ANN model

| Model | Hyperparameter ANN | | | | Validation data accuracy | | | Average accuracy (%) |
|---|---|---|---|---|---|---|---|---|
| | Learning rate | Batchsize | Optimizer | Epoch | 3 s (%) | 10 s (%) | 30 s (%) | |
| LSTM (1) | 0.0001 | 32 | Adam | 30 | 82.4 | 75 | 79.2 | 79 |
| LSTM (2) | 0.0001 | 32 | Adam | 50 | 83.1 | 82.1 | 79.2 | 81 |
| LSTM (3) | 0.001 | 64 | Adam | 30 | 86.7 | 90 | 71.7 | 83 |
| LSTM (4) | 0.001 | 64 | Adam | 50 | 87.7 | 88.6 | 81.1 | 86 |

After comparing the training and validation results, the model is tested with a data test using a confusion matrix. The model used is the model with the best average accuracy among the ANN, CNN, LSTM models obtained in Tables 4-6. Each model with the best accuracy will be evaluated using a confusion matrix for each speech duration. The results of the comparison of the performance evaluation of each model are presented in Tables 7-9.

Table 7. The best test results against test data duration 3 s

| Model | Hyperparameter ANN | | | | Confusion matrix | | | |
|---|---|---|---|---|---|---|---|---|
| | Lr | Bs | Opt | Epoch | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
| ANN (1) | 0.0001 | 32 | Adam | 30 | 85.8 | 85.0 | 85.0 | 85 |
| ANN (2) | 0.0001 | 32 | Adam | 50 | 80 | 80 | 79.9 | 80 |
| ANN (4) | 0.0001 | 64 | Adam | 50 | 78.8 | 78.9 | 78.7 | 78.9 |
| CNN (2) | 0.0001 | 32 | Adam | 50 | 87.4 | 85.5 | 85.2 | 85.5 |
| CNN (4) | 0.0001 | 64 | Adam | 50 | 86 | 86.1 | 86 | 86.1 |
| LSTM (4) | 0.001 | 64 | Adam | 50 | 87.3 | 87.2 | 87.2 | 87.2 |

Table 8. The best test results against test data duration 10 s

| Model | Hyperparameter | | | | Confusion matrix | | | |
|---|---|---|---|---|---|---|---|---|
| | Lr | Bs | Opt | Epoch | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
| ANN (1) | 0.0001 | 32 | Adam | 30 | 80 | 79.6 | 79.5 | 79.6 |
| ANN (2) | 0.0001 | 32 | Adam | 50 | 85.3 | 85.4 | 85.3 | 85.4 |
| ANN (4) | 0.0001 | 64 | Adam | 50 | 79.8 | 79.8 | 79.8 | 79.8 |
| CNN (2) | 0.0001 | 32 | Adam | 50 | 86.6 | 85.6 | 85.4 | 85.6 |
| CNN (4) | 0.0001 | 64 | Adam | 50 | 87.1 | 87.1 | 87 | 87.1 |
| LSTM (4) | 0.001 | 64 | Adam | 50 | 90 | 89.8 | 89.7 | 88.8 |

Table 9. The best test results against test data duration 30 s

| Model | Hyperparameter | | | | Confusion matrix | | | |
|---|---|---|---|---|---|---|---|---|
| | Lr | Bs | Opt | Epoch | Precision (%) | Recall (%) | F1-score (%) | Accuracy (%) |
| ANN (1) | 0.0001 | 32 | Adam | 30 | 85.8 | 85 | 85 | 85 |
| ANN (2) | 0.0001 | 32 | Adam | 50 | 84.8 | 84.4 | 84.6 | 84.4 |
| ANN (4) | 0.0001 | 64 | Adam | 50 | 88.9 | 87.5 | 87.5 | 87.5 |
| CNN (2) | 0.0001 | 32 | Adam | 50 | 88.3 | 88.1 | 88.1 | 88.1 |
| CNN (4) | 0.0001 | 64 | Adam | 50 | 83.1 | 81.9 | 81.9 | 81.9 |
| LSTM (4) | 0.001 | 64 | Adam | 30 | 88.6 | 88.1 | 88.1 | 88.1 |

Table 7 shows that the model with the highest accuracy value is the LSTM (4), which has an accuracy of 87.2% and an f1-score of 87.2%. This model is followed by the CNN model (4), which has an accuracy of 86.1%, CNN (2), which has an accuracy of 85.5%, ANN (1), which has an accuracy of 85%, ANN (2), which has an accuracy of 80%, and ANN (4), which has an accuracy of 78.9%. Table 8 shows that with a duration of 10 s, the model with the highest accuracy value is the LSTM model (4), which has an accuracy value of 88.8% and an f1-score of 87%. This model is followed by the CNN model (4), which has an accuracy of 87.1%, CNN (2), which has an accuracy of 85.6%, ANN (2), which has an accuracy of 85.4%, ANN (3), which has an accuracy of 79.8%, and Table 9, with a duration of 30 s, shows that the models with the best accuracy values are the LSTM (4) and CNN (2) models, both of which have an Accuracy value of 88.1% and an f1-score of 88.1%. This is followed by the ANN (4) model, which has an accuracy of 87.15%, the ANN (2) model, which has an accuracy of 84.4%, the ANN (1) model, which has an accuracy of 85%, and the CNN (4) model.

## 4. CONCLUSION

This work uses ANN, CNN, and LSTM approaches with the MFCC feature to create an identification model for the regional languages of Javanese, Sundanese, Minangkabau, and Buginese. At speech lengths of 3 s, 10 s, and 30 s, each classification method was compared to the others. According to the experimental findings, the LSTM (4) and CNN (2) model achieved the maximum accuracy with a value of 88.1% at a speech duration of 30 s, followed by the ANN (4) model with 87.5%.

For a speech length of 10 s, the LSTM (4) achieves the highest accuracy with an accuracy value of 88.8%, followed by CNN (4) with a value of 87.1%, CNN (2) with a value of 85.6%, and ANN (2) with a value of 85.4%. For a speech time of 3 s, the LSTM (4) model also received the highest score with an Accuracy value of 87.2%, followed by CNN (4) 86.1% and CNN (2) 85.5%. Conclusion: for each speech duration (3 s, 10 s, and 30 s), the LSTM model achieves the maximum accuracy, followed by the CNN model in second place and the ANN model in third place.

## REFERENCES

[1] Ministry of Education and Culture, "Language and language maps in Indonesia," (in *Indonesia*) Kemdikbud, “Bahasa dan peta bahasa di Indonesia,” 2019. https://petabahasa.kemdikbud.go.id/infografisdir/783Leflet_Peta_Bahasa_2019.pdf.

[2] H. S. Das and P. Roy, “A deep dive into deep learning techniques for solving spoken language identification problems,” in *Intelligent Speech Signal Processing*, Elsevier, 2019, pp. 81–100.

[3] M. A. Zissman and K. M. Berkling, “Automatic language identification,” *Speech Communication*, vol. 35, no. 1–2, pp. 115–124, Aug. 2001, doi: 10.1016/S0167-6393(00)00099-6.

[4] R. Bedyakin and N. Mikhaylovskiy, “Low-resource spoken language identification using self-attentive pooling and deep 1D time-channel separable convolutions,” in *arXiv-Audio and Speech Processing*, Jun. 2021, pp. 1012–1020, doi: 10.28995/2075-7182-2021-20-1012-1020.

[5] N. E. Safitri, A. Zahra, and M. Adriani, “Spoken language identification with phonotactics methods on Minangkabau, Sundanese, and Javanese languages,” *Procedia Computer Science*, vol. 81, pp. 182–187, 2016, doi: 10.1016/j.procs.2016.04.047.

[6] L. Ferrer, “Joint probabilistic linear discriminant analysis,” *arXiv-Machine Learning*, no. 4, pp. 1–13, 2017, doi: 10.48550/arXiv.1704.02346.

[7] V. S. Wicaksana and A. Zahra, “Spoken language identification on local language using MFCC, random forest, KNN, and GMM,” *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, pp. 394–398, 2021, doi: 10.14569/IJACSA.2021.0120548.

[8] A. I. Abdurrahman and A. Zahra, “Spoken language identification using i-vectors, x-vectors, PLDA and logistic regression,” *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 2237–2244, Aug. 2021, doi: 10.11591/eei.v10i4.2893.

[9] N. Dehak, P. A. T. -Carrasquillo, D. Reynolds, and R. Dehak, “Language recognition via i-vectors and dimensionality reduction,” in *Interspeech 2011*, Aug. 2011, pp. 857–860, doi: 10.21437/Interspeech.2011-328.

[10] D. Snyder, D. G. -Romero, A. McCree, G. Sell, D. Povey, and S. Khudanpur, “Spoken language recognition using x-vectors,” in *The Speaker and Language Recognition Workshop (Odyssey 2018)*, Jun. 2018, pp. 105–111, doi: 10.21437/Odyssey.2018-15.

[11] P. Heracleous, K. Takai, K. Yasuda, Y. Mohammad, and A. Yoneyama, “Comparative study on spoken language identification based on deep learning,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, Sep. 2018, pp. 2265–2269, doi: 10.23919/EUSIPCO.2018.8553347.

[12] S. Mukherjee, N. Shivam, A. Gangwal, L. Khaitan, and A. J. Das, “Spoken language recognition using CNN,” in *2019 International Conference on Information Technology (ICIT)*, Dec. 2019, pp. 37–41, doi: 10.1109/ICIT48102.2019.00013.

[13] R. Ubale, Y. Qian, and K. Evanini, “Exploring end-to-end attention-based neural networks for native language identification,” in *2018 IEEE Spoken Language Technology Workshop (SLT)*, Dec. 2018, pp. 84–91, doi: 10.1109/SLT.2018.8639689.

[14] M. K. Rai, Neetish, M. S. Fahad, J. Yadav, and K. S. Rao, “Language identification using PLDA based on i-vector in noisy environment,” in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2016, pp. 1014–1020, doi: 10.1109/ICACCI.2016.7732177.

[15] D. V. Leeuwen and N. Brummer, “Channel-dependent GMM and multi-class logistic regression models for language recognition,” in *2006 IEEE Odyssey-The Speaker and Language Recognition Workshop*, Jun. 2006, pp. 1–8, doi: 10.1109/ODYSSEY.2006.248094.

[16] Y. Xu, J. Yang, and J. Chen, “Methods to improve gaussian mixture model for language identification,” in *2010 International Conference on Measuring Technology and Mechatronics Automation*, Mar. 2010, pp. 656–659, doi: 10.1109/ICMTMA.2010.545.

[17] S. Mohanty, “Phonotactic model for spoken language identification in Indian language perspective,” *International Journal of Computer Applications*, vol. 19, no. 9, pp. 18–24, Apr. 2011, doi: 10.5120/2389-3164.

[18] R. W. M. Ng, C.-C. Leung, T. Lee, B. Ma, and H. Li, “Prosodic attribute model for spoken language identification,” in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2010, pp. 5022–5025, doi: 10.1109/ICASSP.2010.5495070.

[19] A. Draghici, J. Abeßer, and H. Lukashevich, “A study on spoken language identification using deep neural networks,” in *Proceedings of the 15th International Conference on Audio Mostly*, Sep. 2020, pp. 253–256, doi: 10.1145/3411109.3411123.

[20] F. M. Rammo and M. N. Al-Hamdani, “Detecting the speaker language using CNN deep learning algorithm,” *Iraqi Journal for Computer Science and Mathematics*, vol. 3, no. 1, pp. 43–52, Jan. 2022, doi: 10.52866/ijcsm.2022.01.01.005.

[21] I. L. -Moreno, J. G. -Dominguez, D. Martinez, O. Plchot, J. G. -Rodriguez, and P. J. Moreno, “On the use of deep feedforward

neural networks for automatic language identification," *Computer Speech & Language*, vol. 40, pp. 46–59, Nov. 2016, doi: 10.1016/j.csl.2016.03.001.

[22] Sarthak, S. Shukla, and G. Mittal, "Spoken language identification using convNets," *arXiv-Computation and Language*, pp. 252–265, 2019, doi: 10.1007/978-3-030-34255-5_17.

[23] M. Jin, Y. Song, and I. McLoughlin, "End-to-end DNN-CNN classification for language identification," *Proceedings of the World Congress on Engineering*, vol. 1, pp. 199–203, 2017.

[24] A. L. -Diez, R. Z. -Candil, J. G. -Dominguez, D. T. Toledano, and J. G. -Rodriguez, "An end-to-end approach to language identification in short utterances using convolutional neural networks," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2015, pp. 403–407, doi: 10.21437/interspeech.2015-164.

[25] D. N. Krishna and A. Patil, "End-to-end language identification using multi-head self-attention and 1D convolutional neural networks," *arXiv-Audio and Speech Processing*, pp. 1–5, 2021, doi: 10.48550/arXiv.2102.00306.

[26] H. Mukherjee, A. Dhar, S. Phadikar, and K. Roy, "RECAL—a language identification system," in *2017 International Conference on Signal Processing and Communication (ICSPC)*, Jul. 2017, pp. 300–304, doi: 10.1109/CSPC.2017.8305857.

## BIOGRAPHIES OF AUTHORS

**Panji Wijonarko** is currently a graduate student in Computer Science Departement of Bina Nusantara University. He received his bachelor's degree in STMIK INTI Indonesia in 2013. He can be contacted at email: panji.wijonarko@binus.ac.id.

**Amalia Zahra** is a lecturer at the Master of Information Technology, Bina Nusantara University, Indonesia. She received her bachelor's degree in computer science from the Faculty of Computer Science, University of Indonesia (UI) in 2008. She does not have a master's degree. Her Ph.D was obtained from the School of Computer Science and Informatics, University College Dublin (UCD), Ireland in 2014. Her research interests cover various fields in speech technology, such as speech recognition, spoken language identification, speaker verification, speech emotion recognition, and so on. Additionally, she also has interest in natural language processing (NLP), computational linguistics, machine learning, and artificial intelligence. She can be contacted at email: amalia.zahra@binus.edu.