Truncated Query of Phonetic Search for Al Qur'an

Aidil Zafran[1], Moch Arif Bijaksana[2], Kemas M Lhaksmana[3]
School of Computing, Telkom University, Indonesia
[1]aidilzafran@student.telkomuniversity.ac.id, [2]arifbijaksana@telkomuniversity.ac.id, [3]kemasmuslim@telkomuniversity.ac.id

***Abstract*—This study examines how to look for a verse of the Quran or a clipped verse in the Qur'an and ranks search results correctly. Because the Qur'an consists of 30 juz, 114 letters and 6236 verses, then if you search for a particular verse then a Quranic verse search system is needed. At present, there is already a verse search system for the Quran, but the search results ranking process still has a number of errors in highlighting the verses that are searched for, the similarity scores that are less precise and have not been good at handling clipped or incomplete verses. When conducting a search that needs to remember a verse from the Koran, one can remember the whole verse of the Quran but there are also some cases that cannot remember a verse completely or cut off. For example, a complete verse search query ذَلِكَ ٱلْكِتَٰب لَا رَيْبَ فِيهِ *zalikal kitabu la raiba fihi* the result will be different from the clipped verse query like this ذَلِكَ ٱلْكِتَٰب فِيهِ *zalikal kitabu fihi*. Because some truncated characters will be ignored by the system and can reduce the value of similarity. This shows that the existing system does not provide good accuracy in string matching. With this research, it will overcome the LCS search problem which ignores previously unreadable characters. Using the LCS method of query search for all data so as to produce candidate results, then research the neglected character of the candidate results to find similarities between neglected characters and one of the candidates previously obtained. So, improve ranking in the lafzi and increase the ability of the lafzi to search for clauses that are clipped or incomplete.**

***Index Terms*—query search, weighting, ranking**

I. Introduction

Islamic scholars have described the Al-Quran as the holy book of Muslims and this book teaches moral, purification, good deeds, and as well as those forbidden by the Almighty Allah. The Al Quran provides guidance to mankind, promotes justice between one another, and provides guidance on how to live on earth. In terms of language, the Quran in Arabic has the meaning of reading or something that is read repeatedly [1]. Statistically, the Quran consists of 114 letters, 6236 verses, and 77845 words. With a large number of letters, verses, and words, manually searching for the words in the text of the Quran is difficult. Therefore, computers can be used to help search verse or piece of verse in the Quran. At present, there are many Quranic verse search applications, both Arabic versions of the Al-quran and a translated version. Most outstanding Al Quran applications only have Quran reading features based on juz and surah indices, and searches according to the words in the translation.

In this research, will improve percentage of the results used on the Lafzi as an existing system. Increasing the percentage or value of the search results can be done by making measurements of LCS values by making the LCS handle the finding of sequential characters. LCS can find the best and longest similarities in candidate search results. But the LCS does not pay attention to the sequence of characters it finds. By doing the LCS twice for the candidate results and adding the distance between the characters sought, this can make the LCS find the query characters that are far apart or opposite as long as the query character sought by the LCS is found and is continuous.

Recitation of Al Quran reading can be obtained from around, such as recitation, study and prayer. For example, how to write in the writing system that is commonly used by Indonesian, if they hear reading ذَلِكَ ٱلْكِتَٰب لَا رَيْبَ فِيهِ most people will write *zalikal kitabu la raiba fihi* because according to the habit of reading and writing Indonesian. However, when the search query is not correct, such as the writing of transliterated verses that are truncated, the results of the percentage similarity of results will be very low. Then the LCS is used the second time to the candidate results that have been obtained. It is expected that the second LCS can find the cut characters of verse or arrangement of search characters whose position is wrong.

II. Related works

This research is based on previous research which built a verse search system of the Quran based on phonetic similarities that were more in line with the representation of Indonesian pronunciation. In the previous study, a Quranic verse search system was built based on phonetic similarities that were more in line with the representation of Indonesian pronunciation [2]. For this purpose, a phonetic coding method was developed based on the matching of the Latin-Arabic script used in Indonesia as well as the similarity in the way of pronouncing the letters in the Koran. The search method used is a search with trigrams that are applied to the approved size phonetic code. Testing of the system is also done to find out how well the system can do a search. The size used to test system performance is search time and average precision.

III. Phonetic Search System

A. General description

In this study, an application will be made that has a feature to read the verses of Al-Quran based on phonetic indexes and search features that use the KNN method. In this research, an application will be made that has features to search for verses of the Koran based on phonetic indexes and search features that use the KNN algorithm. Before doing the Query

978-1-5386-8052-0/19/$31.00 ©2019 IEEE

search process, preprocessing Case Folding data is done first, Tokenisation, Stopwords and query weighting. Next, the search query that has been through preprocessing will be calculated by weighting it to the LCS to find the best candidate results from the entire data. Of the candidates obtained, a limit value is called a Threshold. The threshold will display a number of candidate search results as the final result of the search provided that it passes the threshold value of the Threshold. Furthermore, each candidate result that crosses the threshold will be displayed based on sorting the similarity value or the largest to the smallest percentage of similarity. The update in this research is to deal with the situation when a search query turns out to be a clipped Quranic verse. When searching, it will produce a low LCS value because only a few characters can be assessed by the LCS while the characters are not found because the position of the character sequence will be ignored. These neglected characters will reduce accuracy or similarity between existing search queries and databases. In the search process query, there are a number of characters that will be ignored by the LCS because of the character set that has been missed by several characters. So to deal with this, in the calculation of the first LCS characters that are not counted by the LCS will be ignored, but will be saved to be counted again against the candidates found in the previous search. For example, when the first Query search results in 10 candidates resulting from the entire data, then the second search to increase accuracy will be done on the 10 candidates that have been obtained previously. This is intended to look for the verse fragments as long as the neglected character is in the same verse

B. Flowchart sistem

An overview of the flow or process that occurs in the application being built in accordance with the needs of the system. tambahin kalimat disini

C. Dataset

The dataset used in this study came from previous related studies. In this study, the dataset used is the whole surah and juz in the Quran. The dataset is obtained from Tanzil. Data in the form of text, each line containing one verse in Arabic language and script without punctuation.

D. Query Input

Query Input is the user's input to search for Quran verses using the search feature by typing in a word or verse snippet. In search, this system uses an algorithm where the search algorithm is sequential in that it searches data sequentially starting from the first data to the data sought (key data) obtained or until all data has been searched and the key data is not found. But before the search is carried out, the query input must be preprocessed so that query can be calculated with an algorithm.

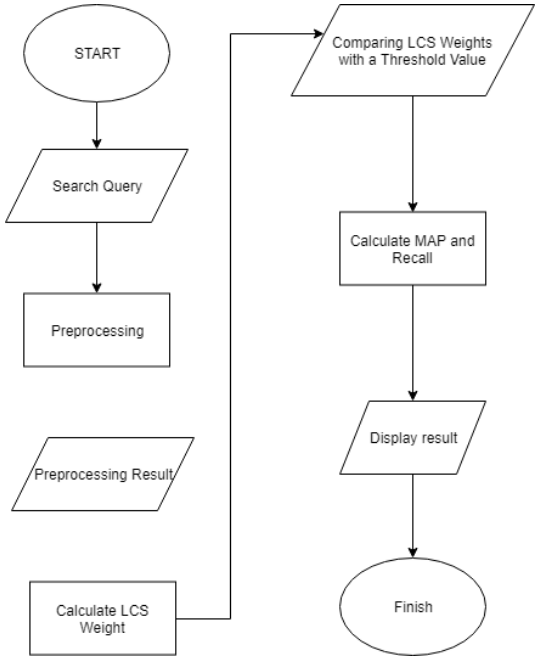

Fig. 1. General description of this work.

E. Case Folding

Case folding. This process functions to generalize letter characters from the process Query Input which is converting capital letters or uppercase letters into lowercase letters. [3].

TABLE I
EXAMPLE OF CASE FOLDING

No	Query Input	Result of Case Folding
1	*Kulhuallahuahad*	*kulhuallahuahad*
2	*BismiLLah*	*bismillah*
3	*Zalikal Kitabu La Raiba fih*	*zalikal kitabu la raiba fih*
4	*LAYUKMINUN*	*layukminun*

F. Tokenisation

Tokenisation is the process of deciphering words from a sentence. The words in the text analysis process are treated as a single entity [3].

TABLE II
EXAMPLE OF TOKENISATION

No	Query Input	Result of Tokenisation
1	*kulhuallahuahad allahusomad*	*kulhuallahuahadallahusomad*
2	*lamyalid walamyulad*	*lamyalidwalamyulad*
3	*innallazina kafaru*	*innallazinakafaru*
4	*wa ula ika humul muflihun*	*waulaikahumulmuflihun*

G. Stopwords/Filtering

A stopword in text analysis is not needed because it needs to be deleted. In this process the symbols entered not used in the analysis will be deleted. Not only conjunctions, words that don't need to be analyzed can be deleted [3].

TABLE III
EXAMPLE OF STOPWORDS/FILTERING

No	Query Input	Result of Stopwords/Filtering
1	*bismillah-hirrohman*	*bismillahirrohman*
2	*wallazinayu'minuna*	*wallazinayuminuna*
3	*wabil-akhiratihumyukinun*	*wabilakhiratihumyukinun*
4	*lamyalidwalamyulad...*	*lamyalidwalamyulad*

H. Preprocessing Results

Preprocessing results are input queries that have gone through preprocessing case folding, tokenisation and stopwords. so that input queries can be carried out trigram searches that match the database. Databases and queries that have found similarities in trigrams will then be weighted and the calculation of similarity with LCS.

I. K-Nearst Neighbor

The K-NN algorithm in this study has the task of matching query as a search keyword with a dataset as the search source *query*. The K-NN algorithm works based on the shortest distance from the query instance to the data set sample to determine some of its closest neighbors so that it can be classified as a single cluster. The distance search method used is Euclidean Distance which is the calculation of the closest distance. The closest distance calculation is needed to determine the number of similarities calculated from the similarity of the appearance of the text that has a query.

J. LCS method will be used

The intended improvement is to increase the percentage by using the LCS twice when the previous search results do not find the results or results found very much but nothing is right. LCS will be carried out to candidates results with the previous LCS method. The way to deal with cases of LCS tolerance to queries in the wrong order. Examples such as the *zalikal kitabu fihi* search query LCS found results with 73%. This is because the *fihi* character at the end of the query is not found to be similar to the LCS. Where all the search queries actually are in the same verse.

Then this research will make the calculation of the LCS a second time to find a better percentage of results. The first LCS result will be removed for each character that has been found for the first time, then it will leave an undiscovered characters. Furthermore, characters that have not been found from the search query will be calculated with the LCS for all candidates that have been found before. At Lafzi, each result found contains a complete verse, the second LCS will calculate the similarity of search queries that have not been found with the results of Lafzi.

Example of the second LCS calculates.

1) The first LCS will make the first calculation *zalikal kitabu fihi*. Found with the percentage of 73% in QS [2: 2].
2) the *fihi* character is not found in one LCS calculation because it doesn't kneel with the previous character In the actual verse, *zalikal kitabu la raiba fihi*.
3) previously found characters will be deleted leaving *fihi* undiscovered characters
4) the *fihi* character will be calculated LCS again on the results of Lafzi search, which is calculated only against QS [2: 2].
5) the *fihi* character will be found in the surah and this LCS value will be combined with the previous LCS value.

LCS will work by looking for similarities between input queries and available databases. LCS does not only function to find each character that is the same as both but looks for the same and sequential characters. So that the character obtained becomes a unity of verses that can be read, can be understood and has meaning. Because if you only find every character that is the same, it's likely that it won't be readable and has no value. Then the LCS method is used to handle the search for verses both intact and truncated.

LCS will work like this, if there is a *onetwothreefourfive* search query then the query in the database becomes the LCS calculation.

1) *onetwothree* will have to be the best solution because there are 11 same and sequential characters
2) *onetwofour* query is correct but not the best solution because there are 6 characters in the same order

This method also supports more grouped search results highlighting functions. This method will make LCS calculations and the percentage of results will be better because the queries that have been found for the first time will be combined with the second query search results. Although in this study the calculation of the similarity of queries was not taken into account, it could increase the percentage of results and improve the ranking of results in Lafzi.

TABLE IV
EXAMPLE OF LCS

No	Query Input	Result of LCS
1	*zalikal kitabu laraiba fihi*	100%
2	*zalikal kitabu fihi*	73%
3	*allazina yu minuna*	81%
4	*allazina minuna*	70%
5	*wa mimma razaknahum yunfiqun*	100%
6	*wa mimma yunfiqun*	50%

K. Ranking of Result Longest Longest Contiguous Subsequence

After giving the weight or value of the trigram similarity search results and generating search results, the system will display output based on the results of previous calculations. The system will display all process results ranging from the largest percentage of similarity to the smallest percentage of similarities. Preprocessing results will be calculated as the number of letters found as a result of the candidate using the LCS method. LCS will calculate the longest connection density and the complexity of several series of connections. The LCS calculates the density value by calculating the

distance or gap between adjacent letters from beginning to end. Furthermore, the sorting value with LCS will be calculated at the level.

L. Search result

The output of the system is in the form of verses from the Qur'an which contain information about the name of the surah, the number of the surah, the number of verses, the verses of the Al-Qur'an Arabic script, the text of the Al-Qur'an in Latin script. If query is entered across the paragraph, then the search results will be displayed more than one verse at a time in one index.

IV. Evaluation

After the system is completed, testing and evaluation are carried out. Testing is done by calculating the value of MAP, recall and correlation from the output of the search results with the entered query to get a value that corresponds to the expected one. Dataset used for measuring the value of MAP is the Al-Qur'an Latin transliteration juz 1 to 30.

A. Mean Average Precission (MAP)

Mean Average Precision (MAP) is a score obtained from measuring system performance in a series of query [4]. A good MAP category can be seen from the AP results. If the overall AP value is low, the system precision is poor. The MAP score is to calculate the average value of average precision for each query.

$$MAP = \frac{1}{\mid Q \mid} \cdot \sum_{i=1}^{Q} AP(Q_i) \tag{1}$$

$$AP = \sum_{i=1}^{N} \left(\frac{TP_{-i}}{TPrank_{-i}} \right) \tag{2}$$

Q is the number of query sets calculated. AP (Qi) is the average precision value for querying i. Before calculating MAP, the average precision value of each query must first be calculated. Average precision is a measurement of precision by calculating the sequence of relevant documents on the system output until the Recall value of the document is equal to 1 or to the output of the document [5].

B. Recall

*Recall*is the probability of relevant information being output compared to the amount of relevant overall information. The score of Recall determines the success rate of the system in conducting verse searches. The maximum value of recall is 1 and minimum 0. If the value of recall system is 1, it means that the system successfully searches for information based on the existing query. Next is the recall calculation notation

$$Recall = \frac{\mid (RelevantInformation) \cap (SystemOutput) \mid}{\mid RelevantInformation \mid} \tag{3}$$

Before calculating the value of MAP and Recall, first set a test set of ten pieces each to search the correct query writing and the query writing is truncated or reversed. This test set will be tried on this system and also Lafzi. Then the results of the MAP and Recall will both be compared.

TABLE V
RESULTS OF SYSTEM TESTING AND RANKING OF SEARCH RESULTS

	Truncated Query		**Normal Query**	
	MAP	Recall	MAP	Recall
Lafzi	0.8904	0.7494	0.9517	0.8504
Lafzi+(this work)	0.9106	0,7636	0.9741	0.8619

The table results of the MAP and Recall tests above show that the search values for normal queries and truncated queries are different. In this study also increases the value of MAP and recall of Lafzi for normal queries in the sense that the results of convergence also occur in this study. The value of MAP and recall for truncated queries at Lafzi also experienced an increase as evidenced by the discovery of neglected characters adding to the value of the query similarity resulting in fewer data outputs that were all correct.

V. Conclusion

Based on the results of the analysis and testing of the existing system, it can be said that the search and ranking of the verses of the Quran using the LCS method. By doing the LCS twice for the queries entered into the search field, it will increase the percentage number of search results. The result will increase the ranking of search results.

By using the datatest in the first juz in the Qur'an, the results of Recall and MAP are shown in Table V. Ranking of results is more appropriate because it calculates the best solution based on the candidate search results that have been obtained. The test results are obtained after the calculation is done several times with the test set normal query resulting in the value of MAP (0.9106) and the value of Recall (0.7636) and truncated query resulting of MAP (0.9741) and the value of Recall (0.8619)

References

[1] Q Abed A Ta'a MA. Al-quran ontology based on knowledge themes. Elsevier; 2017.

[2] Istiadi MA. Sistem Pencarian Ayat Al-Quran berbasis Kemiripan Fonetis. Departemen Ilmu Komputer, IPB. 2012;.

[3] dan Muhammad Rifqi Maarif CH. Implementasi Cosine Similarity dalam aplikasi pencarian Ayat Al-Qur'an berbasis Android. Jurnal Teknologi Informasi dan Komunikasi. 2017;.

[4] Rochmawati Y, Kusumaningrum R. Studi Perbandingan Algoritma Pencarian String dalam Metode Approximate String Matching untuk Identifikasi Kesalahan Pengetikan Teks. Jurnal Buana Informatika. 2016;7(2).

[5] Manning Christoper D PDSH Raghavan. An Introduction to Information Retrieval. Cambridge; 2009.