Syllable-Based Indonesian Automatic Speech Recognition

Danny Henry Galatang and Suyanto

School of Computing, Telkom University
Jalan Telekomunikasi No. 01, Terusan Buah Batu, Bandung 40257, Indonesia
dannygalatang@gmail.com, suyanto@telkomuniversity.ac.id

Abstract: The syllable-based automatic speech recognition (ASR) systems commonly perform better than the phoneme-based ones. This paper focuses on developing an Indonesian monosyllable-based ASR (MSASR) system using an ASR engine called SPRAAK and comparing it to a phoneme-based one. The Mozilla DeepSpeech-based end-to-end ASR (MDS-E2EASR), one of the state-of-the-art models based on character (similar to the phoneme-based model), is also investigated to confirm the result. Besides, a novel Kaituoxu Speech-Transformer (KST) E2EASR is also examined. Testing on the Indonesian speech corpus of 5,439 words shows that the proposed MSASR produces much higher word accuracy (76.57%) than the monophone-based one (63.36%). Its performance is comparable to the character-based MDS-E2EASR, which produces 76.90%, and the character-based KST-E2EASR (78.00%). In the future, this monosyllable-based ASR is possible to be improved to the bisyllable-based one to give higher word accuracy. Nevertheless, extensive bisyllable acoustic models must be handled using an advanced method.

Keywords: monosyllable, speech recognition, read-speech corpus, Indonesian

1. Introduction

An automatic speech recognition (ASR) system is usually developed using a state model of triphone, three context-dependent phonemes. Since 2000 many syllable-based ASR systems have been developed for English with higher performances than the phoneme-based ones [1], [2], [3]. However, they require much more acoustic models, which are commonly implemented using Hidden Markov Model (HMM), dynamic time wrapping, or even deep learning.

This paper discusses an early effort to develop an Indonesian context-independent monosyllable-based ASR using a speech corpus of 44,000 utterances (collected from 400 speakers that read 110 sentences each) [4], [5]. The developed monosyllable-based ASR is then analyzed and compared to the context-independent monophone-based ASR based on their word accuracies and error rates. Besides, it is also compared to both deep learning-based models: MDS and KST, which are some of the most popular modern E2EASR systems, to confirm the performances of the proposed ASR system.

Based on the classification method proposed by Dauer in [6], the Indonesian language is one of the simple syllabic languages. It has twelve syllable structures (or patterns), as listed in Table 1, adapted from [7]. In [8], a study on a vocabulary of 50 k words selected from the Indonesian dictionary called KBBI shows that the Indonesian language is dominated by open syllables (56.63%), which are structures with a vowel at the end. The rests 43.37% are closed syllables (structures with a consonant at the end). The Indonesian has mostly simple CV syllables (C = consonant and V = vowel), up to 50.63% [8]. Hence, Indonesian is categorized as a simple language, where the syllabic complexity is low. Meanwhile, English is categorized as a complex language. It has various both open and closed syllables and much lower CV syllables of 35% [6]. These facts indicate that an Indonesian ASR will be simpler to be developed than the English ASR.

However, the Indonesian has many long words containing 7, 8, even 9 syllables, such as '*menindaklanjuti*' (to follow up), '*memperjualbelikannya*' (to trade it), '*pertelekomunikasian*' (everything related to telecommunication). The study in [8] shows that, on average, any Indonesian word has 3.20 syllables.

Received: April 5th, 2018. Accepted: October 12th, 2020
DOI: 10.15676/ijeei.2020.12.4.2

Table 1. Twelve structures of Indonesian syllables

Number	Syllable structure	Example
1	V	***a***.*ku*
2	CV	***ba***.*ca*
3	CCV	***dra***.*ma*
4	CCCV	***stra***.*te.gi*
5	VC	***ab***.*di*
6	CVC	***cin***.*ta*
7	VCC	***eks***.*tra*
8	CVCC	***teks***.*tur*
9	CVCCC	***korps***
10	CCVC	***prak***.*tik*
11	CCVCC	*kom*.***pleks***
12	CCCVC	***struk***.*tur*

Thus, it has much more polysyllabic words (up to 98.30%) than the monosyllabic ones (only 1.70%). Compared to the Indonesian language, English has lower polysyllabic words (80%) and more monosyllabic words (up to 20%) based on the Wordsmyth dictionary [9]. This statistic seems to be contradictive to the previous facts, but the long syllables in a speech are easy to be recognized using a syllable-based ASR as long as they are categorized as open syllables.

Another interesting fact is that Indonesian has so many phonotactic rules that not all consecutive phonemes are valid to form a word [7]. For instance, a consecutive phonemes /kt/ in /*struktur*/ (structure) should be separated since Indonesian syllables never contain /kt/. These phonotactic rules reduce the possibility of a word generating by the sequence of phonemes. Based on those facts, the Indonesian ASR is expected to perform better if it is developed using a syllable-based approach compared to the phoneme-based one.

The design of Indonesian syllable-based ASR is explained in section 2. The Indonesian speech corpus, syllable-based dictionary, text corpus, and the experimental setup are described in section 3. Next, section 4 discusses the performances of both monophone-based ASR and monosyllable-based one regarding their word accuracies and error rates. The last section gives both conclusion and future work.

2. Indonesian syllable-based ASR

There are two approaches in developing an ASR system. The first one is a traditional pipeline model, which is commonly built using the hidden Markov model (HMM). The second one is a modern E2EASR, which is generally developed using deep learning (DL). The DL-based approach is capable of replacing some algorithms and processing steps in the traditional model. MDS and KST are two of the most popular modern E2EASR. MDS is commonly trained using the character-based technique. It uses the recurrent neural network (RNN) [10], which is considered the same as the acoustic models built in HMM-based ASR [11], [12]. Whereas, KST is a sequence-to-sequence (S2S) model without recurrence [13]. It can be trained more efficiently since it exploits an attention mechanism, which is used to learn the positional dependencies. Furthermore, it can be exploited to restore capitalization and punctuation in the ASR output to improve readability [14]. In [15], it is reported to obtain high performance for both Latvian and English ASR systems and language understanding.

However, since both MDS and KST models need a large speech corpus, they have a problem for low-resource languages, such as Indonesian. In the world, there are more than 95% of languages are low-resource [16], which are hard to provide a vast annotated corpus of speech to be learned by both modern E2EASR systems. There are three solutions to this problem. Firstly, data augmentation can be used to generate large amounts of synthetic data [17]. Secondly, a transfer learning can be used for similar languages, such as English, German, and Dutch, where some layers of convolutional neural networks (CNN) can be entirely or

highly transferable [18], [19]. Thirdly, a Map and Relabel (MaR) can also be efficiently used to quickly construct an ASR system. For instance, MaR successfully trains a reasonable ASR Uyghur system using a small corpus of 500 utterances [20]. Practically, those three solutions have some limitations. The data augmentation is commonly used to synthesize various noisy utterances, not create new clean ones. Both transfer learning and MaR generally produce high performances for some similar languages, not for the different ones with many specificities.

Hence, some researchers prefer to develop a traditional pipelined-ASR based on either phoneme or syllable. In general, the phoneme-based ASR systems are developed using a frame-based technique [3]. In this technique, no automatic segmentation of phonemes is needed. An input speech is independently seen as a frame sequence that statistically constant, as illustrated in Figure 1. An Indonesian utterance '*ke bab satu*' ('go to chapter one' in English) is decomposed into some sequences of frames. The length of a frame is usually 25 milliseconds (ms) with 10 ms shifting. Each frame is extracted to produce some features, which is commonly using Mel-frequency cepstral coefficients (MFCC). Next, the sequences of features are recognized using a classifier, e.g., HMM, dynamic time wrapping, or deep learning, to get some phonemes. By using a dictionary of phoneme lexicons, the sequence of phonemes is concatenated to form some words. Then, the sequences of words are combined to generate several possible sentences, for example: '*kebab satu*' ('kebab one') and '*ke bab satu*' ('go to chapter one'). Finally, a language model (usually a trigram language model) selects the best (most probable) sentence as an output: '*ke bab satu*'.

In the syllable-based ASR, an input speech is also seen as a frame sequence, but the acoustic models are developed in a slightly different way, as illustrated in Figure 2. Each frame is extracted to produce some features. Next, the sequences of features are recognized using an HMM to get some syllables. By using a dictionary of syllable lexicons, the sequences of syllables are concatenated to form some words. Then, the sequences of words are combined to generate some possible sentences. Finally, a trigram language model selects the most probable sentence. In this research, both models are developed using the same speech and text corpora, and their performances are then compared.

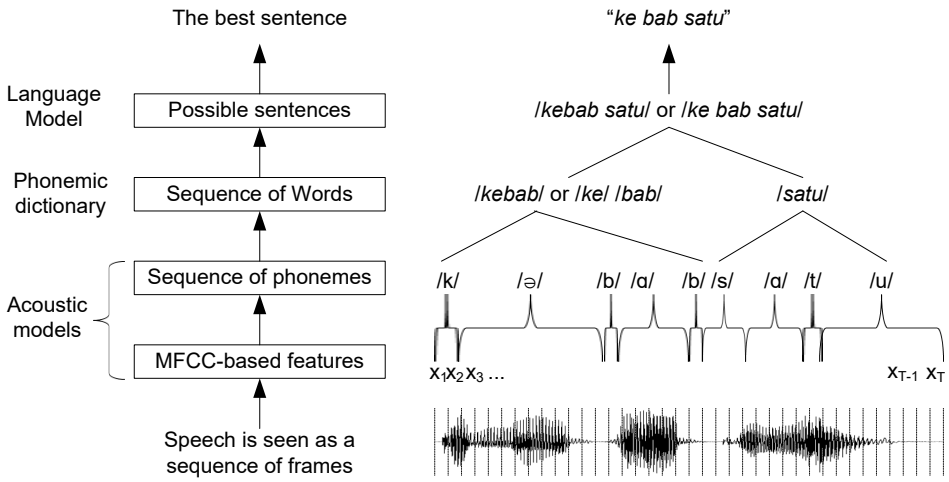

Figure 1. Phoneme-based ASR system

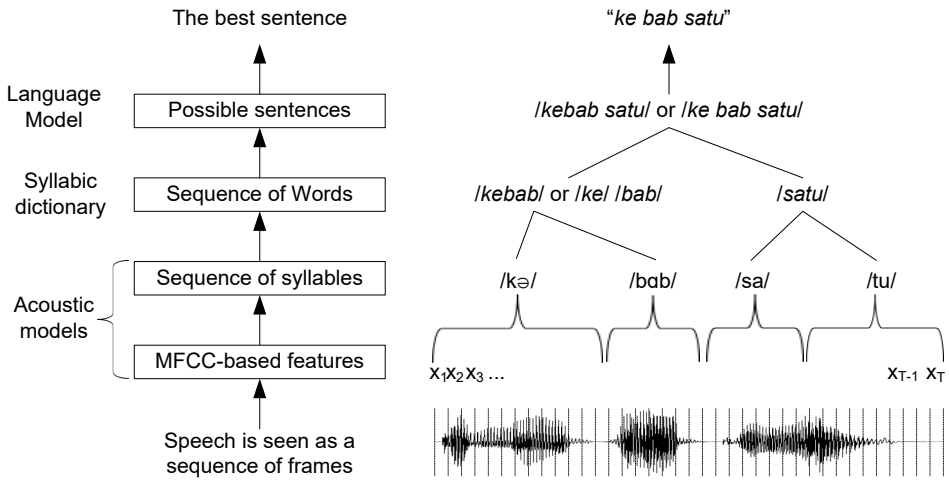

Figure 2. Syllable-based ASR system

3. Experimental Setup

Both acoustic and language models used in this research are the Hidden Markov Model (HMM)-based state model and the trigram language model. Those models require large speech and text corpus for the training process. A reading-speech corpus of 400 speakers with four major dialects [4] is used in this research. This corpus will be trained to develop HMM-based acoustic models. A text corpus of 5 k sentences [5] will be used to develop a trigram language model.

A. Syllable Lexicons

A dictionary of syllable lexicons is needed to generate a set of syllable-based acoustic models. It can be developed in two stages. First, each word in the dictionary is converted into a sequence of phonemes using a grapheme-to-phoneme (G2P) conversion [21], where the monophone pronunciation accuracy is 99.07%. Second, the phoneme sequence is syllabified using the Indonesian syllabification system [8], where the accuracy of syllable pronunciation is 99.36%. In this research, a validation is manually performed to update the few incorrect syllable lexicons. Some examples of syllable lexicons are listed in Table 2.

Table 2. Indonesian dictionary of syllable lexicons

Word	Syllable lexicons
administrasi	*ad mi nis tra si*
besok	*be sok*
favorit	*fa fo rit*
masalah	*ma sa lah*
obat	*o bat*
syarief	*sya rief*
telekomunikasi	*te le ko mu ni ka si*

B. Context-Independent Unit for Syllable

The context-independent unit will list all possible syllables that represent all words in the speech corpus as well as the lexicon model. The 5,439 unique words from 7,150 utterances in the corpus will be modeled into 2,840 monosyllables.

C. ASR Models

In this research, an ASR engine called speech processing, recognition, and automatic annotation kit (SPRAAK) is used to generate both monophone-based and monosyllable-based

acoustic models. In developing the acoustic models, the HMM parameters are mostly set to the default values, as illustrated in Table 3. The SRILM toolkit is exploited to develop a word-based trigram language model, where it is built with a back-off technique for a text corpus of 5 k sentences. Both acoustic and language models are then used for training as well as testing.

Table 3. Parameters of HMM

Word	**Syllable lexicons**
HMM_VNUM	7
DIM1	7650
DIM2	1
NUNIT	34
NSTATE	102
TOPOLOGY	LEFT_TO_RIGHT
TPDIM	2
DENSTYPE	SC_HMM
NMIX	1
TRANS_SCALE	LOGARITHMIC
WEIGHT_SCALE	LOGARITHMIC
TOTAL_MIXDIM	7140
EXTENDED	PARAMSET
NMVG	285
OVLEN	39
REDUCED_SC_HMM	YES

Meanwhile, the parameters of the MDS-E2EASR is also mostly set to the default parameters, as illustrated in Table 4 [22]. Three sensitive parameters: learning rate, dropout, and epochs are carefully tuned to be 0.00005, 150, and 0.2275 that obtain the best performance. A too-high learning rate makes the model premature converge on a high error rate. In contrast, a too-low learning rate slightly reduces the error so that the training is time-consuming. A too-large dropout rate may remove the good network weights, whereas a too-small dropout rate can cause the model to overfit. A too-big epoch makes the model overfit while a too-small epoch causes it to underfit.

Table 4. Parameters of MDS

Parameter	Value
Training batch	88
Validation batch	40
Testing batch	40
Hidden neuron	512
Learning rate	0.00005
Epoch	150
Dropout rate	0.2275
Beam width	1024
lm alpha	1.50
lm beta	2.10

The parameters of the KST-E2EASR is set to the default parameters, as illustrated in Table 5 [22]. Two sensitive parameters: number of encoders and number of decoders, are set to be ten that produce the lowest WER as well as adjust the language specificity. They can avoid overfitting and also optimize the loss function.

Table 5. Parameters of KST

Parameter	Value
Input Layer	80
Encoder Stacks	6
Multi Attention Head (MHA)	8
Dimension of Key (DK)	64
Dimension of Value (DV)	64
Dimension of Model (DM)	512
Dimension of Inner (DI)	512
Dropout Rate	0.10
Positional Encoding max length	5000
Dim of Decoder Embedding	512
Decoder Stacks	6
Share Decoder Embedding	1
Label Smoothing	0.10
Epochs	150
Shuffle	1 (true)
Batch Size	16

4. Results and Discussions

The speech recognition engine SPRAAK v1.1.366 is used as a speech decoder. The experimental results are listed in two tables. Table 6 shows the word accuracy for the three developed systems: monophone-based ASR (MPASR), monosyllable-based ASR (MSASR), character-based MDS-E2EASR, character-based KST-E2EASR. The results show that the proposed MSASR gives a much higher word accuracy (76.57%) than MPASR (only 63.36%). It is comparable to the MDS-E2EASR that produces a word accuracy of 76.90% and the KST-E2EASR that produces 78.00%. These results inform that the base-component is critical in developing an ASR system. A traditional HMM-based ASR model that uses syllables as the base-components can give a competitive accuracy.

Table 6. Accuracy produced by MPASR, MSASR, and MDS-E2EASR

Model	Word Accuracy
MPASR	63.36%
MSASR	76.57%
MDS-E2EASR	76.90%
KST-E2EASR	**78.00%**

Meanwhile, Table 7 describes the error rate for each type of error. Those results show that the MSASR is capable of reducing errors produced by the MPASR. It is capable of significantly reducing all types of errors. The insertions can be relatively decreased by 11.75% (from 5,489 to 4,844). Meanwhile, both deletions and substitutions are relatively reduced by up to 18.88% and 14.23%, respectively. It concludes that syllable is a better base-component of ASR than phoneme.

Table 7. Error rate of monophone-based ASR and monosyllable-based ASR

Error type	MPASR	MSASR
Insertions	5,489	4,844
Deletions	556	451
Substitutions	21,165	18,153

Analyzing the results in more detail finds some improvements in recognizing monosyllable-based ASR. These improvements are caused by the following reasons:

The observation range of monosyllable-based ASR is larger than monophone-based ASR. For example, the speech 'syarief' is wrongly recognized as 'sharif' on the monophone-based ASR but it is correctly recognized as 'syarief' on the monosyllable-based ASR.

The monosyllable-based ASR has lower ambiguity in generating words than the monophone-based one since the syllable structures are composed of several particular phonemes limited by the Indonesian phonotactic rules. For instance, two consecutive phonemes of 'mz' are never appeared in the formal words nor named entity in the Indonesian language. They should be split into different syllables, such as the formal words: 'om.zet' and 'zam.zam' and the named-entities: 'ham.zah', 'im.za', and so on.

The monosyllable-based ASR gives fewer word candidates than the monophone-based one because of the speaker's dialect or uncommon speaking rate. Since a syllable commonly has long frames, the MSASR is not sensitive to both dialect and speaking rate. In contrast, the MPASR is susceptive to both variations since a phoneme may has shorter frames.

However, the MSASR can be enhanced by tuning the HMM's parameters or designing HMM's structure carefully by a human expert. Besides, they also can be automatically optimized using a metaheuristic technique, such as evolutionary algorithm [23], [24], particle swarm optimization [25], [26], firefly algorithm [27], [28], krill herd algorithm [29], [30], modified multi-sonar bat units algorithm [31], grey wolf optimization [32], [33]. Those methods are some of the most popular swarm-based algorithms that give high performance in solving a multi-objective optimization problem.

5. Conclusion

The Indonesian monosyllable-based ASR (MSASR) has been successfully implemented with an absolute improvement of word accuracy up to 13.21% compared to the monophone-based one (MPASR). The proposed MSASR is comparable to character-based state-of-the-art MDS-E2EASR as well as KST-E2EASR. This result proves that the ASR for Indonesian, as a simple language with low syllabic complexity, is much better to be implemented using a syllable-based model. Next, to increase its accuracy, the context-independent monosyllable-based ASR should be extended into a context-dependent bisyllable-based one. However, bisyllable-based ASR needs extensive acoustic models that should be developed and maintained using some advanced methods.

6. References

[1]. A. Ganapathiraju and J. Hamaker, "Syllable-based large vocabulary continuous speech recognition," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 4, pp. 358–366, 2001, doi: 10.1109/89.917681.

[2]. R. Janakiraman, J. C. Kumar, and H. A. Murthy, "Robust syllable segmentation and its application to syllable-centric continuous speech recognition," in *National Conference on Communications (NCC)*, Jan. 2010, pp. 1–5, doi: 10.1109/NCC.2010.5430189.

[3]. X. Liu, M. J. F. Gales, J. L. Hieronymus, and P. C. Woodland, "Investigation of acoustic units for LVCSR systems," in *ICASSP*, 2011, pp. 4872–4875, doi: 10.1109/ICASSP.2011.5947447.

[4]. S. Suyanto, "Signal energy-based automatic speech splitter: a tool for developing speech corpus," *TENCON 2007 - 2007 IEEE Reg. 10 Conf.*, 2007, doi: https://doi.org/10.1109/TENCON.2007.4428892.

[5]. Suyanto and J. Adityatama, "Yooi: An Indonesian Short Message Dictation," *Int. J. Intell. Inf. Process.*, vol. 3, no. 4, pp. 68–74, 2012.

[6]. R. M. Dauer, "Stress-timing and syllable-timing reanalyzed," *J. Phon.*, vol. 11, no. 1, pp. 51–62, 1983.

[7]. H. Alwi, S. Darmowidjojo, H. Lapoliwa, and A. M. Moeliono, *Tata Bahasa Baku Bahasa Indonesia (The Standard Indonesian Grammar)*, 3rd ed. Jakarta: Balai Pustaka, 2014.

[8]. S. Suyanto, S. Hartati, A. Harjoko, and D. Van Compernolle, "Indonesian syllabification using a pseudo nearest neighbour rule and phonotactic knowledge," *Speech Commun.*, vol. 85, pp. 109–118, 2016, doi: http://dx.doi.org/10.1016/j.specom.2016.10.009.

[9]. Y. Marchand, C. R. Adsett, and R. I. Damper, "Automatic syllabification in English: a comparison of different algorithms," *Lang. Speech*, vol. 52, no. Pt 1, pp. 1–27, 2009, doi: 10.1177/0023830908099881.

[10]. D. Amodei *et al.*, "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," *Cornell University Library's arXiv.org*. pp. 1–28, 2015, doi: 10.1145/1143844.1143891.

[11]. A. Hannun *et al.*, "Deep Speech: Scaling up end-to-end speech recognition," *Cornell University Library's arXiv.org*. pp. 1–12, 2014, doi: arXiv:1412.5567v2.

[12]. N. R. Emillia, Suyanto, and W. Maharani, "Isolated word recognition using ergodic hidden markov models and genetic algorithm," *Telkomnika*, vol. 10, no. 1, pp. 129–136, 2012, doi: http://dx.doi.org/10.12928/telkomnika.v10i1.769.

[13]. L. Dong, S. Xu, and B. Xu, "Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2018, vol. 2018-April, pp. 5884–5888, doi: 10.1109/ICASSP.2018.8462506.

[14]. A. Vāravs and A. Salimbajevs, "Restoring punctuation and capitalization using transformer models," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11171 LNAI, pp. 91–102, 2018, doi: 10.1007/978-3-030-00810-9_9.

[15]. M. C. Kenton, L. Kristina, and J. Devlin, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *ArXiv*, 2019.

[16]. Y.-C. Chen, C.-H. Shen, S.-F. Huang, H. Lee, and L. Lee, "Almost-unsupervised Speech Recognition with Close-to-zero Resource Based on Phonetic Structures Learned from Very Small Unpaired Speech and Text Data," *ArXiv*, pp. 1–5, 2018.

[17]. Y. Qian, H. Hu, and T. Tan, "Data augmentation using generative adversarial networks for robust speech recognition," *Speech Commun.*, vol. 114, pp. 1–9, 2019, doi: 10.1016/j.specom.2019.08.006.

[18]. J. A. F. Thompson, M. Schonwiesner, Y. Bengio, and D. Willett, "How Transferable Are Features in Convolutional Neural Network Acoustic Models across Languages?," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2019, vol. 2019-May, pp. 2827–2831, doi: 10.1109/ICASSP.2019.8683043.

[19]. H. Inaguma, J. Cho, M. K. Baskar, T. Kawahara, and S. Watanabe, "Transfer Learning of Language-independent End-to-end ASR with Language Model Fusion," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2019, vol. 2019-May, pp. 6096–6100, doi: 10.1109/ICASSP.2019.8682918.

[20]. Y. Shi, Z. Tang, L. Lit, Z. Zhang, and D. Wang, "Map and Relabel: Towards Almost-Zero Resource Speech Recognition," *2018 Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. APSIPA ASC 2018 - Proc.*, no. November, pp. 591–595, 2018, doi: 10.23919/APSIPA.2018.8659508.

[21]. S. Suyanto, S. Hartati, and A. Harjoko, "Modified Grapheme Encoding and Phonemic Rule to Improve PNNR-Based Indonesian G2P," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 3, pp. 430–435, 2016, doi: https://dx.doi.org/10.14569/IJACSA.2016.070358.

[22]. S. Suyanto, A. Arifianto, A. Sirwan, and A. P. Rizaendra, "End-to-End Speech Recognition Models for a Low-Resourced Indonesian Language," in *2020 8th International Conference on Information and Communication Technology (ICoICT)*, Jun. 2020, pp. 1–6, doi: 10.1109/ICoICT49345.2020.9166346.

[23]. M. H. Aliefa and S. Suyanto, "Variable-Length Chromosome for Optimizing the Structure of Recurrent Neural Network," in ICoDSA 2020, doi: https://doi.org/10.1109/ICoDSA50139.2020.9213012.

[24]. A. C. Rizal and S. Suyanto, "Human-Like Constrained-Mating to Make Genetic Algorithm More Explorative," in *2020 8th International Conference on Information and Communication Technology (ICoICT)*, Jun. 2020, pp. 1–5, doi: 10.1109/ICoICT49345.2020.9166387.
[25]. M. Humam, O. Somantri, M. Boni Abdillah, S. Arif Romadhon, M. Khambali, and R. Rahim, "The application of particle swarm optimization using neural network to optimize classification of employee performance assessment," in *Journal of Physics: Conference Series*, 2019, vol. 1175, no. 1, doi: 10.1088/1742-6596/1175/1/012067.
[26]. K. V. K. Kavuturu and P. Narasimham, "Optimization of Transmission System Security Margin under (N-1) Line Contingency Using Improved PSO Algorithm," *Int. J. Electr. Eng. Informatics*, vol. 12, no. 2, pp. 242–257, 2020, doi: 10.15676/ijeei.2020.12.2.5.
[27]. F. Ahyar and S. Suyanto, "Firefly Algorithm-based Hyperparameters Setting of DRNN for Weather Prediction," in ICoDSA 2020, doi: https://doi.org/10.1109/ICoDSA50139.2020.9212921.
[28]. U. Abdillah and S. Suyanto, "Clustering Nodes and Discretizing Movement to Increase the Effectiveness of HEFA for a CVRP," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 4, pp. 774–779, 2020, doi: https://dx.doi.org/10.14569/IJACSA.2020.01104100.
[29]. M. Abdel-Basset, G.-G. Wang, A. K. Sangaiah, and E. Rushdy, "Krill herd algorithm based on cuckoo search for solving engineering optimization problems," *Multimed. Tools Appl.*, vol. 78, no. 4, pp. 3861–3884, 2019, doi: 10.1007/s11042-017-4803-x.
[30]. M. I. Tawakkal and S. Suyanto, "Exploration-Exploitation Balanced Krill Herd Algorithm for Thesis Examination Timetabling," in ICoDSA 2020, doi: https://doi.org/10.1109/ICoDSA50139.2020.9212837.
[31]. M. A. Tawfeeq, "Optimization of Neural Networks Based on Modified Multi-Sonar Bat Units Algorithm," *Int. J. Electr. Eng. Informatics*, vol. 12, no. 1, pp. 105–116, 2020, doi: 10.15676/ijeei.2020.12.1.9.
[32]. B. Z. Aufa, S. Suyanto, and A. Arifianto, "Hyperparameter Setting of LSTM-based Language Model using Grey Wolf Optimizer," in ICoDSA 2020, doi: https://doi.org/10.1109/ICoDSA50139.2020.9213031.
[33]. A. Tjahjono, D. O. Anggriawan, M. N. Habibi, and E. Prasetyono, "Modified Grey Wolf Optimization for Maximum Power Point Tracking in Photovoltaic System under Partial Shading Conditions," *Int. J. Electr. Eng. Informatics*, vol. 12, no. 1, pp. 94–104, 2020, doi: 10.15676/ijeei.2020.12.1.8.

Danny Henry Galatang received his B.Sc. on Informatics Engineering from Telkom University, Bandung, Indonesia in 2018. His research interests include artificial intelligence, machine learning, deep learning, and computational linguistics.

Suyanto received his B.Sc. on Informatics Engineering from STT Telkom (now: Telkom University), Bandung, Indonesia in 1998, the M.Sc. on Complex Adaptive Systems from Chalmers University of Technology, Goteborg, Sweden, in 2006, and the Ph.D. on Computer Science from Universitas Gadjah Mada in 2016. Since 2000, he joined STT Telkom as a lecturer in the School of Computing. His research interests include artificial intelligence, machine learning, deep learning, swarm intelligence, speech processing, and computational linguistics. Orcid: https://orcid.org/0000-0002-8897-8091.