International Journal of Speech Technology (2020) 23:141–147
https://doi.org/10.1007/s10772-019-09662-1

An evaluation of sentence selection methods on the different phone-sized units for constructing Indonesian speech corpus

Muljono[1] · Agus Harjoko[2] · Nurul Anisa Sri Winarsih[1] · Catur Supriyanto[1]

Received: 3 November 2018 / Accepted: 13 December 2019 / Published online: 23 December 2019

Abstract
Collecting phonetically balanced text corpus is an important step to develop automatic speech recognition and text-to-speech systems. A corpus should have a small number of sentences but contains all phonetic units, such as monophone, triphone, and pentaphone units. There are exist least-to-most greedy algorithm (LTM + Greedy) and its variant to select the minimum sentence set. The variant is on the sentence scoring method, which affect the number of selected sentences. In this paper, we evaluate the sentence scoring methods by Zhang and Suyanto on LTM + Greedy algorithm. The sentence scoring methods are conducted on triphone and pentaphone units on the collection of sentence set. Triphone and pentaphone units have offered higher quality synthesized speech than monophone unit. The dataset of this paper is Indonesian sentences that collected from holy book translation, news, novel, dialog, monologue, and question sentences. Totally 115,489 sentences are used for the experiments. Based on the experiments, LTM + Greedy by Suyanto produces a smaller number of sentences that contain large number of phone units.

Keywords Indonesian minimum sentence set · Phonetically balanced sentence set · Speech corpus · Least-to-most greedy algorithm

1 Introduction

High-quality speech corpus is crucial for developing automatic speech recognition (ASR) and text-to-speech (TTS) synthesis systems. Both ASR and TTS can be used to develop a machine translation. ASR converts from speech into text and TTS converts text into speech. ASR was also implemented in various application such as hands-free operation and control, automatic query answering, telephone interactive voice response systems, and automatic dictation (Alghamdi et al. 2007). Speech corpus contains the audio files and their transcripts (so-called text corpus) (Patel and Kopparapu 2015). The performance of ASR and TTS depends on the phonetically balanced text corpus (Abushariah et al. 2010, 2012). The corpus should have a small number of sentences but cover all phonetic units.

Speech corpus can be constructed based on syllables and phoneme. The difference between syllable and phoneme is that the syllable consists of the vowel and its consonant, whilst the phoneme is the smallest unit of sound. For example, the word 'me' (aku) is made up of one syllable. However, the word 'me' may have two phonemes, i.e. [m] and [e]. The Barkhoda et. al's study (Barkhoda et al. 2009) showed that phoneme-based produced more naturally speech synthesizer than syllable-based. Therefore, we study the phoneme-based speech corpus. In the speech synthesizer, there are several frequently used phone-sized units for producing high quality synthesized speech, i.e. monophone, triphone, and pentaphone units. Longer phone-sized units provide high natural speech synthesizer (Anushiya et al. 2013, 2015). In the study of Mandarin auto speech recognition, (Xu et al.

✉ Muljono
muljono@dsn.dinus.ac.id

Agus Harjoko
aharjoko@ugm.ac.id

Nurul Anisa Sri Winarsih
nurulanisasw@dsn.dinus.ac.id

Catur Supriyanto
catur.supriyanto@dsn.dinus.ac.id

[1] Department of Informatics Engineering, Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang, Indonesia

[2] Department of Computer Sciences and Electronics, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Yogyakarta, Indonesia

2018) showed that the triphone model produced high recognition rate compared to the monophone model.

How to select the minimum sentences set that contain all phonetic units is the issue of this study. The classical method to handle this issue is a greedy search algorithm (van Santen and Buchsbaum 1997). The algorithm selects the sentences from the mother sentence set by scoring the sentence based on the uncovered units. The standard greedy (SG) algorithm has been implemented to create a speech corpus in many languages such as Indonesia (Suyanto 2006), Bangla (Murtoza Habib et al. 2011), and Czech (Matouek and Romportl 2006). SG algorithm searches the minimum sentences by calculating the score of each sentence. For each looping, the sentence with the highest score is selected. Sentences scoring becomes an essential aspect to select the best sentences.

However, SG algorithm produces high computational time and large number of generated sentences. To reduce the computational cost of SG algorithm, Zhang and Nakamura proposed least-to-most greedy (LTM + Greedy) algorithm which selects the sentence only from the subset sentence or those sentences that contain the unit of least frequency (Zhang and Nakamura 2001). The objective of SG and LTM + Greedy algorithms is to generate the minimum sentence set which covers all phonetic units. The objectives of this study is to compare the sentence selection methods on different phone-sized units, especially on the triphone and pentaphone units.

The rest of this paper is organized as follows: Sect. 2 reviews the related works. The methodology is introduced in Sect. 3. Section 4 gives the results and discussion. Section 5 concludes our work and gives future works.

2 Indonesian text corpus

2.1 The Indonesian phoneme

Indonesia is located in Southeast Asia with Jakarta as the capital city. Although Bahasa Indonesia becomes a national language of Indonesia, more than 195 million people in Indonesia speak Bahasa Indonesia as second language (Sakti et al. 2004). Many people in Indonesia speak with the traditional language, such as Javanese, Sundanese, and Balinese (Muljono et al. 2016a). Bahasa Indonesia is a language from the Austronesian family (O'Grady and Archibald 2000). It is spoken not only in Indonesia but also in Malaysia, Singapore, Southern Thailand, and Brunei (called Bahasa Melayu). Besides, Bahasa Indonesia becomes one of the minority language in the Netherlands (Comrie 2009).

Bahasa Indonesia has 35 phonemes, consists of 6 vocal phonemes, 4 diphthong phonemes, 24 consonant phonemes and a silence (Suyanto 2006; Muljono et al. 2016b, c). A phoneme is the smallest sound unit. Determining phoneme is the first step to build a text to speech synthesis (TTS). This step establishes the correct utterance, it directly gives impact to design the speech corpus. This study adds 2 consonant phonemes, there are 'z'and 'x'. Those additional phonemes are borrowed from other languages such as Arabic and English. Table 1 shows Bahasa Indonesia phonemes.

Defining the unit of the phonemes is important for phonetically balanced text corpus (Murtoza Habib et al. 2011). The units are classified into monophones (one phoneme), diphones (two phonemes), triphones (three phonemes), and pentaphones (five phonemes). Diphones are classified into right and left diphones. Right diphones are the diphones that taken from left to right of the sentence and left diphones are the diphones that taken from right to left of the sentence. This study uses triphones and pentaphones since the wave of speech signal depends on its previous and next phonemes (Suyanto 2006). The examples of the phonetically balancing units are shown in Table 2.

2.2 Sentence selection algorithms

Greedy (Zhang and Nakamura 2003) is the classical algorithm for selecting the minimum sentence set for speech corpus. The detail standard greedy algorithm (SG) is shown in Algorithm 1. First, we have to provide the mother sentence set (denoted as S) and to-be-covered units list (denoted as U) which is taken from the mother sentence set. In each iteration, the algorithm scores all sentences in S and the sentence with the highest score S_h is selected. Based on the selected sentence, delete the to-be-covered units that contained in S_h from U. The iteration stops when the to-be-covered units list in U is empty.

Table 1 Bahasa Indonesia phonemes

No	Phoneme	Indonesian	Pronunciation in english	No	Phoneme	Indonesian	Pronunciation in english
Vocal				8	/k/	kasih	keep
1	/a/	kamar	father	9	/l/	lekas	loose
2	/e/	meja	about, ago	10	/m/	makan	main
3	/ê/	gelas	learn	11	/n/	nasi	name
4	/i/	bila	meet	12	/p/	pasang	pen
5	/o/	bola	odd	13	/r/	roti	rise
6	/u/	burung	boot	14	/s/	sopan	small
Diphthong				15	/t/	tidur	team
1	/ai/	sungai	hide	16	/v/	versi	very
2	/ou/	kerbau	how	17	/w/	wakil	west
3	/oi/	tomboi	boy	18	/y/	yang	you
4	/ei/	survei	survey	19	/z/	zalim	zoo
Consonant				20	/x/	xavier	axis
1	/b/	besar	bone	21	/kh/	khabar	loch
2	/c/	cari	cheese	22	/ng/	ngeri	singing
3	/d/	dekat	dig	23	/ny/	nyaman	canyon
4	/f/	faham	flower	24	/sy/	syair	share
5	/g/	ganti	give	*Silence*			
6	/h/	hutan	happy	1	Sil	Silence	
7	/j/	jarum	judge				

Table 2 Examples of phonetically balancing units

Units	Examples
Sentence	Makan malam. (Dinner)
Monophones	[sil] [m] [a] [k] [a] [n] [sil] [m] [a] [l] [a] [m] [sil]
Left diphones	[sil-m] [m-a] [a-k] [k-a] [a-n] [n-sil] [sil-m] [m-a] [a-l] [l-a] [a-m]
Right diphones	[m+a] [a+k] [k+a] [a+n] [n+sil] [sil+m] [m+a] [a+l] [l+a] [a+m] [m+sil]
Triphones	[sil-m+a] [m-a+k] [a-k+a] [k-a+n] [a-n+sil] [n-sil+m] [sil-m+a] [m-a+l] [a-l+a] [l-a+m] [a-m+sil]
Pentaphone	[sil-sil-m+a+k] [sil-m-a+k+a] [m-a-k+a+n] [a-k-a+n+sil] [k-a-n+sil+m] [a-n-sil+m+a] [n-sil-m+a+l] [sil-m-a+l+a] [m-a-l+a+m] [a-l-a+m+sil] [l-a-m+sil+sil]

Algorithm 1 The Standard Greedy Algorithm (SG)

Input: S = {all mother sentence set}, B={null}, U={the to-be-covered units}
Output: B ={the minimum sentence set}
1: **while** U is not empty **do**
2: Compute covering score S_i for each sentence i according to Eq. (1).
3: Select the sentence S_h with the highest score and insert it into B, then delete all newly covered units in S_h from U.
4: **end while**

Algorithm 2 Least-to-Most Greedy Algorithm (LTM+Greedy)

Input: S_{u_k} = {all sentences containing at least one token of u_k}, B={null}, U={the to-be-covered units}
Output: B is the minimum sentence set
1: Put all the to-be-covered units in U to a queue in ascending order, $Q = \{u_1, u_2, , u_w\}$, where u_1 is the least frequent unit and u_w is the most frequent one in S.
2: **while** Q is not empty **do**
3: Use SG search algorithm to find the best sentence S_h and insert it into B
4: Delete all the newly covered units in S_h from Q
5: **end while**

To reduce the computational cost of SG algorithm, (Zhang and Nakamura 2001) sorted the to-be-covered units based on their frequency of appearance in ascending order, the method called least-to-most greedy algorithm (LTM + Greedy). Each uncovered units will have the subset of sentences which contain at least one token of the uncovered unit. LTM + Greedy is faster than SG since LTM + Greedy only find the best sentence from the subset. LTM + Greedy is shown in Algorithm 2.

The similar research was developed by Suyanto (2007) which proposed modified sentence scoring for LTM + Greedy algorithm. The modified sentence scoring by Suyanto is presented in Eq. (2). He addressed the issue that the sentence scoring by Zhang and Nakamura (2003) scored the long sentence with the low score.

$$S_i = \frac{\text{Types of uncovered units in sentence } i}{\text{Total tokens of units in sentence } i} \quad (1)$$

$$S_i = \text{types of uncovered units in sentence } i \quad (2)$$

3 Methodology

3.1 Preprocessing steps

In this study, the raw text corpus was collected from many sources, such as holy book translation, news, novel, dialog, monologue, and question sentences. The preprocessing steps of the raw text corpus are described as follows:

1. Sentence segmentation: Split the sentences based on punctuation such as full stop (.), question mark (?), exclamation mark (!), and quotation marks (' or ").
2. Number and symbol conversion: convert the number or symbol to words, for example, *123* becomes *seratus dua puluh tiga* (one hundred and twenty three) and symbol $ becomes *dolar* (dollar). We also delete the hypen symbol, for example, *laki-laki* (some men) becomes *laki laki*.
3. Inspection of e: Check all letters *e* and adjust based on how to read. For example, *meja* (table) becomes *m@ja*.

Table 3 shows the detailed statistic of our mother sentence set. The preprocessing steps generate 115,489 sentences to become the mother sentence set. There are 6,225,794 triphones and 5,741,062 pentaphones which are appeared in the mother sentence set. Meanwhile, the number of distinct triphones and distinct pentaphones are 13,501 and 214,868, respectively. The number of distinct triphones or distinct pentaphones is the number without any duplication. For example, the number triphones in sentence makan malam are 11 triphones, i.e., [sil-m+a] [m-a+k] [a-k+a] [k-a+n] [a-n+sil] [n-sil+m] [sil-m+a] [m-a+l] [a-l+a] [l-a+m] [a-m+sil]. Triphone [sil-m+a] occurs two times. Thus, the number of distinct triphones are 10 triphones, i.e., [sil-m+a] [m-a+k] [a-k+a] [k-a+n] [a-n+sil] [n-sil+m] [m-a+l] [a-l+a] [l-a+m] [a-m+sil]. The example of pentaphones can be seen in Table 2.

Table 3 Statistic of the mother sentence set

No	Parameter	Count
1	Number of sentences	115,489
2	Number of words appear	826,115
3	Number of distinct words	42,146
4	Average number of phonemes per sentence	48.99
5	Maximum number of phoneme in a sentence	201
6	Minimum number of phonemes in a sentence	4
7	Number of triphones	6,225,794
8	Number of triphones type (distinct triphone)	13,501
9	Number of pentaphones	5,741,062
10	Number of pentaphones type (distinct pentaphones)	214,868

3.2 Experimental design

The two-sentence scoring methods are applied to select the minimum sentence set from the mother sentence set. The first method was proposed by Zhang and Nakamura (2003) as shown in Eq. 1, the second method was proposed by Suyanto (2007) as shown in Eq. 2. The two-sentence scoring methods are evaluated on the LTM + Greedy algorithms. All the experiments run on physical memory (RAM) of 16 GB. Similar to the previous research (Zhang and Nakamura 2003), our experimental results are described in three points of view: the size of the generated sets, search analysis, and computation costs.

4 Experimental results

4.1 Size of the generated sets

Table 4 shows the results of the two methods. Both methods are applied on triphone and pentaphone units. In the number of sentences, LTM + Greedy by Suyanto produces a slightly smaller number of sentences than LTM + Greedy by Zhang in both triphones and pentaphones. In the triphone, LTM + Greedy by Suyanto and Zhang generate 3443 and 3531 sentences, respectively. Meanwhile in the pentaphone, LTM + Greedy by Suyanto and Zhang generate 35,816 and 36,798 sentences, respectively.

Table 4 Results of Algorithms

No	Parameter	LTM + Greedy by Zhang Zhang and Nakamura (2003)		LTM + Greedy by Suyanto Suyanto (2007)	
		Triphones	Pentaphones	Triphones	Pentaphones
1	Number of sentences	3,531	36,798	3,443	35,816
2	Number of words appear	22,728	262,129	24,861	263,543
3	Number of distinct words	10,238	38,366	10,603	38,435
4	Average number of phonemes per sentence	43.71	48.24	49.86	50.07
5	Maximum number of phonemes in a sentence	125	201	157	201
6	Minimum number of phonemes in a sentence	5	5	5	5
7	Number of phone units	158,876	1,797,491	188,623	1,815,989

In the number of words appear and the number of distinct words, LTM + Greedy by Zhang produces the fewer number in triphone and pentaphone. In the average number of the phoneme per sentence, the maximum number of the phoneme in a sentence, and the minimum number of the phoneme in a sentence, the two methods almost have a similar number. Based on the generated sentences from the both methods, LTM + Greedy by Suyanto is capable to select the minimum sentence set that contains large number of phonetic units.

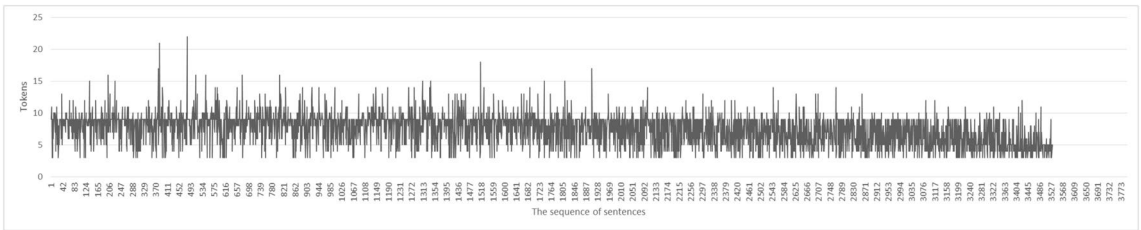
From the seven parameters listed in Table 4, generally, the number of sentence and number of phone units can be used as the performance measure. The method performs best when generating the minimum number of sentence and have a large number of phoneme. The experiments show that LTM + Greedy by Suyanto generated the fewest number of sentence set in triphone and pentaphone units.

4.2 Search analysis

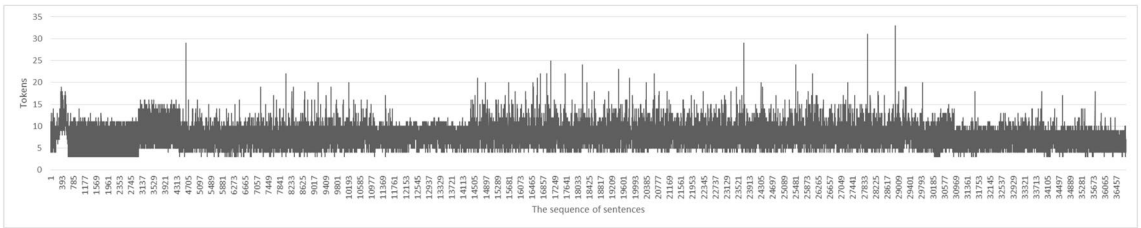
Figures 1 and 2 are used to analyze the search performance of the two methods in the triphone and pentaphone units, respectively. The performance of the two methods is almost similar. The two methods tend to select the sentences with the same number of token in each iteration.

4.3 Computation costs

We report the computational cost of each method in Table 5. The computational cost of LTM + Greedy by Zhang and Suyanto is not different to much. It means that the

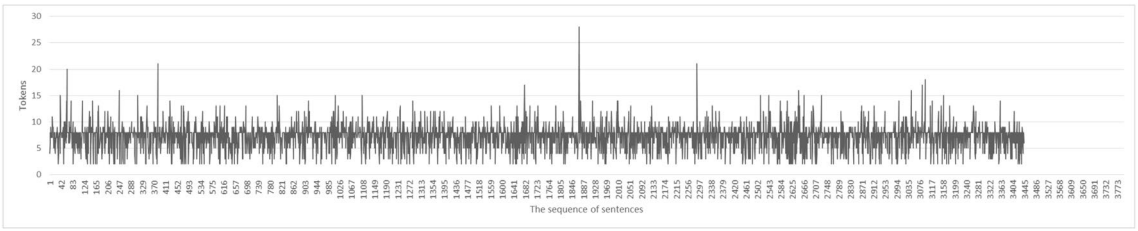


(a) Triphones

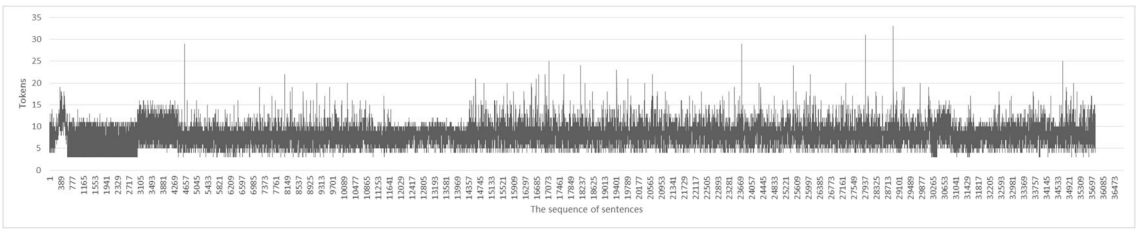


(b) Pentaphones

Fig. 1 Number of Tokens of Different Sentences Scoring on LTM + Greedy Algorithm by Zhang Zhang and Nakamura (2003)



(a) Triphones



(b) Pentaphones

Fig. 2 Number of Tokens of Different Sentences Scoring on LTM + Greedy Algorithm by Suyanto Suyanto (2007)

Table 5 Computational Time

	Phone-sized units	Time
LTM + Greedy by Zhang	Triphones	2 s
	Pentaphones	3 min 16 s
LTM + Greedy by Suyanto	Triphones	2 s
	Pentaphones	3 min 4 s

complexity of both methods is equally well by generate the minimum sentence set in seconds or minutes for triphone and pentaphone units.

5 Conclusions and future works

This paper presents the evaluation of sentence selection methods on different phone-sized units, i.e. triphone and pentaphone units. The selected sentences set is useful for constructing the natural speech corpus. The sentence is collected from several sources in Bahasa Indonesia. The experimental results show that the LTM + Greedy by Suyanto successfully generate the minimum sentence set compared to LTM + Greedy by Zhang for constructing Indonesian text corpus. Not only produces the smaller number of sentences, but also contains large number of phone units. In the future, the generated minimum sentence set can be applied to develop speech corpus for Indonesian TTS synthesis system. We can evaluate how naturally TTS generate the speech from text.

Acknowledgements The research was funded by the Ministry of Research, Technology, & Higher Education of Indonesia through Post Doctoral Research Scheme 2018 (Grant No. 028/K6/KM/SP2H/PENELITIAN/2018).

References

Abushariah, M. A. M., Ainon, R. N., Zainuddin, R., Elshafei, M., & Khalifa, O. O. (2010). Phonetically rich and balanced speech corpus for Arabic speaker-independent continuous automatic speech recognition systems. In *10th international conference on information sciences, signal processing and their applications* (pp. 65–68).

Abushariah, M. A., Ainon, R. N., Zainuddin, R., Elshafei, M., & Khalifa, O. O. (2012). Phonetically rich and balanced text and speech corpora for Arabic language. *Language Resources and Evaluation*, *46*(4), 601–634.

Alghamdi, M., Elshafei, M., & Al-Muhtaseb, H. (2007). Arabic broadcast news transcription system. *International Journal of Speech Technology*, *10*(4), 183–195.

Anushiya Rachel, G., Lilly Christina, S., Sherlin Solomi, V., Ramani, B., Vijayalakshmi, P., & Nagarajan, T. (2013). Development and analysis of various phone-sized unit-based speech synthesizers. In *International conference oriental COCOSDA held jointly with 2013 conference on asian spoken language research and evaluation* (pp. 1–5).

Anushiya Rachel, G., Sherlin Solomi, V., Naveenkumar, K., Vijayalakshmi, P., & Nagarajan, T. (2015). A small-footprint context-independent HMM-based synthesizer for Tamil. *International Journal of Speech Technology*, *18*(3), 405–418.

Barkhoda, W., ZahirAzami, B., Bahrampour, A., & Shahryari, O. (2009). A comparison between allophone, syllable, and diphone based TTS systems for kurdish language. In *International conference oriental COCOSDA held jointly with 2013 conference on asian spoken language research and evaluation (O-COCOSDA/CASLRE)* (pp. 557–562).

Comrie, B. (2009). *The world's major languages* (2nd ed.). NY: Routledge.

Matouek, J., & Romportl, J. (2006). On building phonetically and prosodically rich speech corpus for text-to-speech synthesis. In *Proceedings of the second IASTED international conference on computational intelligence* (pp. 1–6).

Muljono, Sumpeno, S., Arifianto, D., Aikawa, K., & Purnomo, M. H. (2016a). Developing an online self-learning system of Indonesian pronunciation for foreign learners. *International Journal of Emerging Technologies in Learning*, *11*(4), 83–89.

Muljono, M., Sumpeno, S., Arifianto, D., Aikawa, K., & Purnomo, M. H. (2016b). Indonesian text to audio visual speech with animated talking head. *International Review on Computers and Software*, *11*(3), 261–269.

Muljono, Winarsih, N. A., & Supriyanto, C. (2016c). Evaluation of classification methods for Indonesian text emotion detection. In *International seminar on application for technology of information and communication (ISemantic)* (pp. 130–133).

Murtoza Habib, S. M., Alam, F., Sultana, R., Absar Chowdhur, S., & Khan, M. (2011). Phonetically balanced Bangla speech corpus. In *Conference on human language technology for development* (pp. 87–93).

O'Grady, W., & Archibald, J. (2000). Contemporary linguistic analysis: an introduction. *Pearson Canada* (pp. 130–133).

Patel, C., & Kopparapu, S. K. (2015). *A multi-criteria textselection approach for building a speech corpus international conference on text speech and dialogue* (pp. 15–22). Cham: Springer.

Sakti, S., Arman, A. A., Nakamura, S., & Hutagaol, P. (2004). Indonesian speech recognition for hearing and speaking impaired people. In *8th international conference on spoken language processing* (pp. 1037–1040).

Suyanto. (2007). An Indonesian phonetically balanced sentence set for collecting speech database. *Jurnal Teknologi Industri*, *11*(1), 59–68.

Suyanto. (2006). Modified least-to-most greedy algorithm to search a minimum sentence set. TENCON (pp. 1–3).

van Santen, J. P. H., & Buchsbaum, A. L. (1997) Methods for optimal text selection. In *Proceedings of Eurospeech* (pp. 553–556). Rhodes, Greece

Xu, J., Zhu, Y., Xu, P., & Ma, D. (2018). Agricultural price information acquisition using noise-robust Mandarin auto speech recognition. *International Journal of Speech Technology*, *21*(3), 681–688.

Zhang, J., & Nakamura, S. (2001). Least-to-most ordered search for minimum sentence set for collecting speech database. In *Proceedings of ASJ* (pp. 145–146).

Zhang, J., & Nakamura, S. (2003). An efficient algorithm to search for a minimum sentence set for collecting speech database. *Proceedings of ICPhS* (pp. 3145–3148).