Evidences of correspondences

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

1. **First submission (31 December 2018)**
2. LoA with Major Revision (20 March 2019)
3. Respond to Reviewers (10 April 2019)
4. Final submission (10 April 2019)
5. LoA with Fully Accepted (25 April 2019)

Telkom University

SUYANTO SUYANTO <suyanto@telkomuniversity.ac.id>

IJST - Submission Confirmation

1 message

International Journal of Speech Technology (IJST) <em@editorialmanager.com> Mon, Dec 31, 2018 at 2:37 PM
Reply-To: "International Journal of Speech Technology (IJST)" <ramya.thulasingam@springer.com>
To: Suyanto Suyanto <suyanto@telkomuniversity.ac.id>

Dear Dr. Suyanto,

Thank you for submitting your manuscript, Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion, to International Journal of Speech Technology.

During the review process, you can keep track of the status of your manuscript by accessing the Editorial Manager website.

Your username is: suyanto
If you forgot your password, you can click the 'Send Login Details' link on the EM Login page at https://ijst.editorialmanager.com/.

Should you require any further assistance please feel free to e-mail the Editorial Office by clicking on "Contact Us" in the menu bar at the top of the screen.

With kind regards,
Springer Journals Editorial Office
International Journal of Speech Technology

Now that your article will undergo the editorial and peer review process, it is the right time to think about publishing your article as open access. With open access your article will become freely available to anyone worldwide and you will easily comply with open access mandates. Springer's open access offering for this journal is called Open Choice (find more information on www.springer.com/openchoice). Once your article is accepted, you will be offered the option to publish through open access. So you might want to talk to your institution and funder now to see how payment could be organized; for an overview of available open access funding please go to www.springer.com/oafunding.
Although for now you don't have to do anything, we would like to let you know about your upcoming options.

Recipients of this email are registered users within the Editorial Manager database for this journal. We will keep your information on file to use in the process of submitting, evaluating and publishing a manuscript. For more information on how we use your personal details please see our privacy policy at https://www.springernature.com/production-privacy-policy. If you no longer wish to receive messages from this journal or you have questions regarding database management, please email our publication office, stating the journal name(s) and your email address(es): PublicationOfficeSPS@springernature.com

_____
In compliance with data protection regulations, please contact the publication office if you would like to have your personal information removed from the database.

International Journal of Speech Technology

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

--Manuscript Draft--

Manuscript Number:	IJST-D-18-00237
Full Title:	Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion
Article Type:	Manuscript
Keywords:	Bahasa Indonesia; grapheme-to-phoneme conversion; syllabification points; nearest neighbour; probabilistic-based approach
Corresponding Author:	Suyanto Suyanto, Dr. Telkom University Bandung, Jawa Barat INDONESIA
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	Telkom University
Corresponding Author's Secondary Institution:	
First Author:	Suyanto Suyanto, Dr.
First Author Secondary Information:	
Order of Authors:	Suyanto Suyanto, Dr.
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	A Grapheme-to-Phoneme (G2P) conversion is an important task in the field of natural language processing. It is generally developed using a probabilistic-based data-driven approach and directly applied to a sequence of graphemes with no other information. Important research shows that incorporating syllabification point is capable of improving a probabilistic-based English G2P. However, the information should be accurately provided by a perfect orthographic syllabification. Some noises or errors of syllabification significantly reduce the G2P performance. In this paper, incorporation of syllabification points into a probabilistic-based G2P model for Bahasa Indonesia is investigated. This information is important since Bahasa Indonesia is richer than English in terms of syllables. A 5-fold cross-validating on 50 k words shows that the incorporation of syllabification points significantly improves the performance of G2P model, where the phoneme error rate (PER) is relatively reduced by 9.67%. An important contribution of this research is that the proposed G2P model is quite robust to the syllabification error. A syllable error rate (SER) of 2.5% that comes from an orthographic syllabification model just slightly increases the PER of the proposed G2P model from 0.84% to be 0.92%. A higher SER up to 10% just increase its PER to be 1.17%.

Title Page containing all the Author Details

IJST manuscript No.
(will be inserted by the editor)

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

Suyanto Suyanto

Received: date / Accepted: date

This work is supported by Telkom University

Suyanto Suyanto
School of Computing, Telkom University, Bandung, West Java 40257, Indonesia
Tel.: +62 22 7564108, Mobile: +62 812 845 12345
E-mail: suyanto@telkomuniversity.ac.id

IJST manuscript No.
(will be inserted by the editor)

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

Received: date / Accepted: date

Abstract A Grapheme-to-Phoneme (G2P) conversion is an important task in the field of natural language processing. It is generally developed using a probabilistic-based data-driven approach and directly applied to a sequence of graphemes with no other information. Important research shows that incorporating syllabification point is capable of improving a probabilistic-based English G2P. However, the information should be accurately provided by a perfect orthographic syllabification. Some noises or errors of syllabification significantly reduce the G2P performance. In this paper, incorporation of syllabification points into a probabilistic-based G2P model for Bahasa Indonesia is investigated. This information is important since Bahasa Indonesia is richer than English in terms of syllables. A 5-fold cross-validating on 50 k words shows that the incorporation of syllabification points significantly improves the performance of G2P model, where the phoneme error rate (PER) is relatively reduced by 9.67%. An important contribution of this research is that the proposed G2P model is quite robust to the syllabification error. A syllable error rate (SER) of 2.5% that comes from an orthographic syllabification model just slightly increases the PER of the proposed G2P model from 0.84% to be 0.92%. A higher SER up to 10% just increase its PER to be 1.17%.

Keywords Bahasa Indonesia · grapheme-to-phoneme conversion · syllabification points · nearest neighbour · probabilistic-based approach

1 Introduction

A G2P is widely used in many natural language processing-based systems, such as Text-to-Speech (TTS), automatic speech recognition (ASR), computer-assisted language learning (CALL), spoken term detection (STD), spoken document retrieval (SDR), and speech-to-speech machine translation (S2SMT).

Address(es) of author(s) should be given

It can be developed using three different approaches: rule-based, probabilistic-based, and neural-based. The rule-based paradigm is commonly used for a specific low complexity language but the probabilistic-based approach is widely adopted for many high complexity languages. Meanwhile, the neural-based approach is now actively developed for some very high complexity languages since it is promising to handle the out-of-vocabulary (OOV) words. However, the probabilistic-based G2P is still interesting to be used because of its simplicity and flexibility in implementation.

In the 1990s, a G2P is commonly developed using a probabilistic-based approach. This approach generally uses machine learning techniques, such as Instance-Based Learning (IBL) [5], Decision Tree Learning (DTL) [3], Hidden Markov Model (HMM) [15], Pronunciation by Analogy (PbA) [10], and Support Vector Machines (SVM) [4]. In general, these techniques have some disadvantages and give a high PER for varying datasets of languages.

In the 2010s, many researchers propose some advanced probabilistic-based techniques, such as Conditional Random Fields (CRF) [24], Kullback-Leibler divergence-based HMM (KL-HMM) [18], Unsupervised Joint Estimation (UJE) [23], Weighted Finite State Transducer (WFST) [19], and Bayesian Joint-Sequence Models (BJSM) [1]. These techniques give a lower PER for varying languages. Unfortunately, they need an aligned training dataset.

Since 2016, some researchers propose a neural-based model that does not need any aligned training data. For example, an attention-enabled encoder-decoder model in [22] is claimed to give a performance that is comparable to that of the conventional models trained using an aligned dataset. The other examples are Recurrent Neural Networks (RNN) [17] [8], Deep Bidirectional Long Short-Term Memory (DBLSTM) [13], and Multitask Sequence-to-Sequence (Seq2Seq) model [11], [16], [25]. These neural-based models are now actively explored to be applied for low-resource languages and to handle OOV words. Unfortunately, these models have high computational complexity. The other researchers also propose another G2P model that can be used more generally for almost any language (usually called a language-independent G2P), instead of a specific language, as described in [7], [12]. However, a language-independent G2P is too hard to be developed for varying languages, from the simplest to the most complex, since each language has unique characteristics and rules.

Therefore, some researchers focus on developing a specific G2P for a certain language. For instance, the researchers in [20] develop an Indonesian G2P using an instance-based learning approach called a pseudo nearest neighbour rule (PNNR) and exploiting some phonotactics knowledge. This model gives a quite low average PER of 0.93% based on a 5-fold cross validation for a dataset of 50 k words. This result is achieved by using a grapheme encoding called modified partial orthogonal binary encoding (MPOBE) [20] that makes the distance of two intraclass patterns lower and two interclass patterns higher so that they are easily classified by an instance-based classifier that works locally on some neighbours of patterns. But, the MPOBE has a limitation for the words those contain a grapheme ⟨e⟩. As described in [20], the grapheme ⟨e⟩

contributes most errors, up to 82%. This problem is caused by four prefixes 'ber', 'me', 'per', and 'ter' those produce many derivatives with ambiguous conversions to some roots (basic words) as described in Table 1.

Table 1 Some similar Indonesian formal words with different pronunciations those are mostly come from the derivatives with four prefixes 'ber', 'me', 'per', and 'ter'

Roots (basic words)	Derivatives with prefixes 'ber', 'me', 'per', and 'ter'
'*berang*' /bɛraŋ/ (irascible)	'*berangin*' /bəraŋin/ (windy)
'*merek*' /mɛrək/ (brand)	'*mereka*' /mərɛkɑ/ (they)
'*perak*' /pɛrɑk/ (silver)	'*peraka*' /pərɑkɑ/ (space on a ship deck)
'*tering*' /tɛriŋ/ (tuberculosis)	'*teringat*' /təriŋɑt/ (be reminded)

This problem probably can be solved by incorporating the syllabification boundary or point into the G2P model. Including a syllabification point into a pattern makes the distance of two interclass patterns higher. This idea is inspired by the important researches in [10] and [4], which proves that incorporating information of syllabification boundary is capable of improving the English G2P. But, the information should be accurately provided by a perfect orthographic syllabification. If there are some noises or errors in the syllabification boundary, the English G2P does not work. Few errors can be quite harmful to the English G2P [10]. In this research, the effect of incorporating the syllabification points to a G2P is investigated using a PNNR-based model for Bahasa Indonesia. This information is very important since Bahasa Indonesia is a syllable-rich language. In [21], a study on KBBI of 50 k words shows that it has 98.30% polysyllabic and only 1.70% monosyllabic words, where on average it has 3.20 syllables per word. In contrast, a study on Wordsmyth dictionary of 50 k words in [9], English has 80% polysyllabic and 20% monosyllabic words, where on average it has only 2.46 syllables per word.

2 Research Method

The block diagram of the proposed G2P with incorporating the syllabification points is illustrated by Fig. 1. Suppose the input is a grapheme sequence ⟨be.rang⟩ (irascible). First, the grapheme sequence is preprocessed to convert it into some patterns, where /*/ is a blank symbol or there is no phoneme. In this model, a syllabification point is included in the patterns. Next, the phonemic rules designed based on [2] and [6] filter one or more potential phonemes to be selected by the PNNR-based classifier. Finally, the PNNR decides the best phoneme as the conversion of the focused-grapheme in the pattern.

The dataset used in this research is a pair of one-to-one aligned grapheme-phoneme sequences by including the syllabification points. It consists of 50k words selected from *Kamus Besar Bahasa Indonesia* (KBBI), the official Indonesian great dictionary created by Pusat Bahasa (Language Center) of the Indonesian Ministry of Education and Culture. The syllabification rules that represent the major pronunciations in Indonesian described in [2] are used

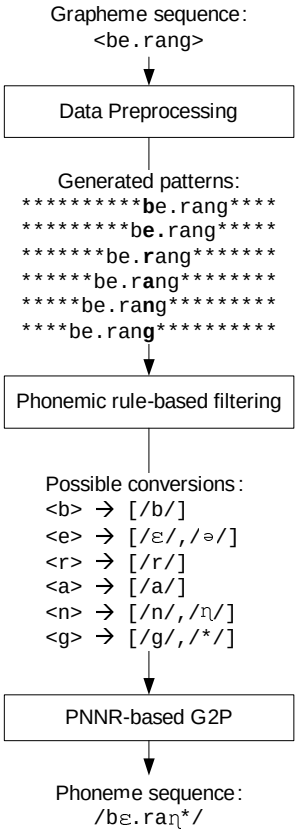

Fig. 1 Block diagram of the PNNR-based G2P with incorporating the syllabification points

to define reference syllabifications. A pair of grapheme-phoneme sequences is converted in the same way as described in [20], but in this research the syllabification points are incorporated into the patterns and a higher contextual length L of 20 (each 10 graphemes on the left and right) is used, instead of 14 (each 7 graphemes on the left and right) as implemented in the G2P without incorporating syllabification points in [20]. The longer L is used here since Bahasa Indonesia has 3.20 syllables per word on average [21]. In other words, there are three syllabification points should be added into the contextual graphemes to decide the pronunciation so that the contextual length should be $7+3=10$ graphemes on the left and right. The one-to-one alignment is illustrated by Fig 2, where /*/ is a blank symbol or there is no phoneme.

Data preprocessing. Each grapheme is firstly mapped into a single phonemic symbol so that a grapheme sequence (including the syllabification points) is one-to-one aligned to the phoneme sequence. For instance, a grapheme sequence ⟨be.rang⟩ (irascible) is aligned to a phoneme sequence /bɛ.rɑŋ/ as

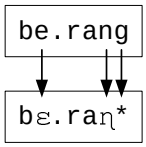

Fig. 2 One-to-one alignment of grapheme into phoneme, where /*/ is a blank symbol or there is no phoneme

illustrated by Fig. 2. Each grapheme in the sequence is then consecutively located as a focused-grapheme and the rests on their appropriate contextual positions using $L = 20$ (each 10 graphemes on the left and right), which is illustrated by Fig. 3 with ⟨*⟩ is a symbol for there is no grapheme.

L_{10} L_9 L_8 L_7 L_6 L_5 L_4 L_3 L_2 L_1 F R_1 R_2 R_3 R_4 R_5 R_6 R_7 R_8 R_9 R_{10}	Class
**********be.rang****	b
*********be.rang*****	ɛ
*******be.rang*******	r
******be.rang********	a
*****be.rang*********	ŋ
****be.rang**********	*

Fig. 3 Converting a sequence of grapheme ⟨be.rang⟩ into six patterns and classes using a contextual length $L = 20$, where F is the focus grapheme, L_i is the ith left contextual grapheme, R_i is the ith right contextual grapheme, and ⟨*⟩ is a symbol for no grapheme

Grapheme encoding The grapheme encoding used in this research is MPOBE as described in [20]. But, a new category containing a symbol of syllabification point is introduced here. Thus, the new MPOBE can be summarized in Table 2 with the detail descriptions as follows:

1. All symbols occur in a grapheme sequence are divided into four main categories, i.e. (I) vowel-oriented graphemes, (II) consonant-oriented graphemes, (III) non-graphemes, and (IV) syllabification point. In Category II, there are 13 groups of consonant-oriented graphemes generated based on their pronunciation similarities in the manner and the place of articulation described in [2];
2. The distance between Category I (vowels) and Category II (consonants) is $\sqrt{6}$; This is the biggest distance since since they have a high different in a contextual word. For instance, a grapheme ⟨a⟩ followed by a vowel ⟨u⟩ in ⟨ker.bau⟩ (buffalo) should be pronounced as /aʊ/, but it should be converted into /ɑ/ when it is followed by a consonant ⟨b⟩ in ⟨bab⟩ (chapter).
3. The distance between Category I (vowels) or II (consonants) and Category III (non-graphemes) is $\sqrt{5}$ since their differences are slightly lower. For

instance, a grapheme ⟨e⟩ in ⟨be.rang⟩ (irascible) is pronounced as /ɛ/ but in ⟨be.rang-be.rang⟩ (beaver) is pronounced as /ə/ since it contains a non-graphemic symbol ⟨-⟩;

4. The distance between two graphemes in the different group but in the same category is $\sqrt{4}$. The grapheme ⟨e⟩ in ⟨be.ban⟩ (load) is converted into /ə/ but in ⟨be.bas⟩ (free) is converted into /ɛ/ since the graphemes ⟨n⟩ and ⟨s⟩ are in the different group;
5. The distance between two graphemes in the same group is $\sqrt{2}$. The grapheme ⟨n⟩ followed by either ⟨g⟩ or ⟨k⟩ is pronounced as /ŋ/, such as in ⟨bang⟩ (brother) and ⟨bank⟩ (bank);
6. The distance between Category IV (syllabification point) and all other categories is $\frac{1}{2}\sqrt{2}$. It is given the lowest distance since shifting a syllabification point produces at least two different graphemes. For example, the grapheme sequences ⟨be.rang⟩ /bɛ.rɑŋ/ (irascible) and ⟨ber.a.ngin⟩ /bər.ɑ.ŋin/ (windy) with the focused-grapheme ⟨e⟩ have two different positions of graphemes ⟨.⟩ and ⟨r⟩.

Table 2 Modified Partial Orthogonal Binary Encoding for 26 Graphemes, 3 Non-graphemes, and a syllabification point

Category	Group	Graphemes	I	II	III	IV	Intragroup
I	1	{a, e, i, o, u}	0	$\sqrt{6}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	2	{b, p}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	3	{t, d}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	4	{k, q, g}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	5	{c, j}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	6	{f, v}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	7	{s, x, z}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	8	{m}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	9	{n}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	10	{h}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	11	{r}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	12	{l}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	13	{w}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	14	{y}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
III	15	{*, -, space}	$\sqrt{5}$	$\sqrt{5}$	0	$\frac{1}{2}\sqrt{2}$	0
IV	16	{.}	$\frac{1}{2}\sqrt{2}$	$\frac{1}{2}\sqrt{2}$	$\frac{1}{2}\sqrt{2}$	0	0

Graphemic contextual weight. Pronouncing a grapheme contextually depends on surrounding graphemes on the left and right. A contextual grapheme close to the focus is more important than the longer one. This concept is modeled using an exponentially decaying function

$$w_i = p^{(L/2)-i+1}, \tag{1}$$

where w_i is the ith contextual grapheme weight, p is an exponential constant around 2.0, and L is the number of surrounding graphemes or contextual length

distributed equally to the left and right of the focused-grapheme [20]. The optimum L varies depending on the characteristics of languages. A research in [20] shows that the optimum L for Indonesian G2P is 14. Such model is adapted in this research using a longer L of 20, instead of 14.

Distance of interclass similar patterns. Based on the previously described MPOBE and the graphemic contextual weight, the distance of two interclass similar patterns can be made longer by incorporating the information of syllabification points. Fig. 4 illustrates the distance of two interclass similar patterns with incorporating the information of syllabification points (b) is much bigger than those without the information (a). In Fig. 4(a), two interclass similar patterns generated from the words '*berang*' /bɛrɑŋ/ (irascible) and '*berangin*' /bərɑŋin/ (windy) have a low distance with only two different graphemes, i.e. $distance = \sqrt{4} \times 2^6 + \sqrt{4} \times 2^5 = 192$ that is calculated using $p = 2$. In contrast, Fig. 4(b) shows that both interclass similar patterns have a much bigger distance since they have seven different graphemes, i.e. $distance = \frac{1}{2}\sqrt{2}\times 2^{10} + \frac{1}{2}\sqrt{2}\times 2^9 + \frac{1}{2}\sqrt{2}\times 2^7 + \sqrt{5}\times 2^6 + \sqrt{2}\times 2^5 + \sqrt{2}\times 2^4 + \sqrt{2}\times 2^3 = 1,398.93$, where the syllabification points shift five graphemes on the right.

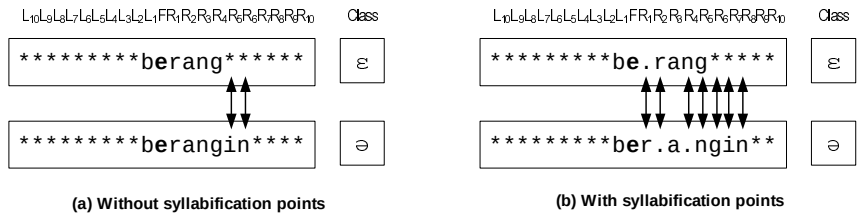

Fig. 4 The distance of two interclass similar patterns with incorporating the information of syllabification points (b) is much bigger than those without the information (a)

Pseudo nearest neighbour rule. Since this research focuses to evaluate the effectiveness of incorporating the information of syllabification points into G2P, the PNNR described in [20] is used in the same way here to easily make the comparison fair. PNNR works locally by considering only k unique patterns in the training dataset, where k is called the neighbourhood size. In [20], the researchers prove that the local scheme is better for handling some anomalies in a G2P. The PNNR uses a neighbourhood weight

$$u_j = \frac{1}{j^c}, \tag{2}$$

where u_j is the neighbourhood weight of the jth neighbour and c is a power constant around 1.0 as described in [20]. This model is directly adopted in this research, but the c will be re-optimized to develop a new optimum G2P model.

The PNNR for a G2P works by finding the minimum probabilistic nearest neighbour distance between the current pattern and all possible classes of phonemes to decide the best conversion. The total distance between the current pattern and a class of phoneme considering the k closest unique patterns is calculated using

$$T = \sum_{j=1}^{k} u_j \sum_{i=1}^{L/2} (d_{li} w_i + d_{ri} w_i) \tag{3}$$

where u_j is the weight for the j-th neighbour calculated using Equation (2), L is the contextual length of 20, d_{li} and d_{ri} are the distances of the i-th contextual grapheme on the left and right respectively calculated based on the MPOBE, and w_i is the ith contextual grapheme weight calculated using Equation (1).

3 Results and Discussion

In this research, a dataset of 50 k formal words from KBBI as described in [20] is used to evaluate the proposed G2P model using a 5-fold cross-validation scheme to make a fair comparison to the PNNR-based G2P model without incorporating syllabification points in [20]. Both models are evaluated based on a PER formulated as

$$\mathrm{PER} = \frac{E}{N}, \tag{4}$$

where E is the number of wrong phonemes and N is the total number of phonemes in the testing data.

3.1 Optimizing the parameters of the proposed model

The proposed PNNR-based G2P with incorporating syllabification points is evaluated based on the five folds cross-validation as used in [20]. To save time and resources, the three parameters of the model are sequentially optimized. Firstly, the neighbourhood size k is optimized. Next, using the optimum k, the best power constant for neighborhood weight c is then searched. Finally, the exponential constant for contextual weight p is optimized using the best k and p.

Neighbourhood size k. The neighbourhood size or the number of neighbour k in PNNR is hard to be predicted since it is varying based on the case. In this research, PNNR is firstly evaluated using $k = 1$ to 9 with a predefined $c = 1.0$ and $p = 2.0$. The experimental results using the five-fold cross-validation, as illustrated in Fig. 5, show that a small k makes PNNR produces high PER because it works too general. On the other hand, a big k also affects PNNR produces high PER since it performs too specific. The lowest PER is generally, occurred on three of the five folds, reached on $k = 5$.

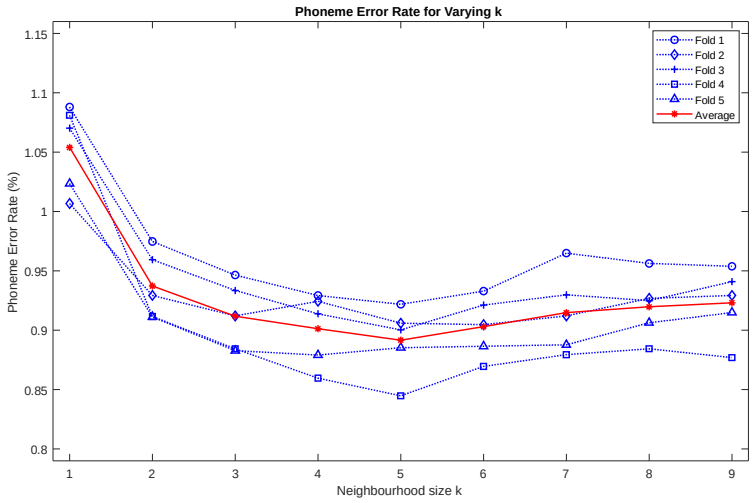

Fig. 5 PER produced by PNNR-based G2P with $c = 1.0$, $p = 2.0$, and varying k for the five folds

Power constant for neighborhood weight c. The PNNR with $k = 5$ and $p = 2.0$ is then evaluated using some power constant for neighborhood weights $c = 0.8$ to 1.6 with a step size of 0.1. The five-fold cross-validation results, illustrated by Fig. 6, show that a too small c or a too big c makes PNNR produces high PER. A small c make a close neighbour has a quite similar distance to the further one. In contrast, a big c make the closer neighbours have very high distances while the furthers have very low distances. The neighborhood weight is balanced on an optimum $c = 1.2$ for all five folds.

Exponential constant for contextual weight p. Finally, PNNR with $k = 5$ and $c = 1.2$ is examined using some exponential constants for contextual weight $p = 1.3$ to 2.1 with a step size of 0.1. The five-fold cross-validation results, illustrated by Fig. 7, show that a too small p or a too big p makes PNNR produces high PER as the contextual weights are not balanced for the closest to the furthest phonemes. The lowest PER is reached using $p = 1.5$ for all five folds that give the lowest averaged PER of 0.84%. This result is better than the G2P without incorporating the information of syllabification points that gives PER of 0.93%. It means that incorporating the information of syllabification points is relatively reduced the PER by 9.67%.

3.2 Detailed analysis

A detailed inspection shows that the proposed G2P model is capable of solving some ambiguous conversions of derivatives caused by four prefixes 'ber', 'me', 'per', and 'ter' as described in Table 1. Unfortunately, it has a limitation to

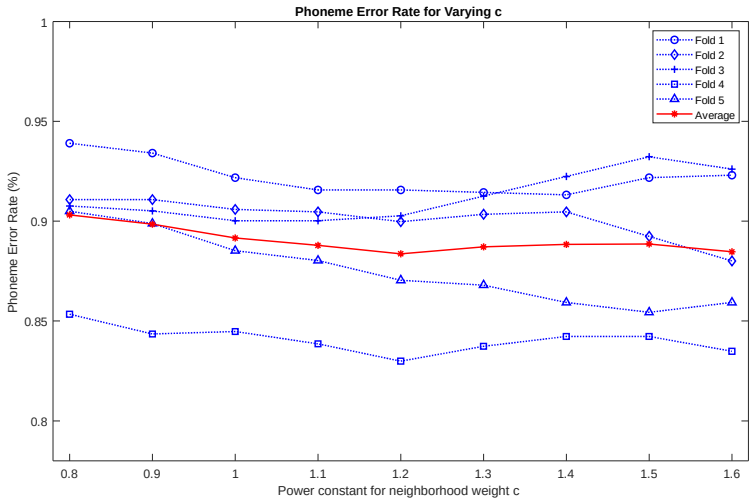

Fig. 6 PER produced by PNNR-based G2P with $c = 1.0$, $p = 2.0$, and varying k for the five folds

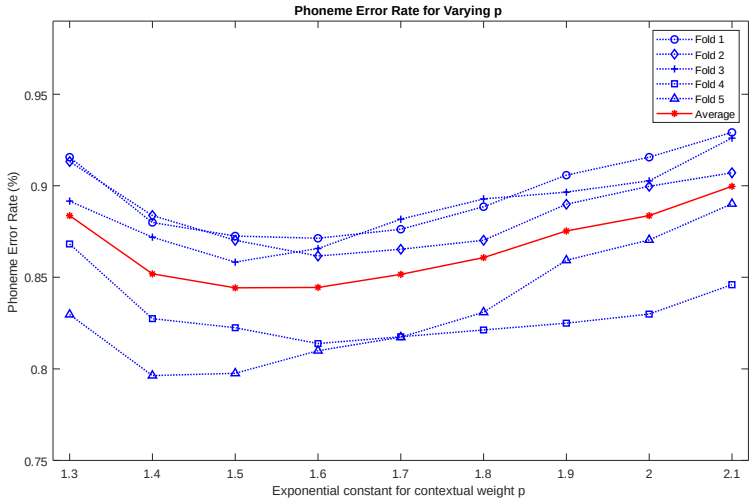

Fig. 7 PER produced by PNNR-based graphemic syllabification with $k = 5$, $c = 1.2$, and varying p for the five folds

solve some similar compound words containing a grapheme ⟨e⟩ that is dynamically pronounced as either /ɛ/ or /ə/, e.g. a word 'reses' (recess) is pronounced as /rəsɛs/ but 'resesi' (recession) is pronounced as /rɛsɛsi/. Such case is quite hard to be solved by the proposed G2P model although it incorporates the syllabification points.

This problem probably is caused by the blank contextual graphemes in the patterns. In the cases of 'reses' and 'resesi', the generated patterns with the contextual length of 20 are mostly dominated by ⟨*⟩. This fact raises a bias in the calculation of the total distance between the current pattern and the possible classes of phonemes. A possible solution to this problem is taking into account other words on the left or right in a sentence-level context since the cross-word patterns can reduce the calculation bias.

3.3 Robustness of the proposed G2P model

The proposed PNNR-based G2P that incorporates syllabification points with the optimum parameters of $k = 5$, $p = 1.2$, and $p = 1.5$ is finally evaluated using four other datasets by including the syllabification noise levels of 2.5%, 5.0%, 7.5%, and 10.0%. The syllabification noise is randomly generated by shifting a position of true syllabification point into either left or right. For instances, a true syllabification point in a word 'ber.a.ngin' is randomly shifted into the left to be 'be.ra.ngin'. The five-fold cross-validation results illustrated by Fig. 8 show that the PNNR is quite robust to the noise. A syllabification noise of 2.5% just slightly increases the PER of the proposed G2P model from 0.84% to be 0.92%. A much higher syllabification noise up to 10% just increases the PER to be 1.17%. Therefore, a graphemic syllabification with an SER of around 2.5% as described in [14] does not increase the PER significantly. The graphemic syllabification for name entities with an SER of around 7.5% as described in [14] just slightly increases the PER. So, this proposed G2P model is very promising to be developed to handle the pronunciation of Indonesian formal words as well as name entities.

The achieved high noise-robustness can be easily explained using a simple illustration in Fig. 9. Using $p = 2.0$, the distance of two interclass patterns with a noise of syllabification is $distance = \frac{1}{2}\sqrt{2} \times 2^7 + \sqrt{5} \times 2^6 + \sqrt{2} \times 2^5 + \sqrt{2} \times 2^4 + \sqrt{2} \times 2^3 = 312.81$ (b), which is lower than that of two interclass patterns without noise (i.e 1,398.93) but bigger than that of two interclass patterns without incorporating the syllabification points (only 192).

4 Conclusion

The three parameters of the proposed G2P model are easily tuned using a sequentially optimization. Using the optimum values of those parameters and the perfect information of syllabification points, the proposed G2P model gives a lower PER (0.84%) than the G2P without incorporating the information of syllabification points (0.93%). It successfully solves some ambiguous conversions of derivatives caused by four prefixes 'ber', 'me', 'per', and 'ter'. But, it fails to solve some similar compound words containing a grapheme ⟨e⟩ that is dynamically pronounced as either /ɛ/ or /ə/, such as a word 'reses' (recess) is

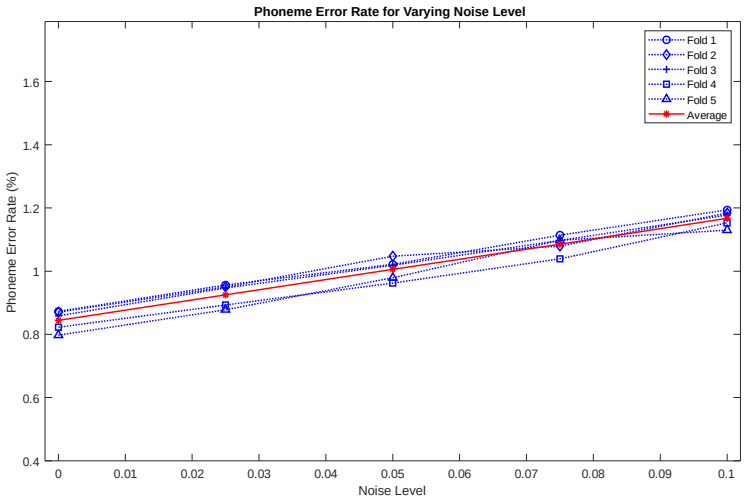

Fig. 8 Effects of noise levels to the PNNR-based G2P with incorporating syllabification points

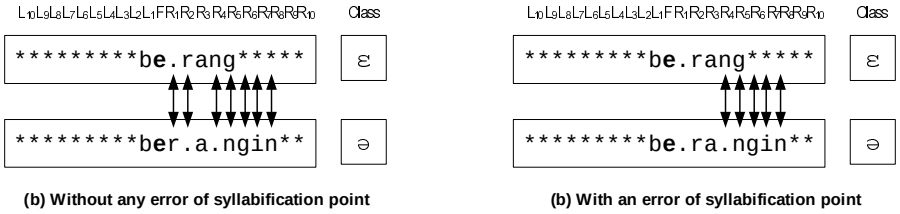

Fig. 9 A shifted syllabification point slightly reduces the distance of two interclass patterns

pronounced as /rəsɛs/ but 'resesi' (recession) is pronounced as /rɛsɛsi/. Another important achievement of this research is that the proposed G2P model is robust to the syllabification noise. The SER of 2.5% just slightly increase the PER of the proposed G2P model from 0.84% to be 0.92%. A higher SER of 10% just increase its PER to be 1.17%. Hence, this model is very promising to handle the pronunciation of Indonesian formal words. In the future, it can be evaluated to pronounce the name entity dataset.

Acknowledgements We would like to thank Telkom University for great support.

References

1. Aachen, R.: Bayesian joint-sequence models for grapheme-to-phoneme conversion. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2836–2840 (2017). DOI 10.1109/ICASSP.2017.7952674
2. Alwi, H., Dardjowidjojo, S., Lapoliwa, H., Moeliono, A.M.: Tata Bahasa Baku Bahasa Indonesia (The Standard Indonesian Grammar), 3 edn. Balai Pustaka, Jakarta (1998)

3. Andersen, O., Dalsgaard, P.: Multi-lingual testing of a self-learning approach to phonemic transcription of orthography. In: EUROSPEECH, pp. 1117–1120 (1995)
4. Bartlett, S., Kondrak, G., Cherry, C.: Automatic syllabification with structured SVMs for letter-to-phoneme conversion. In: Proceedings of Human Language Technologies: The 2008 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 568–576. Columbus, Ohio (2008)
5. Bosch, A.V.D., Daelemans, W.: Data-oriented methods for grapheme-to-phoneme conversion. In: The sixth conference on European chapter of the Association for Computational Linguistics (EACL), pp. 45–53. Association for Computational Linguistics, Morristown, NJ, USA (1993). DOI 10.3115/976744.976751. URL http://portal.acm.org/citation.cfm?doid=976744.976751
6. Chaer, A.: Fonologi Bahasa Indonesia (Indonesian Phonology). Rineka Cipta, Jakarta (2009)
7. Deri, A., Knight, K.: Grapheme-to-Phoneme Models for (Almost) Any Language. In: The 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 399–408 (2016)
8. Jyothi, P., Hasegawa-Johnson, M.: Low-Resource Grapheme-to-Phoneme Conversion Using Recurrent Neural Networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2017)
9. Marchand, Y., Adsett, C.R., Damper, R.I.: Automatic syllabification in English: a comparison of different algorithms. Language and speech **52**(Pt 1), 1–27 (2009). DOI 10.1177/0023830908099881
10. Marchand, Y., Damper, R.I.: Can syllabification improve pronunciation by analogy of English? Natural Language Engineering **13**(01), 1 (2006). DOI 10.1017/S1351324905004043. URL http://www.journals.cambridge.org/abstract_S1351324905004043
11. Milde, B., Schmidt, C., Joachim, K., Augustin, S., Milde, B., Schmidt, C.A.: Multi-task Sequence-to-Sequence Models for Grapheme-to-Phoneme Conversion. In: INTERSPEECH, pp. 2536–2540 (2017). DOI 10.21437
12. Mohr, M.: Asking for Help: Grapheme-to-Phoneme Conversion using Active Learning at Application Time. Ph.D. thesis, Hamburg University (2017)
13. Mousa, A.E.d.: Deep Bidirectional Long Short-Term Memory Recurrent Neural Networks for Grapheme-to-Phoneme Conversion utilizing Complex Many-to-Many Alignments. In: Interspeech, pp. 2836–2840 (2016)
14. Parande, E.A., Suyanto, S.: Indonesian graphemic syllabification using a nearest neighbour classifier and recovery procedure. International Journal of Speech Technology (2018). DOI 10.1007/s10772-018-09569-3. URL https://doi.org/10.1007/s10772-018-09569-3
15. Parfitt, S.H., Sharman, R.A.: A bi-directional model of English pronunciation. In: EUROSPEECH, pp. 801–804 (1991)
16. Peters, B.: Massively Multilingual Neural Grapheme-to-Phoneme Conversion (2017)
17. Rao, K., Peng, F., Sak, H., Beaufays, F.: Grapheme-to-Phoneme Conversion Using Long Short-Term Memory Recurrent Neural Networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2015)
18. Razavi, M., Rasipuram, R., Magimai, M.: Acoustic Data-Driven Grapheme-to-Phoneme Conversion in the Probabilistic Lexical Modeling Framework. Speech Communication **80**, 1–21 (2016)
19. Soky, K., Lu, X., Shen, P., Vanna, C., Kato, H.: Building WFST based Grapheme to Phoneme Conversion for Khmer. In: Khmer Natural Language Processing (2016)
20. Suyanto, S., Hartati, S., Harjoko, A.: Modified Grapheme Encoding and Phonemic Rule to Improve PNNR-Based Indonesian G2P. International Journal of Advanced Computer Science and Applications (IJACSA) **7**(3), 430–435 (2016)
21. Suyanto, S., Hartati, S., Harjoko, A., Compernolle, D.V.: Indonesian syllabification using a pseudo nearest neighbour rule and phonotactic knowledge. Speech Communication **85**, 109–118 (2016). DOI 10.1016/j.specom.2016.10.009. URL http://dx.doi.org/10.1016/j.specom.2016.10.009
22. Toshniwal, S., Livescu, K.: Read, Attend and Pronounce: An Attention-Based Approach for Grapheme-To-Phoneme Conversion (2016)

23. Tsujioka, S., Sakti, S., Yoshino, K., Neubig, G., Nakamura, S.: Unsupervised Joint Estimation of Grapheme-to-phoneme Conversion Systems and Acoustic Model Adaptation for Non-native Speech Recognition. In: INTERSPEECH, 1 (2016). DOI 10.21437/Interspeech.2016-919
24. Wang, D., King, S.: Letter-to-sound Pronunciation Prediction Using Conditional Random Fields. IEEE Signal Processing Letters **18**(2), 122–125 (2011)
25. Yao, K., Zweig, G.: Sequence-to-Sequence Neural Net Models for Grapheme-to-Phoneme Conversion (2015)

Evidences of correspondences

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

1. First submission (31 December 2018)
2. **LoA with Major Revision (20 March 2019)**
3. Respond to Reviewers (10 April 2019)
4. Final submission (10 April 2019)
5. LoA with Fully Accepted (25 April 2019)

Telkom University

SUYANTO SUYANTO <suyanto@telkomuniversity.ac.id>

Decision on your manuscript #IJST-D-18-00237

1 message

International Journal of Speech Technology (IJST) <em@editorialmanager.com> Wed, Mar 20, 2019 at 3:03 AM
Reply-To: "International Journal of Speech Technology (IJST)" <ramya.thulasingam@springer.com>
To: Suyanto Suyanto <suyanto@telkomuniversity.ac.id>

Dear Dr. Suyanto:

We have received the reports from our advisors on your manuscript, "Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion", which you submitted to International Journal of Speech Technology.

Based on the advice received, the Editor feels that your manuscript could be reconsidered for publication should you be prepared to incorporate major revisions. When preparing your revised manuscript, you are asked to carefully consider the reviewer comments which are attached, and submit a list of responses to the comments. Your list of responses should be uploaded as a file in addition to your revised manuscript.

In order to submit your revised manuscript electronically, please access the Editorial Manager website.

Your username is: suyanto
If you forgot your password, you can click the 'Send Login Details' link on the EM Login page at https://www.editorialmanager.com/ijst/.

Please click "Author Login" to submit your revision.

We look forward to receiving your revised manuscript.

Best regards,
Amy Neustein, Ph.D.
Editor-in-Chief
International Journal of Speech Technology

COMMENTS FOR THE AUTHOR:

Reviewer #1: The research is original and well organized, but It would be better if the following points are taken into consideration:
1. page 3 lines 34 and 35. a reference needed
2. I think the model should be tested on different contextual lengths to come up with the best length
3.in Fig 1 data preprocessing and phonetic rule-based filtering need more illustration
4. page 5 line 47. "since" is repeated
5. the distance between categories may be not clear and need more illustration specially for the non specialists
6. I think it is better to include an illustrative example that shows all steps
7. the authors are advised to refer to modern similar work as:

Abu-Soud S. "ILATalk: A New Multilingual Text-To-Speech Synthesizer with Machine Learning", International Journal of Speech Technology, Volume 19, Issue 1, pp 55-64, ISSN 1381-2416, March 2016.

8. finally this model should be compared with similar work

how we use your personal details please see our privacy policy at https://www.springernature.com/production-privacy-policy. If you no longer wish to receive messages from this journal or you have questions regarding database management, please contact the Publication Office at the link below.

_____

In compliance with data protection regulations, you may request that we remove your personal registration details at any time. (Use the following URL: https://www.editorialmanager.com/ijst/login.asp?a=r) Please contact the publication office if you have any questions.

Evidences of correspondences

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

1. First submission (31 December 2018)
2. LoA with Major Revision (20 March 2019)
3. **Respond to Reviewers (10 April 2019)**
4. Final submission (10 April 2019)
5. LoA with Fully Accepted (25 April 2019)

Author's Response To Reviewer Comments

Close

Reviewer #1: The research is original and well organized, but It would be better if the following points are taken into consideration:

1. Page 3 lines 34 and 35. a reference needed
>> The reference is added: “In contrast, a study on the Wordsmyth dictionary of 50 k words described in [10] shows that English has much more monosyllabic words up to 20% and the number of the polysyllabic ones is 80%, where a further investigation shows that it has only 2.46 syllables per word.”

2. I think the model should be tested on different contextual lengths to come up with the best length.
>> An additional experiment is performed to examine some contextual lengths as illustrated by Fig. 10. An additional paragraph discussing the impact of the contextual length as well as the optimum contextual length for the proposed model is also added.

3. In Fig 1 data preprocessing and phonetic rule-based filtering need more illustration
>> An additional illustration of data preprocessing is shown in Fig. 4 and a table of phonetic rule-based filtering is also added in Table 2.

4. Page 5 line 47. "since" is repeated
>> The repeated “since” in the sentence is removed to be: “This is the biggest distance since they have the highest contextual difference in a word.”

5. The distance between categories may be not clear and need more illustration specially for the non-specialists
>> An additional illustration is now added in Table 4.

6. I think it is better to include an illustrative example that shows all steps
>> The illustrative example is now added in Fig. 6.

7. The authors are advised to refer to modern similar work as:

Abu-Soud S. "ILATalk: A New Multilingual Text-To-Speech Synthesizer with Machine Learning", International Journal of Speech Technology, Volume 19, Issue 1, pp 55-64, ISSN 1381-2416, March 2016.

>> The paper is now referred.

8. Finally this model should be compared with similar work
>> The model is compared to ILA, illustrated in Table 5 and Table 6.

In order to make the title clearer, it is slightly modified by inserting “a Model of” to be “Incorporating Syllabification Points into a Model of Grapheme-to-Phoneme Conversion”.

Close

Evidences of correspondences

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

1. First submission (31 December 2018)
2. LoA with Major Revision (20 March 2019)
3. Respond to Reviewers (10 April 2019)
4. **Final submission (10 April 2019)**
5. LoA with Fully Accepted (25 April 2019)

Telkom University

SUYANTO SUYANTO <suyanto@telkomuniversity.ac.id>

IJST - Submission Confirmation

1 message

International Journal of Speech Technology (IJST) <em@editorialmanager.com> Wed, Apr 10, 2019 at 4:39 PM
Reply-To: "International Journal of Speech Technology (IJST)" <ramya.thulasingam@springer.com>
To: Suyanto Suyanto <suyanto@telkomuniversity.ac.id>

Dear Dr. Suyanto,

Thank you for submitting your manuscript, Flipping Onsets to Enhances Syllabification, to International Journal of Speech Technology.

During the review process, you can keep track of the status of your manuscript by accessing the Editorial Manager website.

Your username is: suyanto
If you forgot your password, you can click the 'Send Login Details' link on the EM Login page at https://www.editorialmanager.com/ijst/.

Should you require any further assistance please feel free to e-mail the Editorial Office by clicking on "Contact Us" in the menu bar at the top of the screen.

With kind regards,
Springer Journals Editorial Office
International Journal of Speech Technology

Now that your article will undergo the editorial and peer review process, it is the right time to think about publishing your article as open access. With open access your article will become freely available to anyone worldwide and you will easily comply with open access mandates. Springer's open access offering for this journal is called Open Choice (find more information on www.springer.com/openchoice). Once your article is accepted, you will be offered the option to publish through open access. So you might want to talk to your institution and funder now to see how payment could be organized; for an overview of available open access funding please go to www.springer.com/oafunding.
Although for now you don't have to do anything, we would like to let you know about your upcoming options.

Recipients of this email are registered users within the Editorial Manager database for this journal. We will keep your information on file to use in the process of submitting, evaluating and publishing a manuscript. For more information on how we use your personal details please see our privacy policy at https://www.springernature.com/production-privacy-policy. If you no longer wish to receive messages from this journal or you have questions regarding database management, please contact the Publication Office at the link below.

In compliance with data protection regulations, you may request that we remove your personal registration details at any time. (Use the following URL: https://www.editorialmanager.com/ijst/login.asp?a=r) Please contact the publication office if you have any questions.

International Journal of Speech Technology

Incorporating Syllabification Points into a Model of Grapheme-to-Phoneme Conversion

--Manuscript Draft--

Manuscript Number:	IJST-D-18-00237R1
Full Title:	Incorporating Syllabification Points into a Model of Grapheme-to-Phoneme Conversion
Article Type:	Manuscript
Keywords:	Bahasa Indonesia; grapheme-to-phoneme conversion; syllabification points; nearest neighbour; probabilistic-based approach
Corresponding Author:	Suyanto Suyanto, Dr. Telkom University Bandung, Jawa Barat INDONESIA
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	Telkom University
Corresponding Author's Secondary Institution:	
First Author:	Suyanto Suyanto, Dr.
First Author Secondary Information:	
Order of Authors:	Suyanto Suyanto, Dr.
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	A model to convert grapheme-to-phoneme (G2P) is important in the field of natural language processing (NLP). It is generally developed using a probabilistic-based data-driven approach and directly applied to a sequence of graphemes with no other information. Important research shows that incorporating information of syllabification point is capable of improving a probabilistic-based English G2P. However, the information should be accurately provided by a perfect orthographic syllabification. Some noises or errors of syllabification significantly reduce the G2P performance. In this paper, incorporation of syllabification points into a probabilistic-based G2P model for Bahasa Indonesia is investigated. This information is important since Bahasa Indonesia is richer than English in terms of syllables. A 5-fold cross-validating on 50 k words shows that the incorporation of syllabification points significantly improves the performance of G2P model, where the phoneme error rate (PER) can be relatively reduced by 10.75%. This PER is much lower than the G2P model based on an inductive learning algorithm (ILA). An important contribution of this research is that the proposed G2P model is quite robust to syllabification errors. A syllable error rate (SER) of 2.5% that comes from an orthographic syllabification model just slightly increases the PER of the proposed G2P model from 0.83% to be 0.90%. A higher SER up to 10% just increase the PER to be 1.14%.

Reviewer #1: The research is original and well organized, but It would be better if the following points are taken into consideration:

1. Page 3 lines 34 and 35. a reference needed
>> The reference is added: “In contrast, a study on the Wordsmyth dictionary of 50 k words described in [10] shows that English has much more monosyllabic words up to 20% and the number of the polysyllabic ones is 80%, where a further investigation shows that it has only 2.46 syllables per word.”

2. I think the model should be tested on different contextual lengths to come up with the best length.
>> An additional experiment is performed to examine some contextual lengths as illustrated by Fig. 10. An additional paragraph discussing the impact of the contextual length as well as the optimum contextual length for the proposed model is also added.

3. In Fig 1 data preprocessing and phonetic rule-based filtering need more illustration
>> An additional illustration of data preprocessing is shown in Fig. 4 and a table of phonetic rule-based filtering is also added in Table 2.

4. Page 5 line 47. "since" is repeated
>> The repeated “since” in the sentence is removed to be: “This is the biggest distance **since** they have the highest contextual difference in a word.”

5. The distance between categories may be not clear and need more illustration specially for the non-specialists
>> An additional illustration is now added in Table 4.

6. I think it is better to include an illustrative example that shows all steps
>> The illustrative example is now added in Fig. 6.

7. The authors are advised to refer to modern similar work as:

Abu-Soud S. "ILATalk: A New Multilingual Text-To-Speech Synthesizer with Machine Learning", International Journal of Speech Technology, Volume 19, Issue 1, pp 55-64, ISSN 1381-2416, March 2016.

>> The paper is now referred.

8. Finally this model should be compared with similar work
>> The model is compared to ILA, illustrated in Table 5 and Table 6.

In order to make the title clearer, it is slightly modified by inserting “a Model of” to be “Incorporating Syllabification Points into a Model of Grapheme-to-Phoneme Conversion”.

Title Page containing all the Author Details

IJST manuscript No.
(will be inserted by the editor)

Incorporating Syllabification Points into a Model of Grapheme-to-Phoneme Conversion

Suyanto Suyanto

Suyanto Suyanto
School of Computing, Telkom University, Bandung, West Java 40257, Indonesia
Tel.: +62 22 7564108, Mobile: +62 812 845 12345
Orcid: https://orcid.org/0000-0002-8897-8091
E-mail: suyanto@telkomuniversity.ac.id

IJST manuscript No.
(will be inserted by the editor)

Incorporating Syllabification Points into a Model of Grapheme-to-Phoneme Conversion

Abstract A model to convert grapheme-to-phoneme (G2P) is important in the field of natural language processing (NLP). It is generally developed using a probabilistic-based data-driven approach and directly applied to a sequence of graphemes with no other information. Important research shows that incorporating information of syllabification point is capable of improving a probabilistic-based English G2P. However, the information should be accurately provided by a perfect orthographic syllabification. Some noises or errors of syllabification significantly reduce the G2P performance. In this paper, incorporation of syllabification points into a probabilistic-based G2P model for Bahasa Indonesia is investigated. This information is important since Bahasa Indonesia is richer than English in terms of syllables. A 5-fold cross-validating on 50 k words shows that the incorporation of syllabification points significantly improves the performance of G2P model, where the phoneme error rate (PER) can be relatively reduced by 10.75%. This PER is much lower than the G2P model based on an inductive learning algorithm (ILA). An important contribution of this research is that the proposed G2P model is quite robust to syllabification errors. A syllable error rate (SER) of 2.5% that comes from an orthographic syllabification model just slightly increases the PER of the proposed G2P model from 0.83% to be 0.90%. A higher SER up to 10% just increase the PER to be 1.14%.

Keywords Bahasa Indonesia · grapheme-to-phoneme conversion · syllabification points · nearest neighbour · probabilistic-based approach

1 Introduction

A G2P model is widely used in many NLP-based systems, such as speech recognition, computer-assisted language learning, spoken document retrieval, speech synthesis, speech-to-speech machine translation, etc. It can be developed using

three different approaches: rule-based, probabilistic-based, and neural-based. The rule-based paradigm is commonly used for a specific low complexity language but the probabilistic-based approach is widely adopted for many high complexity languages. Meanwhile, the neural-based approach is now actively developed for some very high complexity languages since it is promising to handle the out-of-vocabulary (OOV) words. However, the probabilistic-based G2P is still interesting to be used because of its simplicity and flexibility in implementation.

In the 1990s, a G2P is commonly developed using a probabilistic-based approach. This approach generally uses machine learning techniques, such as Instance-Based Learning (IBL) [6], Decision Tree Learning (DTL) [4], Hidden Markov Model (HMM) [16], Pronunciation by Analogy (PbA) [11], and Support Vector Machines (SVM) [5]. In general, these techniques have some disadvantages and give a high PER for varying datasets of languages.

In the 2010s, many researchers propose some advanced probabilistic-based techniques, such as Conditional Random Fields (CRF) [26], Kullback-Leibler divergence-based HMM (KL-HMM) [19], Unsupervised Joint Estimation (UJE) [25], Weighted Finite State Transducer (WFST) [20], and Bayesian Joint-Sequence Models (BJSM) [1]. These techniques give a lower PER for varying languages. Unfortunately, they need an aligned training dataset.

Since 2016, some researchers propose a neural-based model that does not need any aligned training data. For example, an attention-enabled encoder-decoder model in [24] is claimed to give a performance that is comparable to that of the conventional models trained using an aligned dataset. The other examples are Recurrent Neural Networks (RNN) [18] [9], Deep Bidirectional Long Short-Term Memory (DBLSTM) [14], and Multitask Sequence-to-Sequence (Seq2Seq) model [12], [17], [27]. These neural-based models are now actively explored to be applied for low-resource languages and to handle OOV words. Unfortunately, these models have high computational complexity. The other researchers also propose another G2P model that can be used more generally for almost any language (usually called a language-independent G2P), instead of a specific language, as described in [8], [13]. However, a language-independent G2P is too hard to be developed for varying languages, from the simplest to the most complex, since each language has unique characteristics and rules.

Therefore, some researchers focus on developing a specific G2P for a certain language. For instance, the researchers in [22] develop an Indonesian G2P using an instance-based learning approach called pseudo nearest neighbour rule (PNNR) combined with the phonemic knowledge. This model gives a quite low average PER of 0.93% for a dataset of 50 k words based on an evaluation of 5-fold cross-validation. This result is achieved using a grapheme encoding called modified partial orthogonal binary encoding (MPOBE) [22], which makes the distance of two intraclass patterns lower and two interclass patterns higher so that they are easily classified by an instance-based classifier that works locally based on some neighbours of patterns. But, the model has a limitation for the words those contain a grapheme ⟨e⟩. As described in [22],

the grapheme ⟨e⟩ dominates the PER of 82% as four prefixes '*ber*', '*me*', '*per*', and '*ter*' create many derivative words with the confused conversions to some roots, as described in Table 1.

Table 1 Some similar Indonesian formal words with different pronunciations those are mostly come from the derivative words

Root (basic word)	Derivative word
'*berang*' /bɛrɑŋ/ (irascible)	'*berangga*' /bərɑŋgɑ/ (horned)
'*merek*' /mɛrək/ (brand)	'*mereka*' /mərɛkɑ/ (they)
'*perak*' /pɛrɑk/ (silver)	'*peraka*' /pərɑkɑ/ (space on a ship deck)
'*tering*' /tɛriŋ/ (tuberculosis)	'*teringat*' /təriŋɑt/ (be reminded)

This problem probably can be solved by incorporating the syllabification boundary or point into the G2P model. Including a syllabification point into a pattern makes the distance of two interclass patterns higher. This idea is inspired by the important researches in [11] and [5], which proves that incorporating information of syllabification boundary is capable of improving the English G2P. But, the information should be accurately provided by a perfect orthographic syllabification. If there are some noises or errors in the syllabification boundary, this scheme does not improve the G2P. Few errors can be quite harmful to the English G2P [11].

In this research, the impact of incorporating the syllabification points to a G2P model is investigated using a PNNR-based model for Bahasa Indonesia. This information is very important because Bahasa Indonesia is a syllable-rich language. In [23], a study on KBBI of 50 k words shows that it has 98.30% polysyllabic and only 1.70% monosyllabic words, where on average it has 3.20 syllables per word. In contrast, a study on the Wordsmyth dictionary of 50 k words described in [10] shows that English has much more monosyllabic words up to 20% and the number of the polysyllabic ones is 80%, where a further investigation shows that it has only 2.46 syllables per word.

2 Research Method

The proposed PNNR-based G2P with incorporating syllabification points, which is called PNNR+SP, is illustrated by a block diagram in Fig. 1. Suppose the input is a grapheme sequence ⟨be.rang⟩ (irascible). First, the grapheme sequence is preprocessed to convert it into some patterns, where /*/ is a blank symbol or there is no phoneme. In this model, a syllabification point is included in the patterns. Next, the phonemic rules designed based on [3] and [7] filter one or more potential phonemes to be selected by the PNNR-based classifier. The PNNR finally chooses the best phoneme as the conversion of the center-grapheme in the pattern.

The dataset used in this research is a pair of one-to-one aligned grapheme-phoneme sequences by including the syllabification points. It consists of 50k words from an Indonesian dictionary, which is commonly known as *Kamus*

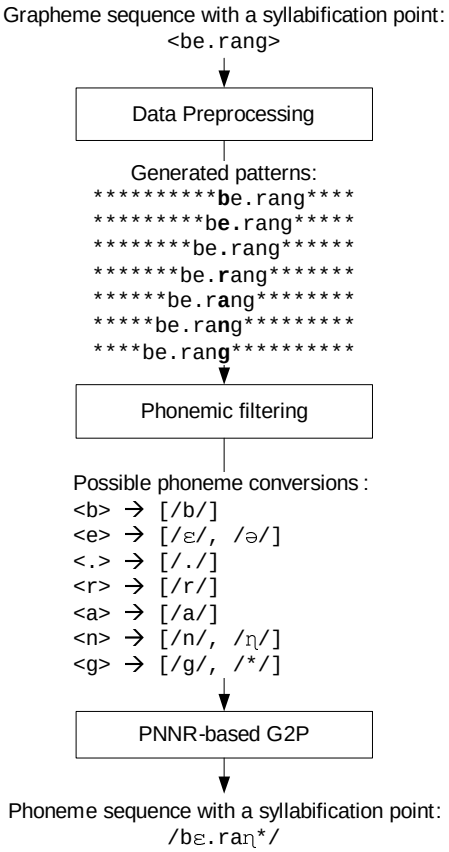

Fig. 1 The PNNR-based G2P with incorporating syllabification points (PNNR+SP)

Besar Bahasa Indonesia (KBBI), created by *Badan Pengembangan dan Pembinaan Bahasa*, *Kemdikbud* (the ministry of education and culture). Here, the rules of syllabifications representing the major pronunciations in all areas in Indonesia described in [3] are referred to be the standard syllabifications.

A pair of grapheme-phoneme sequences is converted in the same way as described in [21] and [22], but in this research the syllabification points are incorporated into the patterns. Therefore, a higher contextual length $L = 20$ (10 surrounding graphemic symbols on the left and right each) is used, instead of 14 as implemented in the G2P without incorporating syllabification points proposed in [21] and [22]. The longer L is needed here since an Indonesian word, on average, contains around 3.20 syllables [23]. In other words, there are three syllabification points should be added into the contextual graphemes to decide the pronunciation so that the contextual length should be $7+3 = 10$. The one-to-one alignment is illustrated by Fig 2, where /*/ is a blank symbol or there is no phoneme.

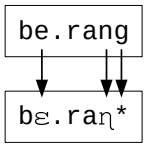

Fig. 2 One-to-one alignment of grapheme into phoneme, where /*/ is a blank symbol or there is no phoneme

2.1 Data preprocessing

Each grapheme is firstly mapped into a single phonemic symbol so that a grapheme sequence (including the syllabification points) is one-to-one aligned to the phoneme sequence. For example, a grapheme sequence with a syllabification point ⟨be.rang⟩ from a word '*berang*' (irascible) is aligned to a phoneme sequence /bɛ.rɑŋ/ as illustrated by Fig. 2.

Every grapheme in the sequence ⟨be.rang⟩ is then consecutively put on the center-grapheme and the others on its surrounding based on the desired L. For instance, using $L = 8$, the grapheme sequence is converted into seven pairs of pattern and class as illustrated by Fig. 3, where **F** is the focus or center-grapheme, L_i and R_i are the ith contextual graphemes on the left and right respectively, and ⟨*⟩ is an empty grapheme. However, a longer L is needed as Bahasa Indonesia has many long words. Fig. 4 illustrates a process of pattern generation for a longer word '*keberangkatan*' (departure) using $L = 20$.

$L_4L_3L_2L_1$**F**$R_1R_2R_3R_4$	Class
****be.ra	b
***be.ran	ɛ
**be.rang	.
be.rang	r
be.rang**	a
e.rang***	ŋ
.rang****	*

Fig. 3 Converting a sequence of grapheme ⟨*be.rang*⟩ into seven pairs of pattern and class using $L = 8$

2.2 Phonemic filtering

A set of specific language-dependent filtering rules can be added to select the potential phonemes those are possible to be the phoneme conversion of the focus or center-grapheme. In this research, a set of filtering rules is carefully designed based on the Indonesian phonemic rules explained in [3].

L_{10} L_9 L_8 L_7 L_6 L_5 L_4 L_3 L_2 L_1 F R_1 R_2 R_3 R_4 R_5 R_6 R_7 R_8 R_9 R_{10}	Class
**********k**e**.be.rang.	k
*********k**e**.be.rang.k	ə
********ke.be.rang.ka	.
*******ke.**b**e.rang.ka.	b
******ke.b**e**.rang.ka.t	ə
*****ke.be.rang.ka.ta	.
****ke.be.**r**ang.ka.tan	r
***ke.be.r**a**ng.ka.tan*	a
ke.be.ran**g.ka.tan**	ŋ
*ke.be.rang**.**ka.tan***	*
ke.be.rang.ka.tan****	.
e.be.rang.**k**a.tan*****	k
.be.rang.k**a**.tan******	a
be.rang.ka.tan*******	.
e.rang.ka.**t**an********	t
.rang.ka.t**a**n*********	a
rang.ka.tan**********	n

Fig. 4 Converting a grapheme sequence ⟨*ke.be.rang.ka.tan*⟩ into seventeen pairs of pattern and class using $L = 20$

Table 2 Set of filtering rules based on a specific Indonesian phonemic knowledge

#	Filtering rules
1	**if** CG = ⟨a⟩ **and** R1 is not in {⟨i, y, u, w⟩} **then** PP is in {/a, ɑ+ʔ}
2	**if** CG = ⟨a⟩ **and** R1 = ⟨.⟩ **and** R2 is not in {⟨a, e, i, o, u⟩} **then** PP is in {/a/}
3	**if** CG = ⟨e⟩ **and** R1 is not in {⟨i, y⟩} then PP is in {/ɛ, e, ɛ+ʔ, ə+ʔ/}
4	**if** CG = ⟨e⟩ **and** R1 is not in {⟨a, e, i, o, u⟩} PP is in {/ɛ, ə/}
5	**if** CG = ⟨g⟩ **and** L1 is not in {⟨n⟩} **then** PP is in {/g/}
6	**if** CG = ⟨i⟩ **and** L1 is not in {⟨a, e, o⟩} **then** PP is in {/i, i+ʔ/}
7	**if** CG = ⟨i⟩ **and** R1 = ⟨.⟩ **and** R2 is not in {⟨a, e, i, o, u⟩} **then** PP is in {/i, */}
8	**if** CG = ⟨k⟩ **and** R1 is not in {⟨h⟩} **then** PP is in {/k, */}
9	**if** CG = ⟨n⟩ **and** R1 is not in {⟨c, j, y⟩} **then** PP is in {/n, ŋ/}
10	**if** CG = ⟨n⟩ **and** R1 is not in {⟨g⟩,⟨k⟩} **then** PP is in {/n, ɲ/}
11	**if** CG = ⟨o⟩ **and** R1 is not in {⟨i, y⟩} **then** PP is in {/o, o+ʔ/}
12	**if** CG = ⟨s⟩ **and** R1 is not in {⟨y⟩} **then** PP is in {/s/}
13	**if** CG = ⟨u⟩ **and** L1 is not in {⟨a⟩} **then** PP is in {/u, u+ʔ/}
14	**if** CG = ⟨u⟩ **and** R1 = ⟨.⟩ **and** R2 is not in {⟨a, e, i, o, u⟩} **then** PP is in {/u, */}
15	**if** CG = ⟨y⟩ **and** L1 is not in {⟨n, s⟩} **then** PP is in {/j/}

Different from the rule set described in [22], here the rule set takes into account the syllabification point. Table 2 illustrates the set of filtering rule used in this research, where CG is the center-grapheme, L1, R1, and R2 are the first and the second contextual graphemes on the left and the right, and PP is the set of possible phonemes to be the conversion results.

2.3 Grapheme encoding

The grapheme encoding used in this research is the same as the one explained in [22], i.e. Modified partial orthogonal binary encoding (MPOBE). As the name suggest, MPOBE is designed using binary codes those are partially orthogonal based on the categories derived from the manners and places of phoneme articulations in a particular language. In this research, a new category containing a symbol of syllabification point is introduced to model the incorporating syllabification points. The proposed new MPOBE for 30 symbols (26 graphemes, 3 non-graphemes, and one syllabification point) are summarized in Table 3.

Table 3 Modified partial orthogonal binary encoding (MPOBE) for 30 symbols: 26 graphemes, 3 non-graphemes, and one syllabification point

Category	Group	Graphemes	I	II	III	IV	Intragroup
I	1	{a, e, i, o, u}	0	$\sqrt{6}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	2	{b, p}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	3	{t, d}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	4	{k, q, g}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	5	{c, j}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	6	{f, v}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	7	{s, x, z}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	8	{m}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	9	{n}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	10	{h}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	11	{r}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	12	{l}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	13	{w}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
II	14	{y}	$\sqrt{6}$	$\sqrt{4}$	$\sqrt{5}$	$\frac{1}{2}\sqrt{2}$	$\sqrt{2}$
III	15	{*, -, space}	$\sqrt{5}$	$\sqrt{5}$	0	$\frac{1}{2}\sqrt{2}$	0
IV	16	{.}	$\frac{1}{2}\sqrt{2}$	$\frac{1}{2}\sqrt{2}$	$\frac{1}{2}\sqrt{2}$	0	0

Some examples of binary encodings for the categories and groups are illustrated in Table 4, where the complete MPOBE can be seen in [22]. The binary codes for category I and II have 6 different bits, category I and III have 5 different bits. The binary codes for two different groups but in the same category II have 4 different bits. Meanwhile, The binary codes in the same groups (intragroup) for all categories have 2 different bits. A special code is designed for the syllabification point $\langle . \rangle$, where it is not represented in a binary string but the distance is directly defined as $\frac{1}{2}\sqrt{2}$ to all other graphemes.

The detail descriptions, as well as the justifications for all categories and groups, are as follows:

1. All symbols occur in a grapheme sequence are clustered into four categories, i.e. (I) vowel-graphemes, (II) consonant-graphemes, (III) non-graphemes, and (IV) the syllabification point. In Category II, there are 13 groups of

Table 4 Some MPOBEs for the categories and groups

Category	Group	Grapheme	MPOBE
I	1	a	00110000000000000000000000000000000000000000000000
I	1	e	00101000000000000000000000000000000000000000000000
I	1	i	00100100000000000000000000000000000000000000000000
I	1	o	00100010000000000000000000000000000000000000000000
I	1	u	00100001000000000000000000000000000000000000000000
II	2	b	11000000110000000000000000000000000000000000000000
II	2	p	11000000101000000000000000000000000000000000000000
II	3	t	11000000000110000000000000000000000000000000000000
II	3	d	11000000000101000000000000000000000000000000000000
III	15	*	01000000000000000000000000000000000000000000000011
III	15	-	01000000000000000000000000000000000000000000000011
III	15	space	01000000000000000000000000000000000000000000000011
IV	16	.	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

consonant-graphemes generated based on their pronunciation similarities in the manner and the place of articulation described in [3];

2. The binary codes are conceptually designed so that the categories and groups have some different bits, which produce various Euclidean distances. Two binary codes with 6 different bits will have an Euclidean distance of $\sqrt{6}$. The complete Euclidean distances and reasonings for all categories and groups are explained below.
3. The Euclidean distance between Category I (vowel-graphemes) and Category II (consonant-graphemes) is $\sqrt{6}$; This is the biggest distance since they have the highest contextual difference in a word. For example, a vowel-grapheme ⟨a⟩ followed by another vowel-grapheme ⟨u⟩ in ⟨*ker.bau*⟩ (buffalo) must be phonemicized as /aʊ/, but it should be pronounced as /ɑ/ whenever the right contextual grapheme is any consonant-grapheme, such as ⟨b⟩ in the word ⟨*bab*⟩ (chapter).
4. The distance between Category I (vowels) or II (consonants) and Category III (non-graphemes) is $\sqrt{5}$ since their differences are slightly lower. For instance, a grapheme ⟨e⟩ in ⟨*be.rang*⟩ (irascible) is pronounced as /ɛ/ but in ⟨*be.rang-be.rang*⟩ (beaver) is pronounced as /ə/ since it contains a non-graphemic symbol ⟨-⟩;
5. The distance between two graphemes in the different groups but in the same category is $\sqrt{4}$. For instance, a grapheme ⟨e⟩ in ⟨*be.ban*⟩ (load) is converted into /ə/ but in ⟨*be.bas*⟩ (free) is converted into /ɛ/ since the graphemes ⟨n⟩ and ⟨s⟩ are in the different groups;
6. The distance between two graphemes in the same group is $\sqrt{2}$. For example, a grapheme ⟨n⟩ followed by one of the consonants in the same group, i.e. ⟨g⟩ or ⟨k⟩, is pronounced as /ŋ/, such as in ⟨*bangsa*⟩ (nation) and ⟨*bankir*⟩ (banker) those are phonemicized as /bɑŋsɑ/ and /bɑŋkir/, respectively;
7. The distance between Category IV (syllabification point) and all other categories is $\frac{1}{2}\sqrt{2}$. It is designed to be the lowest distance since shifting a syllabification point produces at least two different graphemes. For example, the grapheme sequences ⟨*be.rang*⟩ /bɛ.rɑŋ/ (irascible) and ⟨*ber.a.ngin*⟩

/bər.ɑ.ŋin/ (windy) with the focused-grapheme ⟨e⟩ have two different positions of graphemes ⟨.⟩ and ⟨r⟩.

2.4 Graphemic contextual weight

Pronouncing a grapheme contextually depends on the surrounding graphemes. A contextual grapheme nears the center-grapheme, either on the left or right, is more important than the further one. This concept is formulated as

$$w_i = p^{(L/2)-i+1}, \tag{1}$$

where w_i is the weight of the ith contextual grapheme, p is the exponential constant around 2.0, and L is the contextual length as described in [22].

This formula is actually an exponentially decaying function. The optimum L varies depending on the characteristics of languages. Research in [22] shows that the optimum L for Indonesian G2P is 14. The model is adapted in this research using a longer L of 20.

2.5 Distance of interclass similar patterns

Based on the MPOBE and the graphemic contextual weight, the distance of two interclass similar patterns can be made longer by incorporating syllabification points. Fig. 5 illustrates the distance of two interclass similar patterns with incorporating the information of syllabification points (b) is much bigger than that without the information (a).

In Fig. 5(a), two interclass similar patterns generated from '*berang*' /bɛrɑŋ/ (irascible) and '*berangin*' /bərɑŋin/ (windy) have a low distance with only two different graphemes, i.e. $d = \sqrt{4} \times 2^6 + \sqrt{4} \times 2^5 = 192$ that is calculated using $p = 2.0$. In contrast, Fig. 5(b) shows that both interclass patterns have much higher distance since they have seven different graphemes, i.e. $d = \frac{1}{2}\sqrt{2} \times 2^{10} + \frac{1}{2}\sqrt{2} \times 2^9 + \frac{1}{2}\sqrt{2} \times 2^7 + \sqrt{5} \times 2^6 + \sqrt{2} \times 2^5 + \sqrt{2} \times 2^4 + \sqrt{2} \times 2^3 = 1,398.93$, where the syllabification points shift five graphemes to the right.

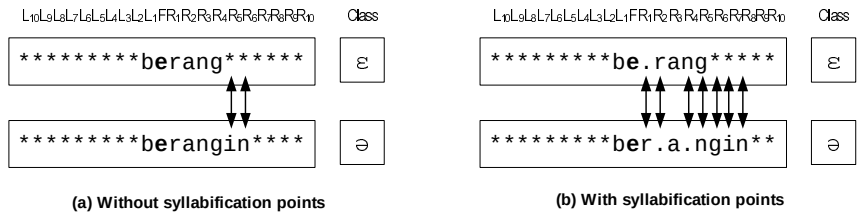

Fig. 5 The distance of two interclass similar patterns with incorporating the information of syllabification points (b) is much bigger than those without the information (a)

2.6 Pseudo nearest neighbour rule

Since this research focuses to evaluate the effectiveness of incorporating the information of syllabification points into G2P, the PNNR described in [22] is used in the same way here to easily make the comparison fair. It works locally by considering a neighbourhood size of k unique patterns in the trainset only, not all available patterns. In [22], the researchers prove that the local scheme is better to solve some problems in a G2P. In this research, the PNNR is also designed to use a neighbourhood weight

$$u_j = \frac{1}{j^c}, \tag{2}$$

where u_j is the neighbourhood weight of the jth neighbour and c is a power constant around 1.0 as described in [22]. This model is directly adopted in this research, but the value of c will be re-optimized to enhance the G2P model.

The PNNR for a G2P model works by minimizing the total distance between the current pattern and k nearest unique-patterns in each possible classes of phonemes to decide the best conversion. The total distance is here calculated using a formula adopted from [22] as follow

$$T = \sum_{j=1}^{k} u_j \sum_{i=1}^{L/2} \left(d_{li} w_i + d_{ri} w_i \right), \tag{3}$$

where u_j is the weight for the j-th neighbour calculated using Equation (2), L is the contextual length, d_{li} and d_{ri} are the distances between the i-th left and right contextual graphemes in both patterns those are calculated using MPOBE, and w_i is the weight of the the ith contextual grapheme calculated using Equation (1).

A holistic illustration is provided here to easily explain all processes in the proposed G2P model. Fig. 6 illustrates an example of converting a grapheme sequence ⟨*a.bai*⟩ (ignore) into a phoneme sequence /ɑbaɪ*/ to clearly explain all steps in the model. In this illustration, the parameters of the PNNR-based G2P model are set to be $L = 4$, $k = 3$, $p = 2$, and $c = 1$ to make it easy to understand. First, a pattern ⟨****a.bai⟩ is generated. Based on the procedure of phonemic rule-based filtering, the possible phoneme conversion for the grapheme ⟨**a**⟩ in the center of pattern is only one, i.e. /ɑ/, then the pattern is directly converted into the phoneme /ɑ/. Next, the second pattern ⟨***a.bai*⟩ is generated. Same as the first step, the patterns is directly converted into a phoneme /./. The third pattern ⟨**a.**b**ai**⟩ is then generated. The pattern is also directly converted into the only possible phoneme /b/.

Next, the fourth pattern ⟨*a.b**a**i***⟩ is generated. This step is different with those previous steps since the pattern has three possible phoneme conversions: /ɑ/, /aɪ/, and /ɑ+ʔ/. In this case, the PNNR is needed to choose the best phoneme conversion. Since the PNNR uses $k = 3$, three closest patterns in each class (phoneme conversion) are selected. Next, the total distance between the three closest patterns in each class are calculated using the formula in Equation

(3). Finally, the class that has the minimum total distance is chosen as the best phoneme conversion. Hence, the grapheme ⟨**a**⟩ in the center of the pattern ⟨*a.b**a**i***⟩ is converted into phoneme /aɪ/.

Finally, the fifth pattern ⟨a.ba**i******⟩ is generated. Same as the fourth step, this pattern has more than one possible phoneme conversions, i.e. /i/ and /*/. In this case, the PNNR is needed to choose the best phoneme conversion. First, three closest patterns in each class (phoneme conversion) are selected. Next, the total distance between the input pattern and the three closest patterns in each class is computed using the formula in Equation (3). Finally, the class that has the lowest total distance is chosen as the best phoneme conversion. In this final step, the grapheme ⟨**i**⟩ in the center of the pattern ⟨a.ba**i******⟩ is converted into phoneme /*/.

3 Results and Discussion

A dataset of 50 k formal words from KBBI as described in [22] is used here to evaluate the proposed G2P model using a 5-fold cross-validation scheme to make a fair comparison to the PNNR-based G2P model without incorporating syllabification points in [22]. Both models are evaluated based on the PER, an error rate in phoneme level, formulated as

$$\text{PER} = \frac{E}{N}, \tag{4}$$

where E is the number of phoneme errors and N is the number of all phonemes in the test set, as used in [22].

3.1 Optimizing the parameters of the proposed G2P model

The proposed PNNR-based G2P with incorporating syllabification points is evaluated based on the five folds cross-validation as used in [22]. To save time and resources, the four parameters of the model are sequentially optimized. Firstly, the parameter k is optimized. Next, the optimum value of c is then searched using the best k. The parameter p is then optimized using those best k and c. Finally, L is tuned using the optimum parameters k, c, and p.

Optimum neighbourhood size k. It is hard to optimize the neighbourhood size k since it is commonly different case by case. Thus, in this research, some experiments are performed for $k = 1$ to 9 using three initial parameters, i.e. $c = 1.0$, $p = 2.0$, and $L = 20$. The five-fold cross-validation shows that a low $k = 1$ produces the highest PER, as illustrated by Fig. 7. A too small k affects the model under fits for the unseen data. In contrast, a too big $k = 9$ makes the model over fits and yields a high PER. In general, the lowest PER is achieved on $k = 5$, where the model yields the smallest PER for the three folds, i.e. Fold 1, Fold 3, and Fold 4.

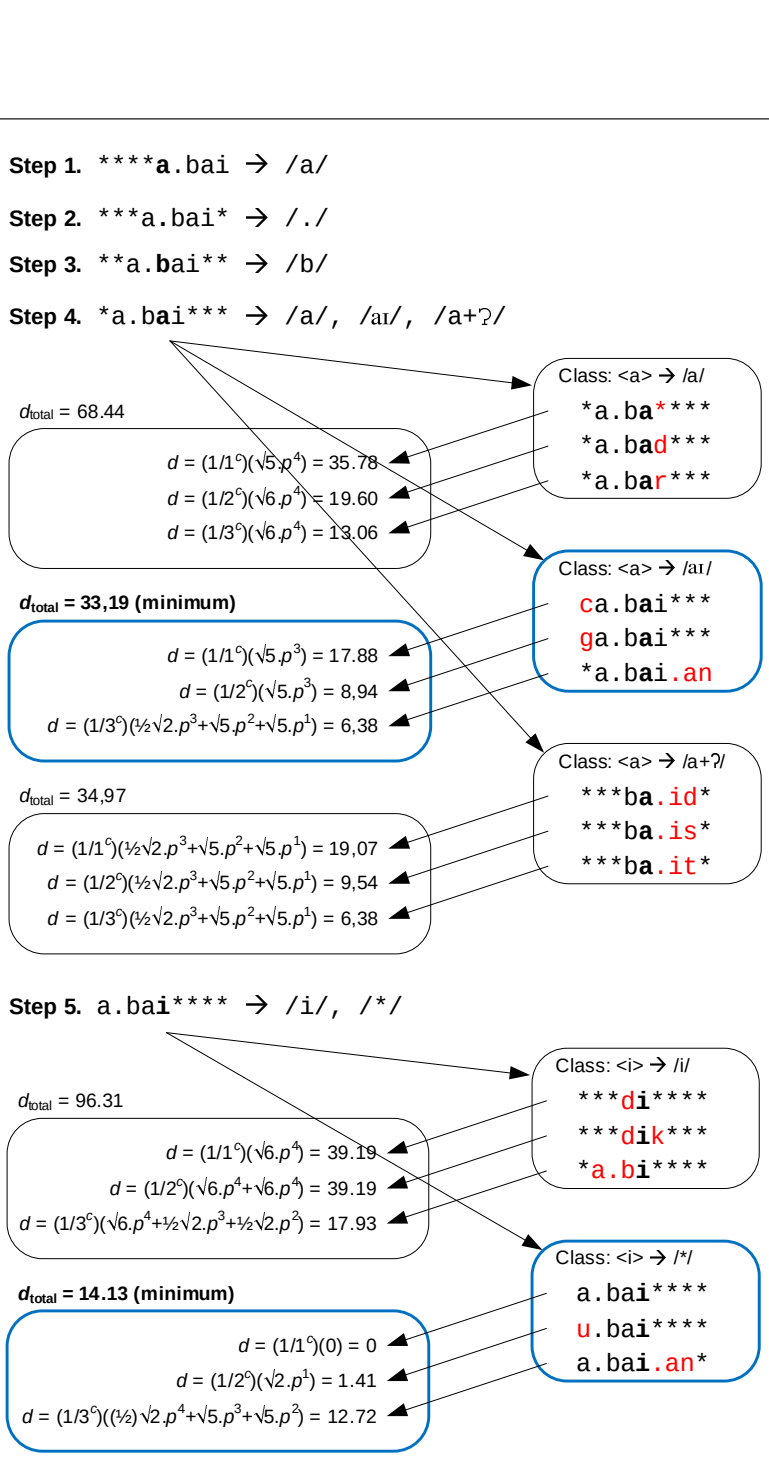

Fig. 6 Example of the proposed PNNR-based G2P model with incorporating syllabification points for a grapheme sequence ⟨*a.bai*⟩ (ignore) into a phoneme sequence /ɑbaɪ*/

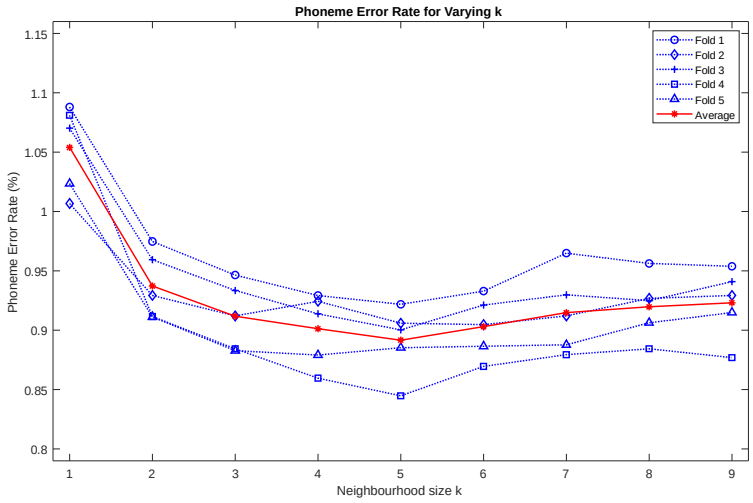

Fig. 7 PER produced by PNNR-based G2P with $c = 1.0$, $p = 2.0$, and varying k for the five folds

Optimum power constant c. Next, the power constant for neighborhood weights c is optimized using the optimum $k = 5$ resulted in the previous experiment as well as two initial parameters $p = 2.0$, and $L = 20$. Here, some experiments are conducted for $c = 0.8$ to 1.6 using the five-fold cross-validation as well. The results in Fig. 8 inform that a small c, lower than 1.0, yields a high PER. A big c more than 1.5 also gives a high PER. A small c make the distances between the close neighbours and the further ones are quite low. In contrast, a big c makes the closer neighbours have significantly high distances while the furthers have low distances. The neighborhood weight is balanced on an optimum $c = 1.2$ for all five folds.

Optimum exponential constant p. The parameter p is then tuned using two optimum parameters $k = 5$ and $c = 1.2$ resulted in the previous experiments and the initial $L = 20$. Some experiments are performed using $p = 1.3$ to 2.1 with a step size of 0.1 using the five-fold cross-validation. The results in Fig. 9 show that a low p yields a high PER. A big p also gives a high PER. Both values make the p is not balanced. The balanced contextual weight is achieved on $p = 1.5$ that produces an averaged PER of 0.84%.

Optimum contextual length L. Finally, the contextual length L is optimized using those three optimum parameters $k = 5$, $c = 1.2$, and $p = 1.5$. Some experiments are conducted using the contextual length $L = 6$ to 22 with a step size of 2. The results in Fig. 10 show that a small $L = 6$ produces a quite high averaged PER up to 4.43% since the model just takes into account a few contextual graphemes. Considering only three contextual graphemes on the

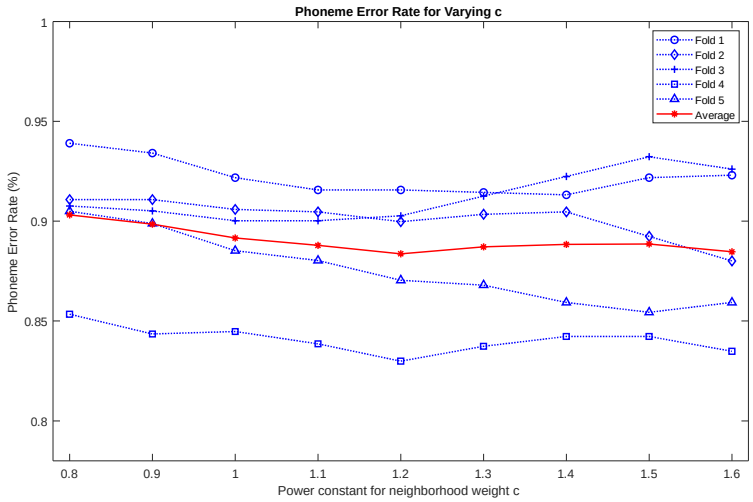

Fig. 8 PER produced by PNNR-based G2P with $c = 1.0$, $p = 2.0$, and varying k for the five folds

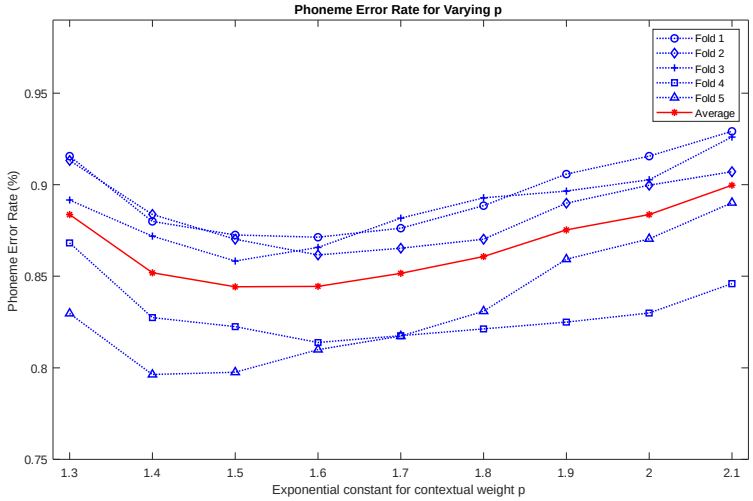

Fig. 9 PER produced by PNNR-based graphemic syllabification with $k = 5$, $c = 1.2$, and varying p for the five folds

left and right produces many ambiguous overlapped patterns. The PER can be sharply reduced by increasing the L. The lowest PERs are reached using $L = 14$ for all five folds that produce the lowest average PER of 0.83%. This result is much better than the PNNR-based G2P model without incorporating

syllabification points that gives PER of 0.93%. It means the incorporating syllabification points relatively reduces the PER by 10.75%.

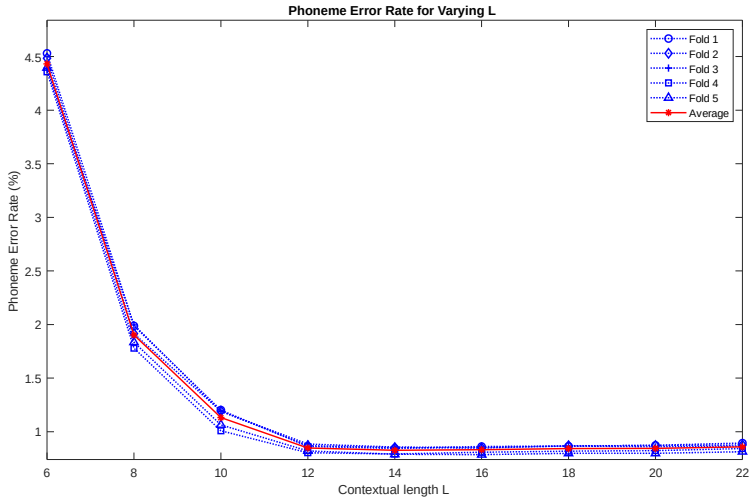

Fig. 10 PER produced by PNNR-based graphemic syllabification with $k = 5$, $c = 1.2$, $p = 1.5$, and varying L for the five folds

3.2 Detailed investigation

A detailed inspection shows that the proposed G2P model is capable of solving some ambiguous conversions of derivatives caused by the four prefixes described in Table 1. Unfortunately, it has a limitation to solve some similar compound words containing a grapheme ⟨e⟩ that is dynamically phonemicized as /ɛ/ or /ə/, e.g. a word '*beban*' (load) is pronounced as /bəbɑn/ but '*bebas*' (free) is pronounced as /bɛbɑs/. Such case is quite hard to be solved by the proposed G2P model although it incorporates the syllabification points.

This problem probably is caused by the blank contextual graphemes in the patterns. In the cases of '*beban*' and '*bebas*', the generated patterns with the contextual length of 20 are mostly dominated by ⟨*⟩. This fact raises a bias in the calculation of the total distance in Equation (3). A possible solution to this problem is taking into account other words on the left or right in a sentence-level context since the cross-word patterns can reduce such bias.

3.3 Robustness of the proposed G2P model

The proposed PNNR-based G2P that incorporates syllabification points with the optimum parameters of $k = 5$, $p = 1.2$, and $p = 1.5$ is finally evaluated

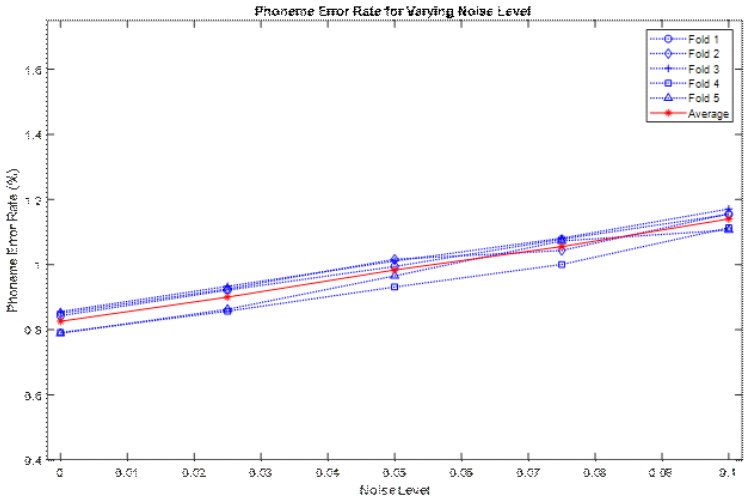

Fig. 11 Effects of noise levels to the PNNR-based G2P with incorporating syllabification points

using four other datasets by including the syllabification noise levels of 2.5%, 5.0%, 7.5%, and 10.0%. The syllabification noise is generated using a random procedure. The procedure is simply implemented by randomly select a position of true syllabification point and then shift it to the left or right. For instances, a true syllabification point in a word '*ber.a.ngin*' is randomly shifted to the left to be '*be.ra.ngin*'.

The results in Fig. 11 show that the proposed model is robust to such noise. A syllabification noise of 2.5% just slightly increases the PER of the proposed G2P model from 0.84% to be 0.92%. A much higher syllabification noise up to 10% just increases the PER to be 1.17%. Therefore, a graphemic syllabification with an SER of around 2.5% as described in [15] does not increase the PER significantly. The graphemic syllabification for name entities with an SER of around 7.5% as described in [15] just slightly increases the PER. So, this proposed G2P model is very promising to be developed to handle the pronunciation of Indonesian formal words as well as name entities.

The achieved high noise-robustness can be explained easily using a simple illustration in Fig. 12. Using $p = 2.0$, the distance of two interclass patterns with a syllabification noise is $d = \frac{1}{2}\sqrt{2} \times 2^7 + \sqrt{5} \times 2^6 + \sqrt{2} \times 2^5 + \sqrt{2} \times 2^4 + \sqrt{2} \times 2^3 = 312.81$, which is lower than that of two interclass patterns without noise (i.e $1,398.93$) but it is bigger than that of two interclass patterns without incorporating the syllabification points described in Fig. 5 (only 192).

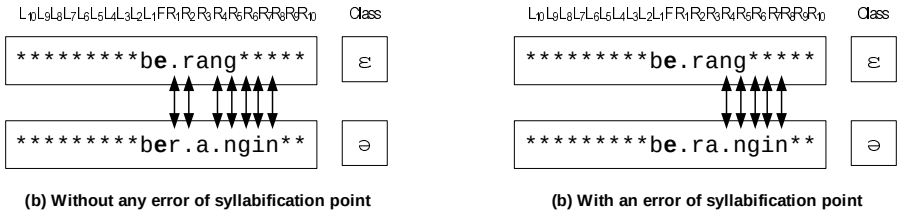

Fig. 12 A shifted syllabification point slightly reduces the distance of two interclass patterns

3.4 Comparison the proposed G2P model with two other works

Finally, the proposed G2P model using the PNNR and the syllabification point (PNNR+SP) is compared with two other models: a PNNR without incorporating syllabification points described in [22] and an inductive learning algorithm (ILA) proposed by Saleh M. Abu-Soud [2]. As reported in [2], ILA is capable of inducing a smaller and simpler rule set, which gives a quite higher accuracy, than an ID3. It also outperforms a neural network-based G2P.

In this research, both models are implemented using the same contextual length L of 14 as used in the PNNR+SP to keep the fairness. Since incorporating syllabification points gives a significant improvement, then ILA is also implemented by incorporating syllabification point so that it is called ILA+SP. Here, ILA+SP is designed to take into account the nearest contextual graphemes first since they have high importance to phonemicize a grapheme. The number of patterns, the number of rules, and the average number of conditions induced by ILA+SP for the five folds are listed in Table 5.

Table 5 Statistics of the rule sets for the five datasets (folds) induced by ILA+SP

Fold	Number of patterns	Number of rules	Average number of conditions
Fold 1	276,719	6,369	1.8848
Fold 2	276,810	6,382	1.8836
Fold 3	277,027	6,336	1.8778
Fold 4	277,101	6,119	1.8784
Fold 5	276,871	6,314	1.8817
Average	**276,906**	**6,304**	**1.8813**

Using the contextual length L of 14, the total number of possible patterns is huge up to $30^{14} = 4.78 \times 10^{20}$. Meanwhile, the actual number of patterns in each fold is tiny of only 276 k. It can be said that the trainset is too small for the problem space. These facts are predicted to make ILA+SP will induce a small rule set that is too hard to generalize the unseen data in the test set. The learning processes on the five folds produce an average number of rules around 6,304 and an average number of conditions only 1.88. With a tiny induced rule set, ILA+SP will take a random decision when no rule available to classify the unseen data. In this research, the classification process

in ILA+SP is simply implemented by selecting the most frequent phoneme as a final decision (output) when no available rule in the induced rule set.

An evaluation using 5-fold cross validation produces the results illustrated in Table 6. The proposed PNNR+SP model produces lower PERs for all five folds. In contrast, ILA+SP gives a worse performance and produces much higher PERs. These are caused by the tiny trainsets in the five folds produce small sets of rules that are under-fit to generalize the unseen data in the test sets, as predicted above. However, ILA-based G2P needs significantly lower computation than PNNR. It just finds the suitable rule in the quite small rule set while PNNR should select k nearest patterns and calculate the total distance from a huge set of patterns in each class.

Table 6 Comparison the proposed PNNR+SP model with the ILA+SP and the PNNR without incorporating syllabification point

Fold	ILA+SP	PNNR	PNNR+SP
Fold 1	5.19	0.94	**0.84**
Fold 2	5.18	0.96	**0.85**
Fold 3	5.22	0.92	**0.86**
Fold 4	4.97	0.93	**0.79**
Fold 5	5.29	0.93	**0.79**
Average	5.17	0.93	**0.83**

4 Conclusion

The parameters of the proposed G2P model based on PNNR+SP are easily tuned using a sequentially optimization. Using the optimum values of those parameters and the perfect information of syllabification points, the proposed G2P model gives a lower PER (0.83%) than the G2P without incorporating information of syllabification points (0.93%). This result is significantly lower compared to the G2P model based on ILA+SP. A further investigation shows that the proposed model successfully solves some ambiguous conversions of derivative words caused by the four prefixes '*ber*', '*me*', '*per*', and '*ter*'. But, it fails to solve some similar compound words containing a grapheme ⟨e⟩ that is randomly pronounced as /ɛ/ or /ə/. Another important achievement of this research is that the proposed G2P model is quite robust to the syllabification noise. The SER of 2.5% just slightly increase the PER of the proposed G2P model from 0.83% to be 0.90%. A much higher SER up to 10% just increase its PER to be 1.14%. Hence, this model is very promising to handle the pronunciation of Indonesian formal words. In the future, it can be evaluated to pronounce the name entity dataset.

Acknowledgements I would like to thank Muhammad Agha Ariyanto for the inspiration as well as to Telkom University for the support.

References

1. Aachen, R.: Bayesian joint-sequence models for grapheme-to-phoneme conversion. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2836–2840 (2017). DOI 10.1109/ICASSP.2017.7952674
2. Abu-soud, S.M.: ILATalk: a new multilingual text-to-speech synthesizer with machine learning. International Journal of Speech Technology **19**(1), 55–64 (2016). DOI 10.1007/s10772-015-9322-4
3. Alwi, H., Dardjowidjojo, S., Lapoliwa, H., Moeliono, A.M.: Tata Bahasa Baku Bahasa Indonesia (The Standard Indonesian Grammar), 3 edn. Balai Pustaka, Jakarta (1998)
4. Andersen, O., Dalsgaard, P.: Multi-lingual testing of a self-learning approach to phonemic transcription of orthography. In: EUROSPEECH, pp. 1117–1120 (1995)
5. Bartlett, S., Kondrak, G., Cherry, C.: Automatic syllabification with structured SVMs for letter-to-phoneme conversion. In: Proceedings of Human Language Technologies: The 2008 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 568–576. Columbus, Ohio (2008)
6. Bosch, A.V.D., Daelemans, W.: Data-oriented methods for grapheme-to-phoneme conversion. In: The sixth conference on European chapter of the Association for Computational Linguistics (EACL), pp. 45–53. Association for Computational Linguistics, Morristown, NJ, USA (1993). DOI 10.3115/976744.976751. URL http://portal.acm.org/citation.cfm?doid=976744.976751
7. Chaer, A.: Fonologi Bahasa Indonesia (Indonesian Phonology). Rineka Cipta, Jakarta (2009)
8. Deri, A., Knight, K.: Grapheme-to-Phoneme Models for (Almost) Any Language. In: The 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 399–408 (2016)
9. Jyothi, P., Hasegawa-Johnson, M.: Low-Resource Grapheme-to-Phoneme Conversion Using Recurrent Neural Networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2017)
10. Marchand, Y., Adsett, C.R., Damper, R.I.: Automatic syllabification in English: a comparison of different algorithms. Language and speech **52**(Pt 1), 1–27 (2009). DOI 10.1177/0023830908099881
11. Marchand, Y., Damper, R.I.: Can syllabification improve pronunciation by analogy of English? Natural Language Engineering **13**(01), 1 (2006). DOI 10.1017/S1351324905004043. URL http://www.journals.cambridge.org/abstract_S1351324905004043
12. Milde, B., Schmidt, C., Joachim, K., Augustin, S., Milde, B., Schmidt, C.A.: Multi-task Sequence-to-Sequence Models for Grapheme-to-Phoneme Conversion. In: INTERSPEECH, pp. 2536–2540 (2017). DOI 10.21437
13. Mohr, M.: Asking for Help: Grapheme-to-Phoneme Conversion using Active Learning at Application Time. Ph.D. thesis, Hamburg University (2017)
14. Mousa, A.E.d.: Deep Bidirectional Long Short-Term Memory Recurrent Neural Networks for Grapheme-to-Phoneme Conversion utilizing Complex Many-to-Many Alignments. In: Interspeech, pp. 2836–2840 (2016)
15. Parande, E.A., Suyanto, S.: Indonesian graphemic syllabification using a nearest neighbour classifier and recovery procedure. International Journal of Speech Technology **22**(1), 13–20 (2018). DOI 10.1007/s10772-018-09569-3
16. Parfitt, S.H., Sharman, R.A.: A bi-directional model of English pronunciation. In: EUROSPEECH, pp. 801–804 (1991)
17. Peters, B.: Massively Multilingual Neural Grapheme-to-Phoneme Conversion (2017)
18. Rao, K., Peng, F., Sak, H., Beaufays, F.: Grapheme-to-Phoneme Conversion Using Long Short-Term Memory Recurrent Neural Networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2015)
19. Razavi, M., Rasipuram, R., Magimai, M.: Acoustic Data-Driven Grapheme-to-Phoneme Conversion in the Probabilistic Lexical Modeling Framework. Speech Communication **80**, 1–21 (2016)
20. Soky, K., Lu, X., Shen, P., Vanna, C., Kato, H.: Building WFST based Grapheme to Phoneme Conversion for Khmer. In: Khmer Natural Language Processing (2016)

21. Suyanto, Harjoko, A.: Nearest neighbour-based Indonesian G2P conversion. Telkomnika (Telecommunication, Computing, Electronics, and Control) **12**(2), 389–396 (2014)
22. Suyanto, S., Hartati, S., Harjoko, A.: Modified Grapheme Encoding and Phonemic Rule to Improve PNNR-Based Indonesian G2P. International Journal of Advanced Computer Science and Applications (IJACSA) **7**(3), 430–435 (2016)
23. Suyanto, S., Hartati, S., Harjoko, A., Compernolle, D.V.: Indonesian syllabification using a pseudo nearest neighbour rule and phonotactic knowledge. Speech Communication **85**, 109–118 (2016). DOI 10.1016/j.specom.2016.10.009. URL http://dx.doi.org/10.1016/j.specom.2016.10.009
24. Toshniwal, S., Livescu, K.: Read, Attend and Pronounce: An Attention-Based Approach for Grapheme-To-Phoneme Conversion (2016)
25. Tsujioka, S., Sakti, S., Yoshino, K., Neubig, G., Nakamura, S.: Unsupervised Joint Estimation of Grapheme-to-phoneme Conversion Systems and Acoustic Model Adaptation for Non-native Speech Recognition. In: INTERSPEECH, 1 (2016). DOI 10.21437/Interspeech.2016-919
26. Wang, D., King, S.: Letter-to-sound Pronunciation Prediction Using Conditional Random Fields. IEEE Signal Processing Letters **18**(2), 122–125 (2011)
27. Yao, K., Zweig, G.: Sequence-to-Sequence Neural Net Models for Grapheme-to-Phoneme Conversion (2015)

Evidences of correspondences

Incorporating Syllabification Points Into Grapheme-to-Phoneme Conversion

1. First submission (31 December 2018)
2. LoA with Major Revision (20 March 2019)
3. Respond to Reviewers (10 April 2019)
4. Final submission (10 April 2019)
5. **LoA with Fully Accepted (25 April 2019)**

Date: 25 Apr 2019
To: "Suyanto Suyanto" suyanto@telkomuniversity.ac.id
From: "International Journal of Speech Technology (IJST)" Ramya.Thulasingam@springer.com
Subject: Decision on your manuscript #IJST-D-18-00237R1

Dear Dr. Suyanto:

We are pleased to inform you that your manuscript, "Incorporating Syllabification Points into a Model of Grapheme-to-Phoneme Conversion" has been accepted for publication in International Journal of Speech Technology.

You will receive an e-mail from Springer in due course with regard to the following items:
1. Offprints
2. Colour figures

3. Transfer of Copyright
Please remember to quote the manuscript number, IJST-D-18-00237R1, whenever inquiring about your manuscript.
With best regards,
Amy Neustein, Ph.D.
Editor-in-Chief
International Journal of Speech Technology