

Computer speech recognition to text for recite Holy Quran

Y A Gerhana^{1*}, A R Atmadja², D S Maylawati², A Rahman¹, K Nufus¹, H Qodim³, Busr³ and M A Ramdhani¹

¹Departement of Informatics, UIN Sunan Gunung Djati Bandung, Jalan A H Nasution 105 Bandung 40614, Indonesia

²Departement of Informatics, Sekolah Tinggi Teknologi Garut, Jalan Mayor Syamsu No 1 Tarogong Kidul Kabupaten Garut 44151, Indonesia

³Faculty of Ushuluddin, UIN Sunan Gunung Djati Bandung, Jalan A H Nasution 105 Bandung 40614, Indonesia

*yanagerhana@uinsgd.ac.id

Abstract. Memorizing Holy Quran or *Tahfidz* is important to worship for Muslim around the world. This research proposed a solution in memorizing and learning Holy Quran easily. To help in remembering the sentence of Holy Quran, Fisher-Yates Shuffle had implemented for randomization of the letter of the Holy Quran. In this research, the sound of Holy Quran had recorded and it was converted into Arabic text to recognize the character of text. Jaro-Winkler was used for text matching algorithm, and Google Speech API help to define speech recognition. The result showed that Fisher-Yates Shuffle Algorithm was successfully applied in randomization with 15 times of experiments. And also, Jaro-Winkler Distance algorithm had performed well as text matching between text from speech recognition and Holy Quran text. The result showed that the percentage of accuracy was around 91% and an average of matching time was 1.9 ms.

1. Introduction

The Qur'an is a Muslim holy book that guides life [1][2]. Understanding the content of the Qur'an is the obligation of all Muslims, including reading and memorizing it. A preliminary survey of 28 respondents with an age range of 19-21 years, proves that 35% of respondents admitted to being lazy to memorize the Qur'an, 35% admitted difficulty in memorizing the Qur'an, and 21% of respondents admitted too much to memorize. In addition, there is a concern that how to memorize the Qur'an is not only an obligation but also an easy and fun thing.

The development of mobile technology such as Android operating system that much in demand by the world community, where the operating system is installed in a mobile so that the user can also be used applications anywhere and anytime [3], [4]. The content required by the user can be accessed easily. Equipped with multimedia facilities that are able to combine text, images and audio in one device. This multimedia system displays and combine texts, sounds, pictures, animation, audio, and videos, implemented using a tool and link to enable users to navigate, interact, work, and communicate [5]. In education, multimedia can be used as instructional media or personal learning media [6]. The use of multimedia for instructions has been proven to create fun learning environments [7], improve learning motivation [8], enhance learning effectiveness [9], enhance understanding [10], promote student-centred



instruction, become an efficient investment of learning aid [11]. Android mobile technology sophistication is one way that can be used to facilitate in memorizing the Qur'an.

In this research, we build Android mobile app that helps memorize Al-Qur'an on Juz 30 with concept of connection of verse which utilizes speech recognition method. There was several previous research about reciting Al-Qur'an that served as a medium of learning Al-Qur'an [12] and evaluate of Al-Qur'an recitation [13]. Android had proven had a good capability in data processing and information display [5], [14]. Android has an Automatic Speech Recognition (ASR) that serves to utilize sound as an input tool in doing a process. ASR consists of two kinds of Speech To Text that converts sound into text and Text to Speech converts text into sound [15], [16]. The use of ASR application itself can be used to perform word search process, search location, and perform a command on android device [17].

The application is built with the concept of Speech To Text that converts the sound into text, the text will be matched using Jaro Winkler Distance Algorithm. The Jaro Winkler Distance algorithm is a variant of Jaro distance metric, an algorithm for measuring the similarity between two strings [18]. The higher Jaro- Winkler Distance for the two strings will be more similar to the string. This algorithm has a good accuracy in matching a relatively short string [19], [20]. In the application also be given a method of randomization problem using the Fisher-Yates Shuffle algorithm which is one of the good randomization method [21], [22]. The advantages of the Fisher-Yates Algorithm are the effectiveness of its randomization method and its optimal algorithmic complexity.

2. Speech to text

Speech to text is one of ASR type where speech recognition is how a computer can recognize or detect the information from sound data [15], [16], [23]. There are four phase in speech recognition, among others [24]: acceptance of input data; extraction, save input data into data storage as well as making the database for the template; comparison / matching, ie the matching stage of new data with sound data (matching grammar) on the template; and validate user identity.

Speech to text is a development of techniques and systems that allow computers or other smart devices to receive input in the form of spoken words [25], [26]. This technology allows the device to understand spoken words using word digitization and equate the digital signal with databases stored in a device or server. The spoken words are identified and then converted into digital signals by converting sound waves into a given set of codes which are then used in identifying those words or phrases. The results of identification of spoken words or sentences are then displayed in the form of text or strings that can be read by technology devices [27].

Google API provides the Google Speech Recognition API that allows developers to convert speech into a text. To be able to use these features, Android developers can use the interface and classes that have been provided. There are interface and classes that can be used for speech recognition. For interface, Recognition Listener is used to get notices from SpeechRecognizer when the related recognition event occurs. And for classes, there are RecognitionService, this class provides the base class for the implementation of the introduction service; RecognitionService.Callback, this class receives a callback from the google voice recognition feature and sends it to the user; RecognizerIntent, a constant to support speech recognition through starting intent; RecognizerResultsIntent, the constants for intents relating to the result show the voice recognition; and SpeechRecognizer, this class provides access to voice recognition services.

3. Fisher-Yates shuffle algorithm

The Fisher-Yates Shuffle algorithm has a simple process that used to generate or obtain a random permutation of numbers 1 through n! [22]. The process of Fisher-Yates Shuffle algorithm is as follows:

- Write the numbers from 1 to N
- Select a random number K between 1 to the number of n numbers that have not been crossed out
- Calculated from below, stroke the un-crossed K number, and write the number elsewhere

- Repeat steps 2 and 3 until all the figures are crossed out
- The sequence of numbers written in step 3 is a random permutation of the starting number.

In the new version (modern) the selected number is not crossed out but exchanged its position to the last digit of the number that has not been selected. “Range” is the number of numbers that have not been selected, “roll” is the selected random number, “scratch” is a list of numbers that have not been selected, the result is the result of permutations that will be obtained. The modern version was introduced because it is more optimal than the original version. The process is the last number that will be moved to the digits that are pulled out and change the numbers drawn out to a final ending number that is not withdrawn for each withdrawal and continues for the next iteration.

4. Jaro-Winkler Distance algorithm

Jaro-Winkler Distance is an algorithm for measuring similarities between two strings or text, many algorithms are used in duplicate detection [19] [28]. Jaro-Winkler algorithm is the best distance and suitable for use in short string comparisons such as person names. Normal scores such as 0 indicate there is no similarity, and 1 is exactly the same. The basic steps of the algorithm are divided into three parts, among others: calculating the length of the string; determining the same number of characters in two strings; and find the number of transpositions. In the Jaro algorithm use the formula below to calculate the distance (dm) between two strings namely S1 and S2:

$$dj = \frac{1}{3} \times \left(\frac{m}{|S_1|} + \frac{m}{|S_2|} + \frac{m - t}{|m|} \right)$$

Where, m is total of characters that equal, $|S_1|$ is length of String 1, $|S_2|$ is length of String 2, and t is total of transpositions. The theoretical distance of two equal characters can be justified if not exceeded of $\left(\frac{\max(|S_1|, |S_2|)}{s} \right) - 1$

But, refer to the value to be generated by the Jaro-Winkler Distance algorithm the distance value is 1 which signifies the similarity of strings are one hundred percent. Usually, string 1 is used as a reference for the sequence in the search for transposition. Transposition is the same character of the two strings that are compared but swapped in order. For example, in CRATE with the word TRACE, when you can see all the characters in string 1 exist and are the same as those in string 2, but in different order. By replacing C and T, we can see CRATE's word change to TRACE. The Jaro-Winkler distance uses a prefix scale (p) that provides a higher level of judgment, and a prefix length (l) that specifies the length of the prefix that is the same character length of the string compared to the discovery of the inequality. When the S1 and S2 strings are compared, the Jaro-Winkler distance (dw) is:

$$d_w = d_j + (l_p(1 - d_j))$$

Where, dw is Jaro distance for S1 and S2, l is the length of the common prefix at the beginning of the string is a maximum of 4 characters (the same character length before the max discovery 4), and p is constant scaling factor. The standard value of this constant according to Winkler is $p = 0.1$

5. Analysis of speech recognition to text for Recite Holy Quran

In this research, we have master data of Al-Quran verse data, the data is taken from Juz 30, among others: *An- Naba, An-Nazi'aat, 'Abasa, At-Takwir, Al-Infithar, Al- Muthaffifin, Al-Insyiqaaq, Al-Buruuj, Ath-Thaariq, Al-'Ala, Al-Ghasyiyah, Al-Fajr, Al-Balad, Asy-Syams, Al- Lail, Add-Dhuha, Al-Inshirah, At-Tiin, Al-'Alaq, Al- Qadr, Al-Bayyinah, Az-Zalzalah, Al-'Aadiyaat, Al- Qaari'ah, At-Takaatsur, Al-'Ashr, Al-Humazah, Al-Fiil, Quraisy, Al-Maa'uun, Al-Kautsar, Al-Kaafiruun, An- Nasr, Al-Lahab, Al-Ikhlash, Al-Falaq, and An-Naas*. The process of this research is shown in figure 1.

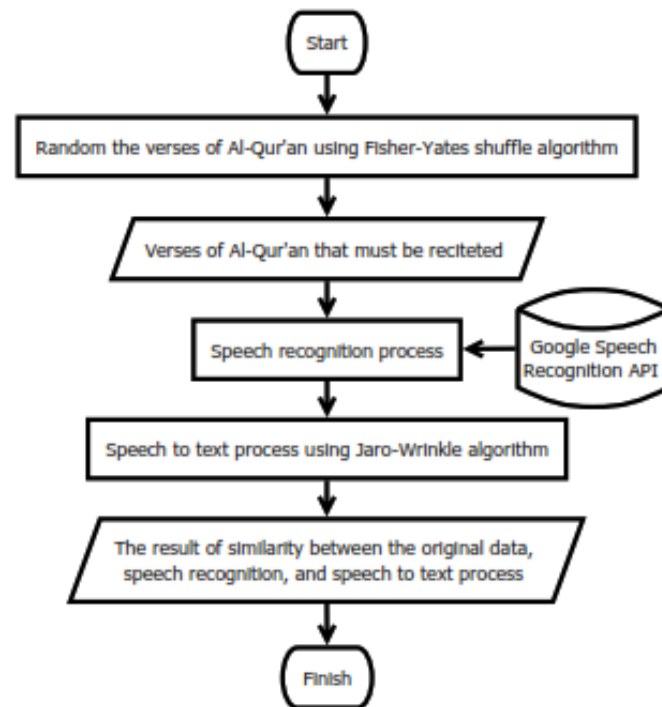


Figure 1. Flowchart of computer speech recognition to text for reciting Al-Qur'an.

Fisher-Yates shuffle algorithm is used to random verses in Jus 30 of Al-Quran. Table 1 is described the example of Fisher-Yates shuffle algorithm. First, take a random number, for example, the randomly drawn number is 4, then moved to the back position and the number 8 is moved to the origin position number 4. Then, do a random retrieval, egg number 3, move the position before the number 4, and move the number 7 to the origin position number 3, as well as the numbers 1 and 6. Do the randomization until the numbers are no longer exchangeable. When all of the numbers have been randomized, the random process is stopped.

Table 1. The example of Fisher-Yates shuffle algorithm process.

Range	Roll	Scratch	Result
		12345678	
1-8	4	123 8 567	4
1-7	3	127856	34
1-6	1	6 2785	134
1-5	2	6 578	2134
1-4	4	6 57	82134
1-3	1	7 5	682134
1-2	1	5	7682134
			57682134

The verses of Al-Qur'an that must be recited are the result of Fisher-Yates shuffle algorithm process. Then, those verses are recited and the system will process speech recognition of the sound data of Al-Qur'an recitation. Next, the Jaro-Winkler distance algorithm is used to measure the similarity between two strings. The text of a speech to text results will be matched with this, the Jaro-Winkler distance algorithm has a maximum distance value of 1 which denotes the string equation. Usually, String 1 is used as a reference for sequences in search of transposition. For example, from text data of verses in table 2 are known that: $m = 15$, $l = 4$, $|S1| = 15$, $p = 0.1$, $|S2| = 15$, and $t = 0$.

Table 2. The example of verses of Al-Qur'an.

No	Name and Verses of Al-Quran	Text Sample
1	<i>Al-Kautsar</i> : 3	اش ن ا رنبلاو دهكؤن
2	<i>Al-Kautsar</i> : 3	رنبل او دهكؤن اش ن ا

$$dj = \frac{1}{3} \times \left(\frac{m}{|S_1|} + \frac{m}{|S_2|} + \frac{m-t}{|m|} \right)$$

$$dj = \frac{1}{3} \times \left(\frac{15}{|15|} + \frac{15}{|15|} + \frac{15-0}{|15|} \right)$$

$$dj = 1$$

$$d_w = d_j + (l_p(1 - d_j))$$

$$d_w = 1 + (4(0,1)(-1))$$

$$d_w = 1$$

The correct or wrong answer is determined by the matching value of the speech to text results with the text in the database. In this case, when the Jaro-Winkler value is 1 then the answer is true, and less than one answer is wrong.

6. Result and discussion

We did several experiments for this research. Besides black box testing for making sure all of the functionality of system was running well and as expected, we evaluated the Fisher-Yates shuffle algorithm to random the verses of Al-Qur'an that must be recited and Jaro-Winkler algorithm for text matching. The result of Fisher-Yates shuffle algorithm (described in table 3) with 15 times of testing experiment showed that Fisher-Yates shuffle algorithm could random the verses of Al-Qur'an well, same as previous research [22]. Testing proses for speech to text recognition and text matching with Jaro-Winkler algorithm were done with several scenarios, among others evaluate the time process of text matching (the result was described in table 4), then evaluate the accuracy between the result of speech to text process and text matching with the Jaro-Winkler algorithm. The accuracy testing was done with 15 people who recites verses of Al-Qur'an (the result was described in table 5). We also did several testing for similar letters in pronunciation and the result was shown in table 6.

From the test results with 15 different people, there are six-person pronounced clearly, but the results of speech to text did not match between the original text and the result based on Jaro-Winkler distance, because the value of distance was not worth 1. In this case, the pronunciation should be clear for better results. So, the results of speech recognition had not been accurate to recognize the recitation of the Qur'an. On the other hand, Google speech cannot recognize in distinguishing the short length of the letters Al-Quran when the testers recite *Al-Buruj*:3, which should have long pronunciation of "ha" but not read long. However, the results of speech to text came out the same as the original text, then the Jaro-Winkler distance value was 1, which means the correct answer. In addition, the testing cannot have maximal results because testing was done in a crowded place and there were many voices detected by Google.

The results of similar letters in pronunciation, for the 7 letters that have similarities in pronunciation, Google speech API had not been able to in detail distinguish the Arabic letters in the verses of Al-Quran. So it was necessary to have another method or algorithm to further refine this Tahmid application in distinguishing the Arabic letters in the verses of Al-Quran.

Table 3. The result of Fisher-Yates shuffle algorithm.

Testing No.	Result of Randomization
1	<i>Al-Falaq : 3-4, At-Tiin : 5-6, Al-Qodr : 3-4, Al-Zalzalah : 3-4 Al-Kautsar : 1-2, Al-Ikhlās : 1-2, Al-Kafirun : 1-2, Al-Lahab : 3-4, Al-Fiil : 1-2, An-Nas : 1-2, Al-Adiyat : 6-7, Al-Humazah : 6-7, Al-Kafirun : 2-3, Al-Alaq : 1-2, Al-Humazah : 4-5</i>
2	<i>Al-Kautsar : 1-2, Al-Zalzalah : 3-4, Al-Fiil : 1-2, Al-Alaq : 1-2, Al-Falaq : 3-4, Al-Humazah : 6-7, Al-Adiyat : 6-7, Al-Humazah : 4-5, Al-Lahab : 3-4, At-Tiin : 5-6, An-Nas : 1-2, Al-Qodr : 3-4, Al-Kafirun : 1-2, Al-Kafirun : 2-3, Al-Ikhlās : 1-2</i>
3	<i>An-Nas : 1-2, Al-Adiyat : 6-7, At-Tiin : 5-6, Al-Kafirun : 1-2, Al-Alaq : 1-2, Al-Ikhlās : 1-2, Al-Humazah : 6-7, Al-Zalzalah : 3-4, Al-Humazah : 4-5, Al-Qodr : 3-4, Al-Kafirun : 2-3, Al-Fiil : 1-2, Al-Falaq : 3-4, Al-Kautsar : 1-2, Al-Lahab : 3-4</i>
4	<i>Al-Falaq : 3-4, Al-Humazah : 4-5, Al-Adiyat : 6-7, An-Nas : 1-2, Al-Kautsar : 1-2, Al-Alaq : 1-2, Al-Kafirun : 1-2, Al-Fiil : 1-2, Al-Kafirun : 2-3, Al-Qodr : 3-4, Al-Ikhlās : 1-2, Al-Humazah : 6-7, At-Tiin : 5-6, Al-Lahab : 3-4, Al-Zalzalah : 3-4</i>
5	<i>Al-Qodr : 3-4, Al-Ikhlās : 1-2, At-Tiin : 5-6, Al-Falaq : 3-4, Al-Humazah : 6-7, Al-Kafirun : 2-3, An-Nas : 1-2, Al-Lahab : 3-4, Al-Alaq : 1-2, Al-Kafirun : 1-2, Al-Kautsar : 1-2, Al-Zalzalah : 3-4, Al-Fiil : 1-2, Al-Humazah : 4-5, Al-Adiyat : 6-7</i>
6	<i>Al-Lahab : 3-4, Al-Adiyat : 6-7, Al-Humazah : 6-7, Al-Alaq : 1-2, Al-Kautsar : 1-2, Al-Falaq : 3-4, Al-Kafirun : 1-2, Al-Fiil : 1-2, Al-Zalzalah : 3-4, Al-Ikhlās : 1-2, Al-Qodr : 3-4, At-Tiin : 5-6, An-Nas : 1-2, Al-Kafirun : 2-3, Al-Humazah : 4-5</i>
7	<i>Al-Qodr : 3-4, Al-Humazah : 4-5, Al-Ikhlās : 1-2, Al-Kautsar : 1-2, An-Nas : 1-2, Al-Kafirun : 1-2, Al-Falaq : 3-4, At-Tiin : 5-6, Al-Alaq : 1-2, Al-Adiyat : 6-7, Al-Kafirun : 2-3, Al-Lahab : 3-4, Al-Zalzalah : 3-4, Al-Humazah : 6-7, Al-Fiil : 1-2</i>
8	<i>Al-Kafirun : 2-3, Al-Adiyat : 6-7, Al-Humazah : 6-7, Al-Zalzalah : 3-4, An-Nas : 1-2, Al-Qodr : 3-4, Al-Alaq : 1-2, Al-Humazah : 4-5, Al-Fiil : 1-2, Al-Lahab : 3-4, Al-Kautsar : 1-2, Al-Falaq : 3-4, Al-Ikhlās : 1-2, At-Tiin : 5-6, Al-Kafirun : 1-2</i>
9	<i>Al-Kautsar : 1-2, Al-Qodr : 3-4, At-Tiin : 5-6, Al-Ikhlās : 1-2, Al-Adiyat : 6-7, Al-Falaq : 3-4, Al-Humazah : 4-5</i>
10	<i>Al-Alaq : 1-2, Al-Kafirun : 1-2, Al-Humazah : 6-7, Al-Lahab : 3-4, Al-Zalzalah : 3-4, An-Nas : 1-2, Al-Kafirun : 2-3, Al-Fiil : 1-2</i>
11	<i>Al-Humazah : 4-5, Al-Zalzalah : 3-4, Al-Adiyat : 6-7, An-Nas : 1-2, Al-Falaq : 3-4, Al-Fiil : 1-2, At-Tiin : 5-6, Al-Qodr : 3-4, Al-Alaq : 1-2, Al-Lahab : 3-4, Al-Kafirun : 1-2, Al-Kafirun : 2-3, Al-Kautsar : 1-2, Al-Humazah : 6-7, Al-Ikhlās : 1-2</i>
12	<i>Al-Falaq : 3-4, Al-Humazah : 4-5, Al-Qodr : 3-4, At-Tiin : 5-6, Al-Ikhlās : 1-2, Al-Adiyat : 6-7, Al-Fiil : 1-2, Al-Humazah : 6-7, Al-Zalzalah : 3-4, Al-Lahab : 3-4, An-Nas : 1-2, Al-Kafirun : 2-3, Al-Kafirun : 1-2, Al-Kautsar : 1-2, Al-Alaq : 1-2</i>
13	<i>Al-Kautsar : 1-2, Al-Fiil : 1-2, Al-Qodr : 3-4, Al-Lahab : 3-4, Al-Ikhlās : 1-2, Al-Adiyat : 6-7, An-Nas : 1-2, Al-Humazah : 4-5, Al-Kafirun : 1-2, At-Tiin : 5-6, Al-Zalzalah : 3-4, Al-Humazah : 6-7, Al-Kafirun : 2-3, Al-Alaq : 1-2, Al-Falaq : 3-4</i>
14	<i>Al-Lahab : 3-4, Al-Falaq : 3-4, Al-Alaq : 1-2, Al-Kafirun : 1-2, Al-Humazah : 6-7, An-Nas : 1-2, At-Tiin : 5-6, Al-Fiil : 1-2, Al-Ikhlās : 1-2, Al-Qodr : 3-4, Al-Adiyat : 6-7, Al-Kafirun : 2-3, Al-Zalzalah : 3-4, Al-Humazah : 4-5, Al-Kautsar : 1-2</i>
15	<i>Al-Zalzalah : 3-4, Al-Lahab : 3-4, Al-Humazah : 4-5, Al-Kafirun : 2-3, Al-Kautsar : 1-2, An-Nas : 1-2, Al-Fiil : 1-2, Al-Adiyat : 6-7, Al-Humazah : 6-7, At-Tiin : 5-6, Al-Kafirun : 1-2, Al-Qodr : 3-4, Al-Falaq : 3-4, Al-Ikhlās : 1-2, Al-Alaq : 1-2</i>

Table 4. The result of time process of Jaro-Winkler distance algorithm.

No.	Text 1	Text 2	Value of Jaro-Winkler	Time Process
1	نیملا دلیلا اذه و	نیملا دلیلا اذه و	1.0	1ms
2	اهلام ناسنلا لاقو	اهلام ناسنلا لاقو	1.0	0ms
3	دلوي مل و دلي مل	دلوي مل و ديل مل	0.97	1ms
4	رتبلا وه کنناش نا	رتبلا وه کنناش ان	0.97	2ms
5	رجفلا علطم نتح يه ملاس	رجفلا علطم نتح يه ملاس	1.0	1ms
6	اهلام ناسنلا لاقو	اهلام ناسنلا لاقو	1.0	2ms
7	هدصوم مهيلع اهنا	هدصوم مهيلع اهان	0.97	4ms
8	دبك يف ناسنلا انقلخ دقل	دبك يف ناسنلا انقلخ دقل	1.0	3ms
9	دو هس نينموملاب نولعفيام بلع مهو	دو هس نينموملاب نولعفيام بلع مهو	1.0	3ms
10	اباسح عاطع كبر نم عاج	اباسح عاطع كبر نم عاج	1.0	3ms
Average of Time Process			0.99	1.9ms

Table 5. The Result of Accuracy Testing between Speech to Text and Text Matching with Jaro-Winkler.

Person	Original Text	Result of Speech to Text	Value of Jaro-Winkler	Percentage
Person 1	رتبلا وه کنناش نا	رتبلا وه کنناش ان	0,97	97%
Person 2	ميرك لوسر لوقل هنا	ميرك لوسر لوقل هنا	1.0	100%
Person 3	نیملا دلیلا اذه و	نیملا دلیلا اذه و	1.0	100%
Person 4	مرکلا کيرو ارقا	مرکلا کيرو ارقا	1.0	100%
Person 5	هنيزاوم تلقت نم اماف	هنيزاوم تلقت نم اماف	1.0	100%
Person 6	رجفلا علطم نتح يه ملاس	رجفلا علطم نتح يه نم م لاس	0.89	89%
Person 7	دسم نم لبح اهديج يف	دسم نم لبح اهديج يف	1.0	100%
Person 8	رتبلا وه کنناش نا	رتبلا وه کنناش نا	1.0	100%
Person 9	دو هسوم دهاشو	دو هسوم دهاشو	1.0	100%
Person 10	رجح يذل مسق كلذ يف له	رجح يذل مسق كلذ يف له	1.0	100%
Person 11	داصر ملايل كبر نا	داصر ملايل كبر نا	1.0	100%
Person 12	داصر ملايل كبر نا	امل لاكا ثارتلا نولكاتو	0.60	60%
Person 13	تجوز سوفنلا اذاو	تجوز سوفنلا اذاو	1.0	100%
Person 14	دبك يف ناسنلا انقلخ دقل	دبك يف نينثا لوحد نوکيلا	0.68	68%
Person 15	هنيزاوم تلقت نم اماف	يتاوامس تلوق ساماما	0.65	65%
Average of Accuracy			0.91	91%

Table 6. The Result of Similar Letters in Pronunciation.

Testing No.	Letter	Original Text	Pronunciation	Result of Speech to Text
1	ش-س	رتبلا وه کنناش نا	رتبلا وه کنناس نا	رتبلا وه کنناش نا
2	ث-س	رثوکللا کانيطعا انا	رسوکللا کانيطعا انا	رثوکللا کانيطعا انا
3	ذ-د	ميتيلا عدي يذلا كلذف	ميتيلا عدي يذلا كلذف	ميتيلا عدي يذلا كلذف
4	ه-ح	ليبابا اريط مهيلع لسرا و	ليبابا اريط محيلع لسرا و	ليبابا اريط مهيلع لسرا و
5	ز-ج	اهلازلز ضرلا تلزلز اذا	اهلازلز ضرلا تل جل جاذا	اهلازلز ضرلا تلزلز اذا
6	ض-ظ	نيلاضلا لاو مهيلع بوضغلا	نيلاضلا لاو مهيلع بوضغلا	نيلاضلا لاو مهيلع بوضغلا
7	ع-ا	ريغ اسابل ليلا انلعجو	ريغ اسابل ليلا انلاجو	ريغ اسابل ليلا انلعجو

7. Conclusion

Fisher-Yates Shuffle algorithm can be applied to randomize the problem on Al-Qur'an recitation application properly and optimally. From the results of testing the emergence of the problem successfully randomized, with the emergence of different problems. Jaro-Winkler Distance algorithm has good precision in text matching. From the test results, the average time required for text matching is 1.9 ms. The speech recognition process is done using Google Speech API quite well and can be compared with the original Quran text, with 91% of accuracy. But not yet able to distinguish in detail the Arabic letters in verses of Al- Quran that have similarities in pronunciation.

In its development, this application is still far from perfect therefore needed further development, for that suggestion will greatly assist the process of research development and application in the future. The ideas or suggestions that can be used as material improvement, development or refinement of this implementation in the future include: the results of speech recognition can be processed to display the legal error of reading; applications can be used offline; not only in juz 30 but the whole of the Qur'an, and developed on other sites like iOS platform.

Acknowledgement

We say many thanks to Research and Publication Center of UIN Sunan Gunung Djati Bandung Indonesia for supporting publication of our research

References

- [1] I Riyani and Y Huriani 2017 Reinterpretasi Asbāb Al-Nuzūl bagi Penafsiran Alquran *Wawasan J. Ilm. Agama dan Sos. Budaya* **2**(1) pp. 113–130
- [2] C Slamet, A Rahman, M A Ramdhani and W Darmalaksana 2016 Clustering the Verses of the Holy Qur'an Using K-Means Algorithm *Asian J. Inf. Technol.* **15**(24) pp. 5159–5162
- [3] A Hardiansyah 2016 Designing Android Based Augmented Reality Location-Based Service Application *JOIN (Jurnal Online Inform.)* **2**(2) pp. 110–115
- [4] H Septian, E W Hidayat and A Rahmatulloh 2018 Aplikasi Pengenalan Bahasa Arab dan Inggris untuk Anak-Anak Berbasis Android *JOIN (Jurnal Online Inform.)* **2**(2) p. 71
- [5] S Sari, D M Aryana, C Z Subarkah and M A. Ramdhani 2018 Multimedia Based on Scientific Approach for Periodic System of Element *IOP Conf. Ser. Mater. Sci. Eng.* **288**(1) p. 12137
- [6] I Farida, I Helsy, I Fitriani and M A Ramdhani 2018 Learning Material of Chemistry in High School Using Multiple Representations *IOP Conf. Ser. Mater. Sci. Eng.* **288**(1) p. 12078
- [7] Aisyah R, Zakiyah I A, Farida I and Ramdhani M A 2017 Learning Crude Oil by Using Scientific Literacy Comics *In Journal of Physics: Conference Series* **895**(1) p. 012011 IOP Publishing.
- [8] Sari, F S Irwansyah, I Farida and M A Ramdhani 2017 Using Android-Based Educational Game for Learning Colloid Material Using Android-Based Educational Game for Learning Colloid Material *J. Phys. Conf. Ser.* **895**(1) p. 12012
- [9] F S Irwansyah, Y M Yusuf, I Farida and M A Ramdhani 2018 Augmented Reality (AR) Technology on the Android Operating System in Chemistry Learning *IOP Conf. Ser. Mater. Sci. Eng.* **288**(1) p. 12068
- [10] I Helsy, Maryamah, I Farida and M A Ramdhani 2017 Volta-Based Cells Materials Chemical Multiple Representation to Improve Ability of Student Representation *J. Phys. Conf. Ser.* **895**(1) p. 12010
- [11] F S Irwansyah, I Lubab, I Farida and M A Ramdhani 2017 Designing Interactive Electronic Module in Chemistry Lessons *J. Phys. Conf. Ser.* **895**(1) p. 12009
- [12] M L Fadilah 2016 Rancang bangun game sambung ayat Al-quran pada Juz 30 Menggunakan linear congruent method Berbasis mobile (Bandung: UIN Sunan Gunung Djati)
- [13] E Haryanto and R Cahyana 2005 Pengembangan aplikasi mutabaah tahfidz alquran untuk

- mengevaluasi hafalan *J. Algoritm. Sekol. Tinggi Teknol. Garut* **12**(1) pp. 1–4
- [14] F S Irwansyah, I Lubab, I Farida and M A Ramdhani 2017 Designing Interactive Electronic Module in Chemistry Lessons *J. Phys. Conf. Ser.* **895**(1) p. 12009
- [15] P Saini and P Kaur 2013 Automatic Speech Recognition: A Review *Int. J. Eng. Trends Technol.* **4**(3) pp. 132–136
- [16] J Li, L Deng, R Haeb Umbach and Y Gong 2016 Fundamentals of speech recognition in *Robust Automatic Speech Recognition 2016* pp. 9–40
- [17] Supriyanta, W Pudji and S Bakti 2014 Aplikasi Konversi Suara Ke Teks Berbasis Android Menggunakan Google Speech Api *Bianglala Inform.* **2**(2) pp. 11–19
- [18] X Li, A Guttmann, S Cipiere, L Maigne, J Demongeot, J Y Boire and L Ouchchane 2014 Implementation of an extended Fellegi-Sunter probabilistic record linkage method using the Jaro-Winkler string comparator in *2014 IEEE-EMBS International Conference on Biomedical and Health Informatics BHI 2014* pp. 375–379
- [19] A Kornain, F Yansen and T Tinaliah 2014 Penerapan Algoritma Jaro-Winkler Distance Untuk Sistem Pendeteksi Plagiarisme Pada Dokumen Teks Berbahasa Indonesia *Progr. Stud. Tek. Inform. STMIK GI MDP* pp. 1–10, 2014.
- [20] Y Rochmawati and R Kusumaningrum 2016 Studi Perbandingan Algoritma Pencarian String dalam Metode Approximate String Matching untuk Identifikasi Kesalahan Pengetikan Teks *J. Buana Inform.* **7**(2)
- [21] M A Hasan, S Supriadi and Z Zamzami 2017 Implementasi Algoritma Fisher-Yates Untuk Mengacak Soal Ujian Online Penerimaan Mahasiswa Baru (Studi Kasus : Universitas Lancang Kuning Riau) *J. Teknol. dan Sist. Inf.* **3**(2) p. 291
- [22] I Haditama, C Slamet and D Fauzy Rahman 2016 Implementasi Algoritma Fisher-Yates dan Fuzzy Tsukamoto dalam Game Kuis Tebak Nada Sunda Berbasis Android *J. Online Inform.* **1**(1) pp.51–58
- [23] A S Sunny 2009 Speech Recognition Menggunakan Algoritma Program Dinamis *Speech Recognit. menggunakan Algoritm. Progr. Din.* p. 4
- [24] G Melissa 2008 Pencocokan Pola Suara (Speech Recognition) dengan Algoritma FFT dan Divide and Conquer Makal *If2251 Strateg. Algoritm.*
- [25] S Furui, T Kikuchi, Y Shinnaka and C Hori 2004 Speech- to-text and speech-to-speech summarization of spontaneous speech in *IEEE Transactions on Speech and Audio Processing*, **12**(4) pp. 401–408
- [26] B R Reddy and E Mahender 2013 Speech to Text Conversion using Android Platform *Int. J. Eng. Res. Appl.* **3**(1) pp. 253–258
- [27] A I H Kusuma 2014 *Implementasi Speech To Text Dan Text To Speech Pada Aplikasi Traveling Conversations Assistance Berbasis Mobile* (Universitas Komputer Indonesia)
- [28] A Kurniawati, A K Sekarwati and I W S Wicaksana 2012 Arsitektur Untuk Aplikasi Deteksi Kesamaan Dokumen *Konf. Nas. Sist. Inf.* pp. 297–302