Stemmer Impact on Quranic Mobile Information Retrieval Performance

Huda Omar Aljaloud
Computer Science Department
King Abdulaziz University
Jeddah, Saudi Arabia

Mohammed Dahab
Computer Science Department
King Abdulaziz University
Jeddah, Saudi Arabia

Mahmoud Kamal
Information Systems Department
King Abdulaziz University
Jeddah, Saudi Arabia

***Abstract*—Stemming algorithms are employed in information retrieval (IR) to reduce verity variants of the same word with several endings to a standard stem. Stemmers can also help IR systems by unifying vocabulary, reducing term variants, reducing storage space, and increasing the likelihood of matching documents, all of which make stemming very attractive for use in IR. This paper aims to study the impact of using stemming techniques in mobile effectiveness. Two-word extraction stemming techniques will be used: a light stemmer and a dictionary-lookup stemmer. Also, three sets of experiments were conducted in this research in order to raise the efficiency of mobile aapplications. Implementing the two stemming approaches and assessing their accuracy by calculating the precision, recall, MAP, and f-measure, produced results which show that the light10 stemmer outperforms the dictionary-lookup stemmer in precision and MAP. Furthermore, the mobile performance of the light10 stemmer exceeds that of the dictionary-based stemmer.**

Keywords—stemming; information retrieval; light10; Quran lexicon; mobile performance; natural language processing

I. INTRODUCTION

The Holy Quran is a global source of knowledge for humanity in general and Muslims in particular. Studying and learning the Holy Quran plays a central role in the lives of all Muslims. Since the Holy Quran is the divine revelation and the word of God, it needs careful handling when processed by automated methods of natural language processing (NLP).The Holy Quran is written in the Arabic language, which is known to be one of the more challenging natural languages in the field [1]. Most researchers have been interested in the development of search techniques for the Quranic text. The techniques employed to retrieve information from the Quran can be classified into two types: semantic-based and lexical-based. The lexical-based search yields results according to the morphological analysis for a query.

Compared to any other kind of communication device, the mobile phone has proved its superiority in communicating and in gaining information. Recognizing this, many companies have focused unprecedented attention on technologies and mobile applications [2]. As a result, the development and evaluation of new technologies for mobile phones occurs very quickly. According to a recent study, smartphone devices will surpass computers as the primary tool by 2020 [3]. This development has inspired researchers to exploit smartphones in various areas, especially in the field of mobile information retrieval (IR) and necessary preprocess phases like stemming. Mobile IR is considered a subset of traditional IR [3].

Stemming has been shown to be more efficient for Arabic retrieval than for English [4]. After several decades of intensive research activity on English stemmers, the techniques of Arabic morphological analysis have become a popular area of research. Early research in this field was performed using small collections until the TREC 2001 Arabic track became available [5]. Root-based stemming, light stemming, and dictionary-lookup stemming are three different types of stemming [6].

To motivate researchers and develop more advanced techniques, Al-sughaiyer et al. in [5] introduced, classified, and surveyed Arabic morphological analysis techniques in an attempt to summarize and organize the information available in the literature of this research area. However, stemmers achieve a noticeable improvement in related NLP tasks [7]. Also, in [8], a comparative study was conducted on most of the existing stemmers (almost twenty) that used different approaches for stemming. The results showed that from 2000 to 2014, the stemmers were used mostly in information retrieval, followed by text classification, with light stemmers being the most commonly used.

A review of semantic search methods was done to retrieve information from the Qur'an corpus in [9]. A proposal for further research in Quranic Knowledge Map was presented in [10]. Moreover, although the Holy Quran is written in Arabic, many efforts to improve word-stemming algorithms are done in other languages [11]. For example, Atwell et al., in [12], investigated the effectiveness of information retrieval in the verse retrieval problem for translated Quranic text in Malay, English, and stemmed English language. A thorough research of the relevant literature found that no research study was done that examined and compared automatic stemming versus manual stemming techniques on Quranic mobile application performance the way it is done in this paper.

In this paper, three questions were investigated: 1) Are automated stemmers more effective than manual-based dictionary-lookup stemmers? 2) Do light stemmers enhance mobile application performance compared to other stemmers? 3) Does using stemmers offline and storing processed tokens with the dataset in the mobile application increase the performance?

The rest of this paper is organized as follows: section two is dedicated to determining the processes needed for preparing the text of the Holy Quran. In section three, the experiments that were applied are presented, while in section four, the results from the experiments are discussed. Finally, section five concludes the paper.

II. Text Preparation

To make the Quran text suitable for stemming, preprocessing its verses was crucial. Preprocessing involved extracting words from the documents by identifying tokens based on document boundary rules. An example of such is the symbol (۝) which marks the end of a verse. During this process, many other operations were applied to the extracted tokens; these operations included text extraction, normalization, removal of stopwords, and stemming. Fig. 1 shows the impact of the preprocessing phase on the verses.

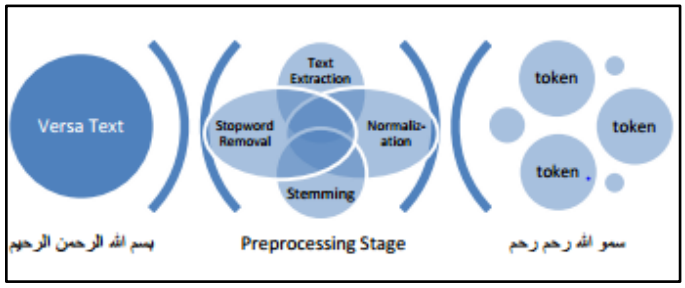

Fig. 1. Impact of preprocessing phase on verse text

A. Text Extraction

Verse text is composed of tokens that are separated by spaces or punctuation marks which are used as token boundaries. The first step was determining the punctuation marks, pause marks, and diacritical marks used in the Quran, and removing them from the text. Punctuation marks, like (۞ ۩), are used to organize the Quran. Pause marks, such as (ۖ ۗ ۘ ۙ ۚ ۛ ۜ ∴ ∴), help reciters decide when to pause and when to recite continuously. Diacritical marks, such as (ً ٌ ٍ ّ ْ ٓ ٔ ٕ), give letters different pronunciation when marked or unmarked differently. Tatweel, which is an elongated version of a letter, was also removed to reflect only the letter. Both the diacritical marks and the tatweel were removed before the normalization process took place. All punctuation and pause marks were removed by using ASCII code obtained from the Uthmanic Scripts Hafs document, downloaded by the King Fahd Complex for the Printing of the Holy Quran (KFGQPC). Table 1 shows the effects of term extraction on the verse text. As can be seen from Table 1, the terms that were extracted did not include any of the unwanted marks. For example, the first word of verse text 1, (أُولَٰئِكَ), contains the following marks that were removed (ُ ٰ ِ ْ ٔ). Once these marks were removed, the resulting term that was left to be extracted was (أولئك).

B. Normalization

Tokens can be written in different forms. In English, a token may appear capitalized at the start of a sentence, and in lower case elsewhere. However, in Arabic, characters have different shapes, and additional variation is added by changing the writing position. For example, the letter "ا " can be written as " أ," "إ," "آ," or "ا." This causes the same token to look different, and critically, to have a different set of character encodings. Normalization was used to represent the tokens in a uniform manner. The effects of normalization on the verse text are shown in Table 2. As can be seen from Table 2, after normalization, characters were presented in a uniform manner. For example, the first word of verse text 1, (أُولَٰئِكَ), contains the following first character (أ). After normalization, the character became (ا), and the resulting normalized word became (اولئك).

TABLE I. Effects of Term Extraction on the Verse Text

Text No	Verse Text	Term Extraction
1	أُولَٰئِكَ عَلَىٰ هُدًى مِّن رَّبِّهِمْ ۖ وَأُولَٰئِكَ هُمُ الْمُفْلِحُونَ	أولئك على هدى من ربهم وأولئك هم المفلحون
2	كَلَّا لَا تُطِعْهُ وَاسْجُدْ وَاقْتَرِب ۩	كلا لا تطعه واسجد واقترب
3	أَمْوَاتٌ غَيْرُ أَحْيَاءٍ ۖ وَمَا يَشْعُرُونَ أَيَّانَ يُبْعَثُونَ	أموات غير أحياء وما يشعرون أيان يبعثون

C. Stopwords Removal

Words that appear very frequently in a data set are considered to add little document-specific information. To avoid the noise that is likely to arise from such words, as well as to reduce the size of the index, they are often omitted during the indexing stage [4]. Stopwords have to be chosen carefully since they affect retrieval. In the Arabic language, stopwords include pronouns, prepositions, adverbs, and function words. Regrettably, Quranic stopwords lists differ substantially, and no single widely accepted list exists. The Quranic stopwords list for this research was adopted from "Stop Word QuranView source – Wiki [1]" (2013) and manually updated. The stopwords list contained the positions of people's names and places. These stopwords were removed from the prime list. Furthermore, a java code was created to collect the most frequent words in the Holy Quran and added the revision list results (word by word) to the Quranic stopwords list used in the mobile application. The updated stopwords list reduced the index by 12%.

TABLE II. Effects of Normalization on the Verse Text

Text No	Verse Text	Normalization
1	أُولَٰئِكَ عَلَىٰ هُدًى مِّن رَّبِّهِمْ ۖ وَأُولَٰئِكَ هُمُ الْمُفْلِحُونَ	اولئك على هدى من ربهم واولئك هم المفلحون
2	كَلَّا لَا تُطِعْهُ وَاسْجُدْ وَاقْتَرِب ۩	كلا لا تطعه واسجد واقترب
3	أَمْوَاتٌ غَيْرُ أَحْيَاءٍ ۖ وَمَا يَشْعُرُونَ أَيَّانَ يُبْعَثُونَ	اموات غير احياء وما يشعرون ايان يبعثون

D. Stemming Technique

Root-based stemming, light stemming, and dictionary-lookup stemming are three different types of stemming [6]. Root-based stemmers are based on a pattern-matching technique to find the root of the word. Patterns include prefixes, infixes, and suffixes to indicate number, gender, and tense [13]. In contrast, light stemming refers to the process of stripping off a small set of prefixes and/or suffixes, without trying to deal with infixes or recognize patterns and find roots.

Dictionary-lookup stemming is based on the manual construction of dictionaries. The root of each processed word is found in the lexicon [14] . Dictionary-lookup stemming is fast since it does not require word analysis, but it does require space and precision in preparing the dictionary [4]. In contrast, light stems use morphological rules to strip off suffixes; light stemming is therefore considered to be a less complicated type of stemming analysis.

Accordingly, to study the impact of different stemming approaches on the accuracy of Quranic IR, dictionary-lookup and light10 stemmers were used in the mobile application. The dictionary-lookup stemmer was based on the Lexicon of the Raw Stems of the Words of the Holy Quran, a manual dictionary by Mohamed Aldabbagh [15]. The electronic database version of the afore-mentioned dictionary was obtained by direct contact with the author. Table 3 shows the effect of stemming on the Quranic word. As can be seen from the first verse word in the table, (بسم), the Light10 stemmer returned the same word (بسم), while the dictionary stemmer resulted in the root word (سمو).

TABLE III. THE EFFECT OF STEMMING ON THE QURANIC WORD

Verse Word	Light10 Stem	Dictionary Root
بسم	بسم	سمو
الله	له	الله
الرحمن	رحمن	رحم
الرحيم	رحيم	رحم

III. EXPERIMENTS

For this research, the Holy Quran was used as a dataset. The dataset contained a total of 6,236 verses, obtained from http://tanzil.net/, and a collection of fifty queries. The queries with corresponding relevance judgments were generated by the authors of this paper. To study the impact of stemming in mobile performance, three experiments were conducted. First, the impact of two stemmer algorithms on IR performance results was evaluated. Second, the mobile performance results of the selected stemmer techniques were compared. The last experiment investigated the ability of offline and online stemmers to raise mobile efficiency.

A. Impact of Stemmer Technique on IR Performance

To examine the impact of text preprocessing on Quranic IR performance, two stemmer techniques were applied. In this experiment, a stem-based technique was employed using light10 and dictionary-lookup stemmers based on the Lexicon of the Raw Stems of the Words of the Holy Quran [15]. In order to determine the effectiveness of verse retrieval in the Holy Quran, the recall, precision, F-measure, and MAP were calculated [16].

B. Impact of Stemmer Technique on Mobile Performance

The effectiveness of a mobile application is usually measured in terms of processing time and memory requirements. SystemPanel 1.4 Task Manager Application was used to examine the CPU and memory usage. Also, the overall Android application package (APK) size was calculated. APK is the package file format used by the Android operating system for distribution and installation of mobile applications.

To gauge the tradeoff between the storage capacity and CPU efficiency, the next two experimental studies were applied. The experiments in this study set were calculated in seconds to figure out the CPU time consumed. The experiments studied the influence of two stemmer techniques used to manipulate smartphone performance. The light10 and dictionary-based stemmers were applied and then demonstrated which allowed more efficient use of the resources.

In the last experiment, two different preprocessing stage periods were applied. Preprocessing involved extracting words from documents, normalization, removing stopwords, and stemming. First, the preprocessing stage was completed before uploading the data collection to a smartphone. Therefore, the verse text preprocessing was completed before the mobile application launched (an offline preprocess). Second, the preprocessing of the verse text was completed after the mobile application launched (an online preprocess).

IV. RESULTS AND DISCUSSION

In this paper, experiments were performed on the Holy Quran (the dataset) with fifty queries between two and three words. Additionally, three sets of experiments were conducted to raise the performance of the mobile application.

Most of the research related to Arabic stemmers is either based on a dictionary of Arabic roots or uses a set of rules to identify the stem of Arabic words [17]. The results of the first experiment, displayed in Table 4, showed that using the light10 system and MAP would give the highest precision rather than using the dictionary-based system. MAP provided a brief summary of ranking effectiveness used by the IR system. In this case, a higher MAP score indicated that the relevant verses were distributed at the top ranks. Similarly, the light10 stemmer outperformed various stemmers in IR. It is widely utilized, as some studies have proven [14][18][7][19]. On the other hand, dictionary stemming for Quranic words achieved better recall than did the light10 system, as shown in Fig. 2.

TABLE IV. STEMMER ALGORITHMS PERFORMANCE MEASURE RESULTS

Stemmer	Recall	Precision	F-measure	MAP
Light10	0.7401	0.9046	0.7700	0.9359
Dictionary Based	0.9891	0.7447	0.8087	0.8318

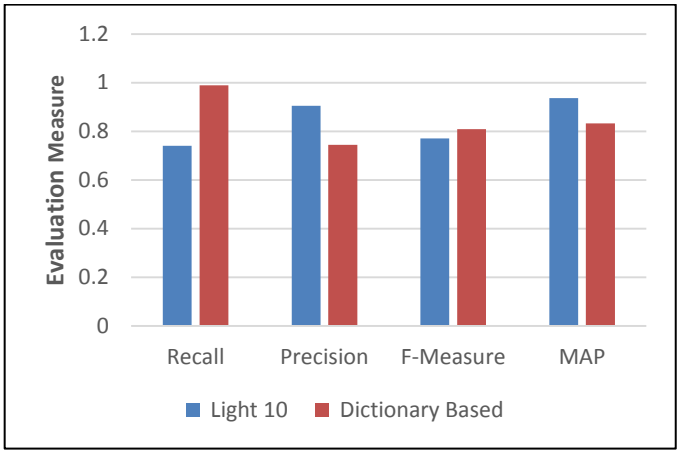

Fig. 2. IR Performance results for stemmers

Table 5 and Fig. 3 present the performance results of the second experiment. The performance of the light10 stemmer, exceeded that of the dictionary-based stemmer in terms of the speed of the CPU. Even the light10 APK file was smaller than the other; 1.918 MB compared to 2.252 MB for the dictionary based stemmer's APK size. The dictionary-based stemmer consumed the CPU without credit in memory usage.

TABLE V. SECOND EXPERIMENT RESULTS

Stemmer	APK Size	CPU Time	Memory Usage
Light 10	1.918 MB	2.75 s	3.44 MB
Dictionary Based	2.252 MB	4.8 s	3.38 MB

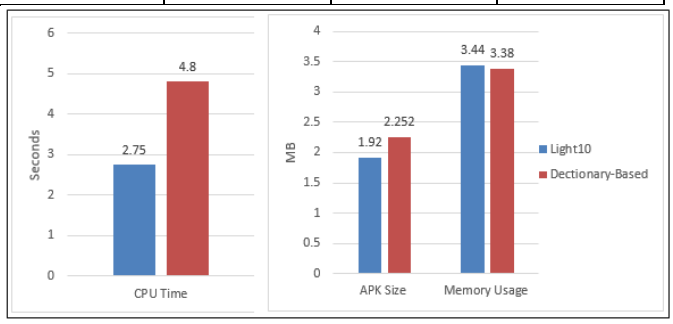

Fig. 3. Second experiment performance

For the third experiment, Table 6 illustrates that the offline preprocessing surpassed the online preprocessing in smartphone performance, despite the fact that the APK size file was slightly larger in offline preprocessing. Offline preprocessing contributed to increasing the performance, as shown in Fig. 4.

TABLE VI. THIRD EXPERIMENT RESULTS

Experiment	APK Size	CPU Time	Memory Usage
Offline Preprocess	1.918 MB	2.75 s	3.44 MB
Online Preprocess	1.789 MB	3.65 s	5.26 MB

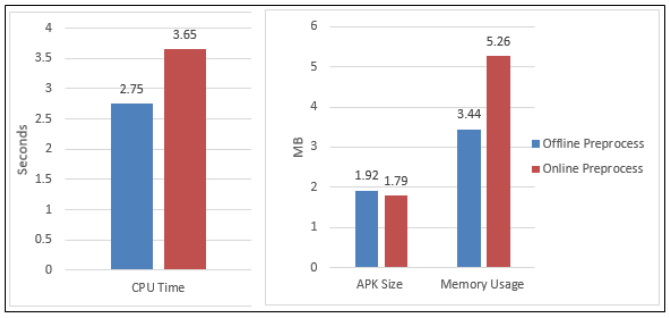

Fig. 4. Third experiment performance

V. CONCLUSION

In this paper, the efficacy of stemmer techniques was examined by studying the impact of different stemming approaches on the accuracy of Quranic IR and mobile performance. Dictionary-lookup and light10 stemmers were used in the mobile application.

In addition, three sets of experiments were conducted in this research to raise the efficiency of the mobile application. First, an experiment was conducted to evaluate the impact of two stemmer algorithms on IR performance results. In the second experiment, the mobile performance results of the selected stemmer techniques were compared. In the final experiment, the impact of offline and online stemmers on mobile efficiency was investigated.

Based on the results of the first experiment, the light10 stemmer demonstrates the highest precision and MAP. Most modern studies indicate that using stems outperforms roots [20]; which these results confirm.

Moreover, to improve the mobile application performance, the results suggested using an offline preprocessing stemmer stage which allows the light10 stem-based system to use mobile resources more efficiently.

Future work based on this study would be to compare automatic root extraction with the manual one used. As for the manual dictionary-based stemmer root approach, the main limitation was in extracting all words to tri-roots only. This can be improved by using an automatic root extractor and including more root patterns.

ACKNOWLEDGMENT

The authors thank Mohamed Aldabbagh for obtaining the electronic database version from the root Quranic lexicon.

Also, the authors thank tanzil.net for providing the data for this research. They also appreciate the efforts of the Zekr Quran Project and Quran Code Desktop Software, which was used to obtain the relevant judgments for the query terms.

REFERENCES

[1] A. Mohammad, H. Mohamed, and H. Anwer, "Processing the Text of the Holy Quran: a Text Mining Study," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 6, no. 2, pp. 262–267, 2015.

[2] P. Racherla and J. Hills, "What We Know and Do Not Know About Mobile App Usage and Stickiness:," International Journal of E-Services and Mobile Applications, vol. 7, no. 3, pp. 48–69, 2015.

[3] F. S. Tsai, M. Etoh, X. Xie, W. Lee, and Q. Yang, "Introduction to Mobile Information Retrieval," IEEE Intelligent Systems, vol. 25, pp. 11–15, 2010.

[4] A. Nwesri, "Effective Retrieval Techniques for Arabic Text," RMIT University, 2008.

[5] I. a. Al-sughaiyer and I. a. Al-kharashi, "Arabic morphological analysis techniques: A comprehensive survey," Journal of the American Society for Information Science and Technology , vol. 55, no. 3, pp. 189–213, 2004.

[6] F. Harrag, E. El-Qawasmah, A. M. S. Al-Salman, E. El-Qawasmah, and F. Harrag, "Stemming as a Feature Reduction Technique for Arabic Text Categorization," in Programming and Systems (ISPS), 2011 10th International Symposium on. IEEE, pp. 128–133, 2011.

[7] M. El-defrawy, "Enhancing Root Extractors Using Light Stemmers," in 29th Pacific Asia Conference on Language, Information and Computation: , pp. 157–166, 2015.

[8] M. Y. Dahab, A. Al Ibrahim, and R. Al-Mutawa, "A Comparative Study on Arabic Stemmers," International Journal of Computer Applications, vol. 125, no. 8, 2015.

[9] E. Atwell and M. Algahtani, "A Review of Semantic Search Methods to Retrieve Information from the Qur ' an Corpus," in corpus Linguistics 2015, pp. 7–9, 2015.

[10] E. Atwell, C. Brierley, K. Dukes, M. Sawalha, and A. Sharaf, "An Artificial Intelligence approach to Arabic and Islamic content on the internet," Proceedings of NITS 3rd National Information Technology Symposium. Leeds, pp. 1–13, 2011.

[11] K. Mohamad Nizam, A. W. Amirudin, M. Mohd Aizaini, and Z. Anazida, "Word stemming challenges in Malay texts: A literature review," in Information and Communication Technology (ICoICT), 4th International Conference, 2016.

[12] A. M. Sultan, A. Azman, R. A. Kadir, and M. T. Abdullah, "Evaluation of Quranic text retrieval system based on manually indexed topics," 2011 International Conference on Semantic Technology and Information Retrieval. IEEE, pp. 156–161, 2011.

[13] M. Y. Ai-nashashibi, D. Neagu, and A. A. Yaghi, "an Improved Root Extraction Technique for Arbic Words," Computer Technology and Development (ICCTD), 2010 second International Conference on. IEEE, pp. 264–269, 2010.

[14] L. S. Larkey, L. Ballesteros, and M. E. Connell, "Light Stemming for Arabic Information Retrieval," Arabic computational morphology. Springer Netherlands, pp. 221–243, 2007.

[15] M. Aldabbagh, Lexicon of the Raw Stems of the Words of the Holy Quran, First edit. Baghdad: avaliable on line by archive.org, 2015.

[16] H. Aljaloud, M. Dahab, and M. Kamal, "Intelligent Information Retrieval Approach Using Discrete Wavelet Transform for the Holy Quran in a Smartphone Application," in 4th International Conference on Islamic Applications in Computer Science and Technology, 2016, in press.

[17] M. N. Al-Kabi, S. A. Kazakzeh, B. M. Abu Ata, S. A. Al-Rababah, and I. M. Alsmadi, "A novel root based Arabic stemmer," Journal of King Saud University-Computer and Information Sciences, vol. 27, no. 2, pp. 94–103, 2015.

[18] M. A. Otair, "Comparative analysis of arabic stemming algorithms," International Journal of Managing Information Technology, vol. 5, no. 2, pp. 1–12, 2013.

[19] M. Ahmed and R. Mamoun, "Arabic text stemming: Comparative analysis," in Basic Sciences and Engineering Studies (SGCAC), 2016.

[20] M. I. Eldesouki and W. M. Arafa, "Stemming techniques of Arabic Language: Comparative Study from the Information Retrieval Perspective," The Egyptian Computer Journal, vol. 36, no. 1, pp. 30–50, 2009