Design of Javanese Text to Speech Application

Yulia, Liliana, Rudy Adipranata, Gregorius Satia Budhi
Informatics Department, Industrial Technology Faculty, Petra Christian University
Surabaya, Indonesia
yulia@petra.ac.id

***Abstract*—Javanese is one of the many regional languages used in Indonesia. Javanese language is used by most of the population in Java. But now along with the development of the era, the use of regional languages including Javanese language is to be reduced especially among the younger generation. One way to help conserve the use of Javanese language is to utilize information technologies, one of them is by developing a text to speech application that can be used to find out how the pronunciation of Javanese language. In this paper, we discussed the design for Javanese text to speech applications uses finite state automata. The design result will be used as rules to separate syllables when implementing text to speech application.**

***Index Terms*—Javanese language; Finite state automata; Text to speech.**

I. Introduction

Javanese language is a language widely spoken by the people of Java. It is one of the regional languages of many regional languages spoken in Indonesia. As one of the assets of national culture, Javanese language needs to be preserved. The younger generation is now more interested in learning a foreign language, rather than the native Indonesian local language. One of the reasons is that it is difficult to learn the local language where the local language in particular has pronunciation and special form of characters.

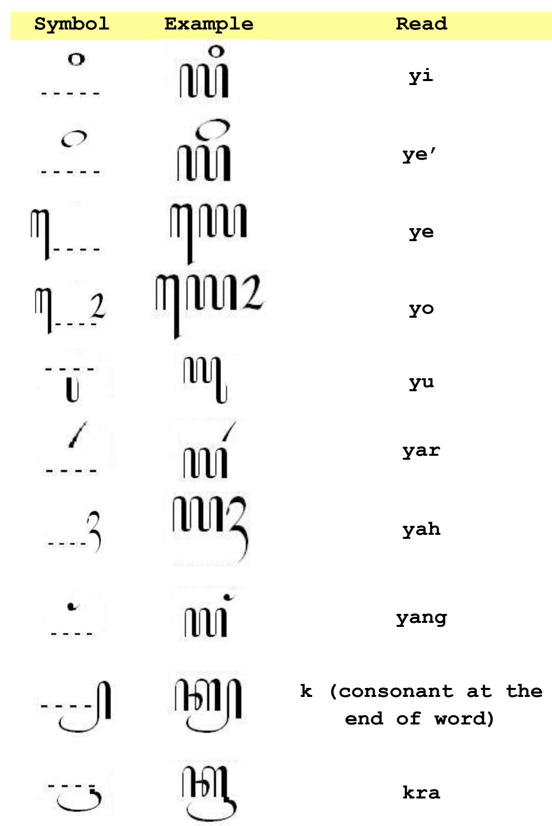
To increase interest in the use of Javanese language, can take advantage of information technology. In the research that has been done, we have developed system to recognize Javanese characters with the aim to facilitate learning the Javanese character [1,2,3,4]. In addition to the recognition of Javanese character system, it can also develop text to speech application that can be used to learn the pronunciation of the Javanese language. In this paper we will discuss the design for Javanese text to speech application. In this design, the finite state automata method will be used to create rules for classifying and declaring syllables into the smallest syllables in Javanese language. These rules later will be used when implementing text to speech applications.

II. Javanese Language

Javanese language has a special form of characters called the Javanese characters. The Javanese characters consist of 20 basic letters called carakan, can be seen in the Figure 1 [5].

Figure 1: Basic Javanese characters

In addition to the basic characters, the Javanese character has supplementary characters, consist of symbols for expressing vowels as well as a combination of two specific consonants. This supplementary characters is called *sandhangan* and can be seen in Figure 2 [5].



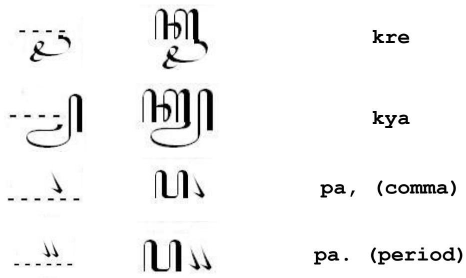


Figure 2: *Sandhangan* characters

In addition to the basic characters and *sandhangan*, Javanese characters have another symbol to describe a consonant that terminates a syllable, called *pasangan*. For example the word sam-pun, in word sam, the last letter is m. Basic character always followed by the vowel, so to declare the consonant at the end, *pasangan* is used, so the character is become: basic character ‘sa’, basic character ‘ma’, *pasangan* ‘pa’, *sandhangan* ‘u’, basic character ‘na’ and *sandhangan* for consonant at the end of the word. Each basic character will have its *pasangan*. This *pasangan* can be seen in Figure 3 [5].

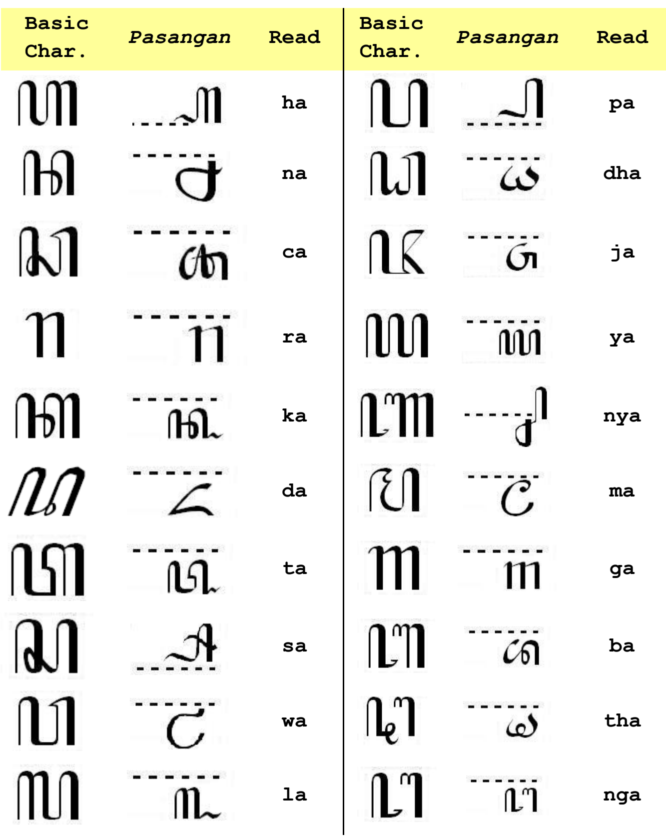


Figure 3: *Pasangan* characters

For the pronunciation of the Java language, there are several rules used [6].

If in the middle of the word there is a consonant between two vowels, the separation is done before the consonant. Example: ba-pak, pe-lem, pi-tik

If in the middle of the word there is a combination of consonant that symbolize a consonant phoneme, the combination consonants are not separated so that the separation is done before or after the combination consonants. Example: bang-sa, ba-nyak, ba-thok, go-dhong.

If in the middle of the word there are two consecutive consonants which are not special combination, the separation is done between the two consonants. Example: mum-pung, pan-ti, sir-na

If in the middle of the word there are two consonants in sequence and special combination, those consonants are not separated. Example: ka-wruh, ke-plok, mi-tra

If in the middle of the word there are three consonants and not a special combination, the separation is done between the first and second consonants. Example: am-byur, gam-blang, tin-trim.

III. AUTOMATA AND LANGUAGE THEORY

Automata is an abstract machine that can recognize, accept or generate a sentence in a particular language. Automata consist of a number of finite states, where states express information about the input [7].

The language in the dictionary is a system that includes the expression of ideas, facts, concepts, including a set of symbols and rules to perform manipulation. Language can also be called a series of symbols that has meaning.

The input of automata is considered a language that must be recognized by the machine. After input then the machine will determine whether the input is acceptable or unacceptable.

In addition to the above understanding of automata is also a system consisting of a number of finite states that learn about the abstract engine that receives input and output in discrete form that called finite state automata. Finite state automata is an automated machine of the regular language. A finite state automata has a finite state, and can move from a state to another state [7]. Formally finite state automata is expressed by 5 tuples or M = (Q, Σ, δ, S, F), where:

Q = the set of states / positions
Σ = set of input / alphabet symbols
δ = transition function
S = initial state
F = set of final states

Finite state automata which has exactly one subsequent state for each input is called deterministic finite automata. But if finite state automata has input that may generate more than one next state is called non-deterministic finite automata [8]. Example of deterministic finite automata can be seen in Figure 4.

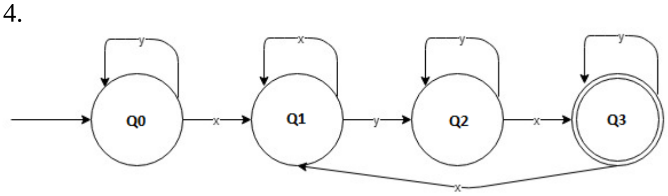


Figure 4: Example of deterministic finite automata

In Figure 4, each input has only one subsequent state, and it can be expressed as:

Q = {Q0, Q1, Q2, Q3}
Σ = {x,y}
S = Q0
F = {Q3}

And the transition functions are as follows:

δ (Q0,x) = Q1
δ (Q0,y) = Q0
δ (Q1,x) = Q1
δ (Q1,y) = Q2
δ (Q2,x) = Q3
δ (Q2,y) = Q2
δ (Q3,x) = Q1
δ (Q3,y) = Q3

The example of non-deterministic finite automata can be seen in Figure 5.

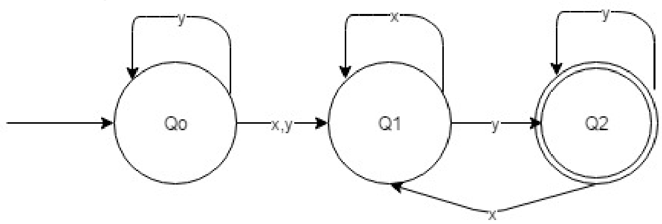


Figure 5: Example of non-deterministic finite automata

In Figure 5, it can be expressed as:
Q = {Q0, Q1, Q2}
Σ = {x,y}
S = Q0
F = {Q2}
And the transition functions are as follows:
δ (Q0,x) = {Q1}
δ (Q0,y) = {Q0,Q1}
δ (Q1,x) = {Q1}
δ (Q1,y) = {Q2}
δ (Q2,x) = {Q1}
δ (Q2,y) = {Q2}

The difference between non-deterministic finite automata and deterministic finite automata is in non-deterministic finite automata, an input may have more than one subsequent state, as in Figure 5, the transition function Q0 with input y has two subsequent states: Q0, Q1.

IV. Finite State Automata Design

To be able to create text to speech applications, first needed the separation of syllables. Those syllables that have been separated will be voiced. To perform syllable separation, the design is done using finite state automata. The design results can be seen in Figure 6.

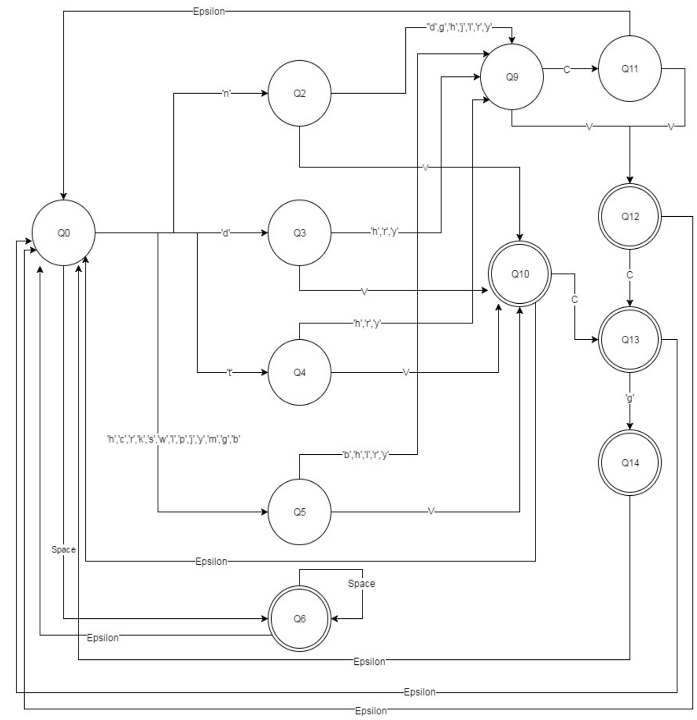


Figure 6: Design of syllables separation

In the design in Figure 6, the character V represents vowel, the character C represents consonant, while the lowercase character enclosed in quotation marks are input characters. From the design it can be seen that the input come to the initial state Q0, will split into four subsequent states. State Q2 will be selected if input is character n, Q3 to be selected if input is character d, Q4 is selected if input is character t and Q5 is selected if input is character h, c, r, k, s, w, l, p, j, y, m, g, or b. This selection is based on the Javanese basic characters, *sandhangan* and *pasangan* in Figure 1, Figure 2 and Figure 3. Especially for characters n, d, and t, in the Javanese basic characters there is more than one basic character with the same prefix characters, which are na and nga, da and dha, also ta and tha. So the design for those characters are separated on its own state.

The final states are Q6, Q10, Q12, Q13 and Q14.
Q6 declares the final state if there is a space as input.
Q10 declares the final state if syllables are composed of one consonant and one vowel.
Q12 declares the final state if syllables are composed of two consonants followed by one vowel or three consonants followed by one vowel.
Q13 declares the final state if syllables are composed of:

- one consonant, one vowel and one consonant
- two consonants, one vowel and one consonant, or
- three consonants, one vowel and one consonant

Q14 declares the final state if syllables are composed of:

- one consonant, one vowel and two consonants
- two consonant, one vowel and two consonants, or
- three consonant, one vowel and two consonants

When it reaches the final state, it will return to the initial state to find the next syllable.

V. EXPERIMENTAL RESULTS

In the experiment, we try to separate the syllables from some words and see if they can reach the final states according to the design that has been created. The experimental results of the syllables separation with their final states can be seen in Table 1.

Table 1
The Result of Syllables Separation

Word	Syllable separation	Final state
ragil	ra-gil	Q10,Q13
bumpet	bum-pet	Q13,Q13
tetabuhan	te-ta-bu-han	Q10,Q10,Q10,Q13
ngetikake	nge-ti-ka-ke	Q12,Q10,Q10,Q10
nyuwara	nyu-wa-ra	Q12,Q10,Q10
bureng	bu-reng	Q10,Q14
ngumpluk	ngum-pluk	Q12,Q12
patamanan	pa-ta-ma-nan	Q10,Q10,Q10,Q13
turu	tu-ru	Q10,Q10
buthek	bu-thek	Q10,Q13
dhamprat	dham-prat	Q13,Q13
cadhangan	ca-dha-ngan	Q10,Q12,Q13
dipenggak	di-peng-gak	Q10,Q14,Q13
bludhasbludhus	blu-dhas-blu-dhus	Q12,Q13,Q12,Q13
sumelang	su-me-lang	Q10,Q10,Q14
reged	re-ged	Q10,Q13
kecemplung	ke-cem-plung	Q10,Q13,Q14
nyemplungake	nyem-plu-nga-ke	Q13,Q12,Q12,Q10
ditegke	di-teg-ke	Q10,Q13,Q10
punggowo	pung-go-wo	Q14,Q10,Q10
ngawula	nga-wu-la	Q12,Q10,Q10
lengganan	leng-ga-nan	Q14,Q10,Q13
bekakas	be-ka-kas	Q10,Q10,Q13
prabot	pra-bot	Q12,Q13
mborok	mbo-rok	Q12,Q13
nyekarep	nye-ka-rep	Q12,Q10,Q13
nggatekake	ngga-te-ka-ke	Q12,Q10,Q10,Q10
nyangking	nyang-king	Q14,Q14
nglimput	nglim-put	Q13,Q13
tanggungan	tang-gu-ngan	Q14,Q10,Q14
ngumbah	ngum-bah	Q13,Q13
ngrukunake	ngru-ku-na-ke	Q12,Q10,Q10,Q10
nglanglang	nglang-lang	Q14,Q14
pendhapa	pen-dha-pa	Q13,Q12,Q10
nggirahi	nggi-ra-hi	Q12,Q10,Q10
sungu	su-ngu	Q10,Q12
ndongakake	ndo-nga-ka-ke	Q12,Q12,Q10,Q10
ndhisiki	ndhi-si-ki	Q12,Q10,Q10
ngapusi	nga-pu-si	Q12,Q10,Q10
mangkat	mang-kat	Q14,Q13
jenenge	je-ne-nge	Q10,Q10,Q12
mulya	mul-ya	Q13,Q10
pangastuti	pa-ngas-tu-ti	Q10,Q13,Q10,Q10
jayadiningrat	ja-ya-di-ning-rat	Q10,Q10,Q10,Q14,Q13
lebur	le-bur	Q10,Q13
manungsa	ma-nung-sa	Q10,Q14,Q10
ngunduh	ngun-duh	Q13,Q13
cedhak	ce-dhak	Q10,Q13
lakumu	la-ku-mu	Q10,Q10,Q10
pakarti	pa-kar-ti	Q10,Q13,Q10
sliramu	sli-ra-mu	Q12,Q10,Q10
kawula	ka-wu-la	Q10,Q10,Q10
kulawarga	ku-la-war-ga	Q10,Q10,Q13,Q10
mangerteni	ma-nger-te-ni	Q10,Q13,Q10,Q10
senggolan	seng-go-lan	Q14,Q10,Q13
mangesthi	ma-nges-thi	Q10,Q13,Q12
mangastuti	ma-ngas-tu-ti	Q10,Q13,Q10,Q10
gegandhengan	ge-gan-dhe-ngan	Q10,Q13,Q12,Q13
nganggo	ngang-go	Q14,Q10
rumangsa	ru-mang-sa	Q10,Q14,Q10
kersaning	ker-sa-ning	Q13,Q10,Q14
bebarengan	be-ba-re-ngan	Q10,Q10,Q10,Q13
tresno	tres-no	Q13,Q10
nyesek	nye-sek	Q12,Q13
ku-wi	ku-wi	Q10,Q10

From the experimental results in Table 1, there has been a separation of syllables with many variations of consonants and vowels. For all combinations, separation can be done in accordance with the design that has been made. The combinations include:

- One consonant and one vowel (CV)
- Two consonants and one vowel (CCV)
- Three consonants and one vowel (CCCV)
- One consonant, one vowel and one consonant (CVC),
- Two consonants, one vowel and one consonant (CCVC)
- Three consonants, one vowel and one consonant (CCCVC)
- One consonant, one vowel and two consonants (CVCC)
- Two consonants, one vowel and two consonants (CCVCC)
- Three consonants, one vowel and two consonants (CCCVCC)

VI. Conclusion

In this research we have discussed the design for Javanese text to speech application. From the experimental results, the designs that have been created to separate the syllables of Javanese language, can separate the syllables according to existing rules. For further development, this design will be used as rules when implementing the text to speech application.

Acknowledgment

This research was funded by DIPA Directorate General of Research and Development Reinforcement (Direktorat Jenderal Penguatan Riset dan Pengembangan) no. SP DIPA-042.06.1.401516/2017, fiscal year 2017. We also thank Ivan Aldinata for his help in this research.

References

[1] Budhi, G.S., Adipranata, R., "Handwritten javanese character recognition using several artificial neural network methods," Journal of ICT Research and Applications vol.8, no.3, pp. 195-212, 2015

[2] Adipranata, R., Liliana, Indrawijaya, M., Budhi, G.S., "Feature extraction for Java character recognition," Communications in Computer and Information Science 516, pp. 278-288, 2015

[3] Budhi, G.S., Adipranata, R., "Comparison of bidirectional associative memory, counterpropagation and evolutionary neural network for Java characters recognition," in proceeding of International Conference on Advanced Informatics: Concept, Theory and Application, ICAICTA, 2014

[4] Budhi, G.S., Adipranata, R., "Java characters recognition using evolutionary neural network and combination of Chi2 and backpropagation neural network," International Journal of Applied Engineering Research vol.9, no.22, pp. 18025-18036, 2014

[5] Daryanto, "Kawruh basa jawa pepak," Surabaya: Apollo, 1999

[6] Balai Bahasa Yogyakarta, "Pedoman umum ejaan bahasa Jawa huruf Latin yang disempurnakan (General guidelines for improved Javanese language using Roman characters)," Yogyakarta: Kanisius, 2006

[7] Kandar, S., "Introduction to Automata Theory, Formal Languages and Computation," Pearson Education India, 2016

[8] Ghosh, D., "Introduction To Theory Of Automata, Formal Languages, And Computation," Phi Learning, 2013