LIGHTWEIGHT VERSION FOR DIGITAL QURAN MODEL BY HANDLING DUPLICATION

Ashraf Saleh Mohammad Alomoush[1], Norita Md Norwawi [2*]

[1] Faculty of Prince al Hussein bin Abdullah II for IT, The Hashemite University, Zarqa, Jordan
[2]Cybersecurity and System Research Unit, Faculty of Science and Technology, Universiti Sains Islam Malaysia, Bandar Baru Nilai, Negeri Sembilan, Malaysia

Article history

Received: 20 Apr 2023

Received in revised form: 25 May 2022

Accepted: 27 June 2023

Published online: 30 June 2023

*Corresponding author: norita@usim.edu.my

ABSTRACT

Digital Quran may be implemented as image-based, text-based, audio, or video based. In terms of storage, the compilation of Quranic verses in images, audio, or video will consume ample storage space. On the other hand, the digital Quran in text-based format benefitted the Hexadecimal representation due to the use of Unicode. However, space was not optimized due to the string concatenation approach of the Unicode of each letter that occurs in a particular Quranic word. Thus, with the Unicode representation, the storage size is directly proportional to the length of a word. This paper presents a new Digital Quran Model (DQM) that aims to reduce storage requirements through individual word conversion into Hexadecimal instead of combining letter sequences. Due to the repetitive nature of Quran words, space usage can be further improved by handling Quranic word duplication. DQM was implemented using Visual Studio and Java servers. The solution quality was measured by comparing the file size before and after applying the DQM model. For example, space reduction for surah Al-Fatihah is 46.91%. The outcome helps the researchers to develop a lightweight Digital Quran application that users can benefit from due to its reliability, validity, and storage management efficiency and thus can be used as a standard application.

Keywords: *Digital Quran, Space Optimization, Word Duplication Handling, Unicode, Hexadecimal*

1.0 INTRODUCTION

Quran is the sacred, most original, and tamper-proof book of Allah SWT since its revelation over 14 centuries ago, mainly in a printed version called Mushaf. Caliphate Abu Bakr initiated the compilation of the Quran, a noble effort continued by Caliphate Umar and completed during Caliphate Uthman were collecting, validating, and finally producing the Mushaf Uthmani for the Muslim world. According to Leaman [1], Prophet Muhammad SAW assigned Zayd ibn Thabit

as the primary scribe whose duty was to gather all the Quranic texts. The task required Zayd ibn Thabit to collect written copies of the Quran and validate each verse with the oral testimony of at least two companions. The Quran was organized under the auspices committee of four senior ranking Companions headed by Zayd ibn Thabit. This canonical corpus is closed and fixed because nothing can be changed or modified in the Quran. This is the model of Quran preservation from the companions to ensure the authenticity of the Quran.

With the advent of Industry 4.0 and the pandemic, Islamic applications on the Internet is awe-inspiring. Daily life routine has been transformed under a single touch, including the use of online technology for spreading religion, learning about their faith with online education and distance education, Quran memorization, Quran teaching, online businesses, banking, socializing, politics and communication, sharing news, research works, and spreading their religion [2] However, Larsson and Hoffman [3] claimed that online Quran and Islamic books are lagging in employing structured digital content

2.0 SPACE OPTIMIZATION WITH QURAN WORDS DUPLICATION HANDLING

The Quran is initially written in classical Arabic characters. In the computer application, Arabic letters are represented by UTF-8 character encoding having compatibility with the ASCII code in a backward manner. UTF-8 is a variable-sized coding method to encode the text; each character needs two bytes to code [4,5]. According to [6], the Digital Qur'an consists of 68 characters, where 44 characters can be stored in an ANSI text file, while the other 24 characters can be in a UTF-8 text file. They use UTF-8 character encoding for the digital Quran to represent chapter names, verse numbers, page numbers, and verses, reducing the storage size compared to the actual Arabic characters.

The Arabic language is written using some 28 letters, where 16 of them have one dot, two, or three dots. Arabic is written from right to left, and numerous fonts exist, with letters changing their shape according to where they occur in the text. The Quran contains 114 chapters, 30 juzu, 6236 verses, 77797 words, and 330,709 letters [7]. As the number of words is considerably high, storage optimization and safe searching time are essential.

Almazrooie [4] proposed a cryptographic hash function for digital Holy Quran verses and converted the Arabic alphabet to hexadecimal for compression. The compression method proposed includes the sample representing the verse **الله الصمد** with the size of 33 Bytes, and after the compression, it becomes 17 Bytes. The same algorithm was applied to the verse **اياك نعبد واياك نستعين** then the sample size was reduced from 73 Bytes to 38 Bytes. Almazrooie [4] and Hakak [8] adopt strings concatenation, where the length correlates with the number of letters in a word, thus requiring space. However, the algorithm ignores duplication of word **اياك** where space can be optimized with handling words duplication mechanism. This paper will present a new Digital Quran model using hexadecimal representation and improve the content data structure with Quranic word duplication handling.

Accordingly, the current study builds on the existing literature and aims to develop a model that can further optimize storage usage for the digital Quran. It is important to note that the current gaps within the literature, according to several studies, are the storage optimization aspect [4,8-10]. In addition, the Arabic language requires further analysis and evaluation regarding

applications and UTF measures [4]. The extant literature on Chinese, Hindi and Arabic languages have been addressed several techniques to be represented. For instance, Chinese characters are approximately 20,000, with 6.700 commonly used [11]. There are compound words shaped by these characters that can vary in length, such as “海上” and “上海” as above (上) and sea (海). The word “海上” translates into above the sea. However, t上海” means Shanghai [4]

After English and Chinese, the Hindi language comes third in the context of a Unicode function being retrieved from the web [12]. This has been linked to a need for more understanding of the language and how the Hindi language is presented. Limited literal matched patterns also complicate good algorithms [13]. Relevant to the context of this research, studies have included and investigated the Arabic language regarding its modern standard form. This has created a challenge for establishing Quranic Arabic (an ancient form of the Arabic language). However, according to several studies, the Quranic Arabic has significance for Muslims worldwide and is not neglected [14]. Some studies address the importance and challenges of the digital Quran. It has been established that transforming Quranic words in Arabic requires further optimization, analysis, and model developments to ease the app’s usability for all users across all devices [8].

2.1 Digital Quran Model in Lightweight Version

For the Holy Quran, there are 6236 verses where the shortest verse has one word only, and the most extended verse has 129 words which are in Al-Baqarah verse 282 [7] Thus, if the whole Quran is presented in a table or matrix, the dimension is 6236 rows, and a maximum of 129 columns where each row represents one verse, and the column represents the word in the verse. The Digital Quran content structure uses table representation or matrix as adopted by [4], who used 6236 rows of elements to represent each verse in the Quran, whereas [8] used 6234 rows, as shown in Fig 1. The memory size can be computed as 6236 row ×129 column ×Max size in Bytes for word = 4.022 MB (Assumes max size = 5B)

word / verse	w_1	w_2	w_3	....	w_i
v_1	ID_{11}	ID_{12}	ID_{13}	....	ID_{1i}
v_2	ID_{21}	ID_{22}	ID_{23}	....	
v_3	ID_{31}	ID_{32}	ID_{33}	....	
⋮	⋮	⋮	⋮	⋮	
v_i	ID_{i1}	ID_{i2}			ID_{ii}

Figure 1: Sparse Matrix Represent Words (w) and Verses (v) of the Quran

There are two concerns for space optimization. First, the space without word entry in the matrix, as shown in Fig. 1, is a storage wastage commonly named Sparse Matrix. Second, the duplication of words that reoccur in other verses (rows) requires different storage space. For example, Surah Al-Fatihah contains 143 characters, 28 words with 19 unique words in seven verses can be represented in a matrix as in Fig. 2. The verse بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ is represented in UTF8 Unicode as FDFD in Hexadecimal.

	1	2	3	4	5	6	7	8	9	10
1										بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ
2							الْعَالَمِينَ	رَبِّ	لِلَّهِ	الْحَمْدُ
3									الرَّحِيمِ	الرَّحْمَنِ
4								الدِّينِ	يَوْمِ	مَالِكِ
5						نَسْتَعِينُ	إِيَّاكَ	و	نَعْبُدُ	إِيَّاكَ
6								الْمُسْتَقِيمَ	الصِّرَاطَ	اهْدِنَا
7	الضَّالِّينَ	لا	وَ	عَلَيْهِمْ	الْمَغْضُوبِ	غَيْرِ	عَلَيْهِمْ	أَنْعَمْتَ	الَّذِينَ	صِرَاطَ

Figure 2: 10 x 7 matrix representation of Surah Al-Fatiha

Referring to Fig. 2, if we use a two-dimension array, then the space reserved for this surah is 10x7 = 70 memory allocations. However, only 28 spaces are being used. Thus, a wastage of 60 % of space was allocated for the content structure for surah Al-Fatihah as computed using Eq. 1.

$$waste = \frac{(70 - 28)}{70} = 60\% \quad (1)$$

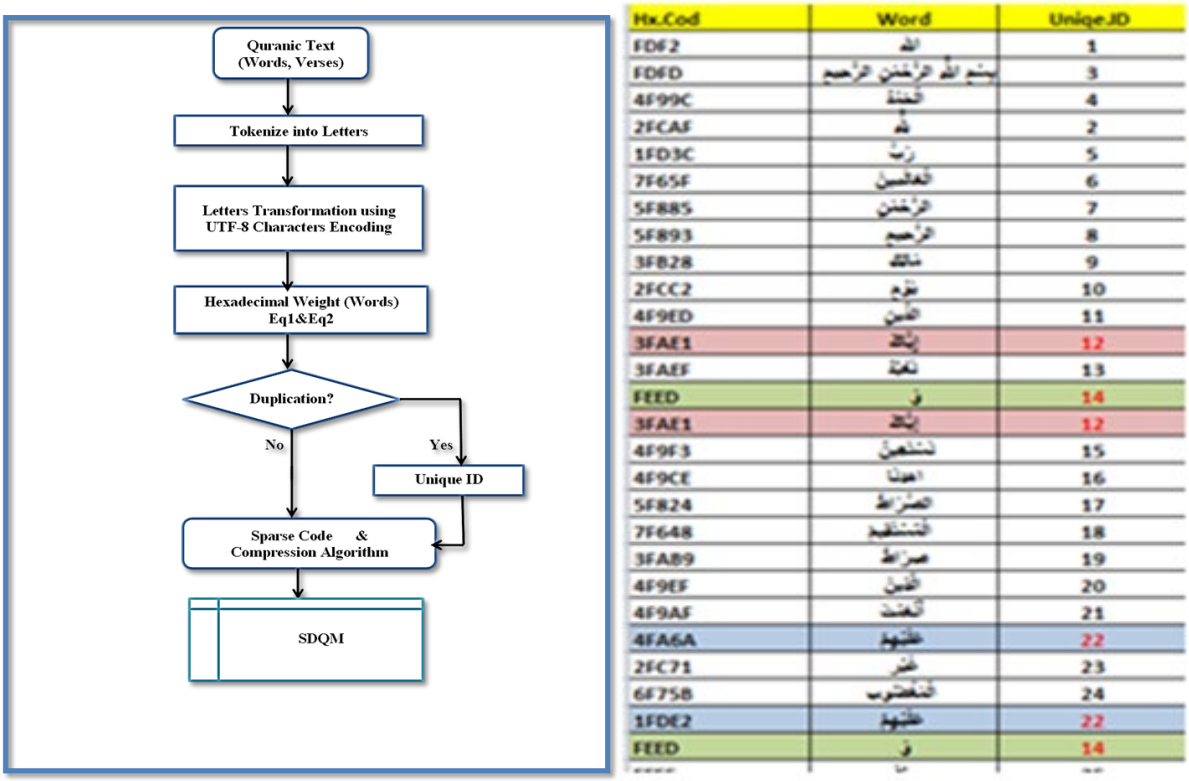
Besides that, repeated words are given a space. Quranic words repeat themselves throughout the Quran. The Quran has 77,797 words but only 14,870 are unique [7]. Based on these facts, this study proposed an index-like listing with a unique ID for each word to make up a look-up table that can further reduce the verse's representation by only referring to the unique ID. Fig. 3 illustrates an example of the look up table feature

Kalimah	Count	New Hex	ID
اليهود	6	5F8E4	1766
بنورهم	6	5F8E3	1767
نفعهما	6	5F8E5	1768
يكذبون	6	5F8E0	1769

Figure 3: Lookup Table with Unique ID for each word

An index listing with a unique ID for each word was constructed. Thus, the content structure of the surah, for instance, will only be represented by the unique ID in numbers which use less space due to its binary representation on the presentation layer. This approach will reduce memory, thus saving space and time due to faster access at the presentation layer in a binary bitwise form. Fig. 4 illustrates the process of creating the digital Quran content structure with a look-up table to handle repeating words with a sparse matrix.

Duplications were handled by creating an index listing all unique Quran words. A word representation lookup table was created with three columns: the Hexadecimal code of the word, the Arabic text and the unique ID. Verses will be rewritten and represented using the unique ID in the form of a matrix, as shown in Fig. 5.



Hx.Cod	Word	UniqeJD
FDF2	الله	1
FDFD	بسم الله الرحمن الرحيم	3
4F99C	الحمد	4
2FCAF	لله	2
1FD3C	رب	5
7F65F	العالمين	6
5F885	الرحمن	7
5F893	الرحيم	8
3FB28	مالك	9
2FCC2	يوم	10
4F9ED	الدين	11
3FAE1	إياك	12
3FAEF	نعبد	13
FEED	و	14
3FAE1	إياك	12
4F9F3	نستعين	15
4F9CE	اهدنا	16
5F824	الصراط	17
7F648	المستقيم	18
3FAB9	صراط	19
4F9EF	الذين	20
4F9AF	انعمت	21
4FA6A	عليهم	22
2FC71	غير	23
6F75B	المغضوب	24
1FDE2	عليهم	22
FEED	و	14

Figure 4: The flowchart for Duplication Handling with Unique ID Look Up Table

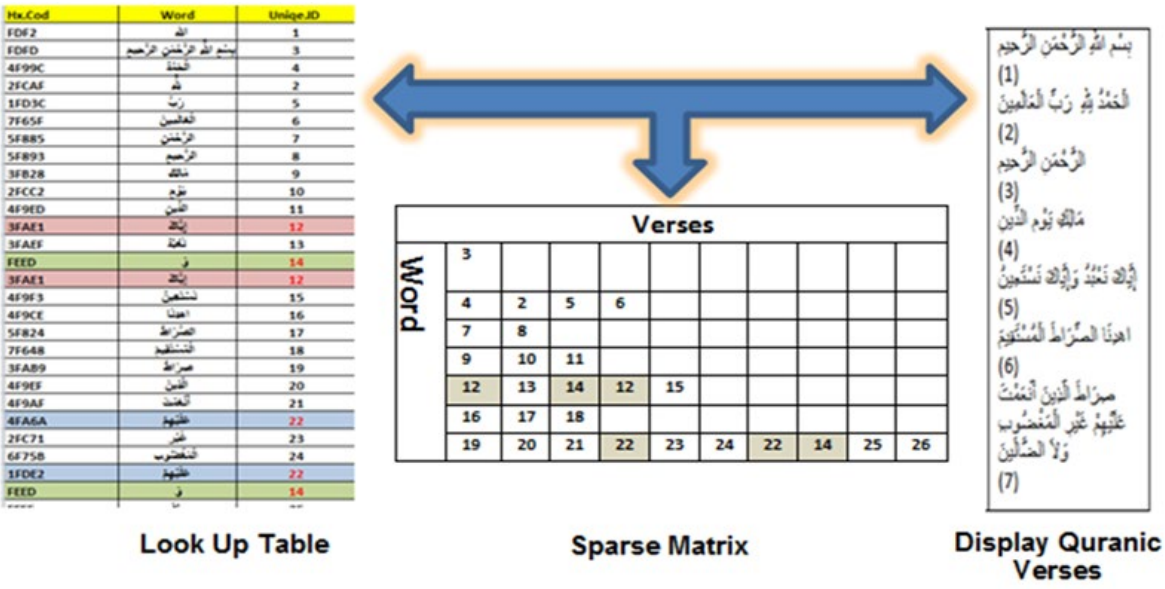


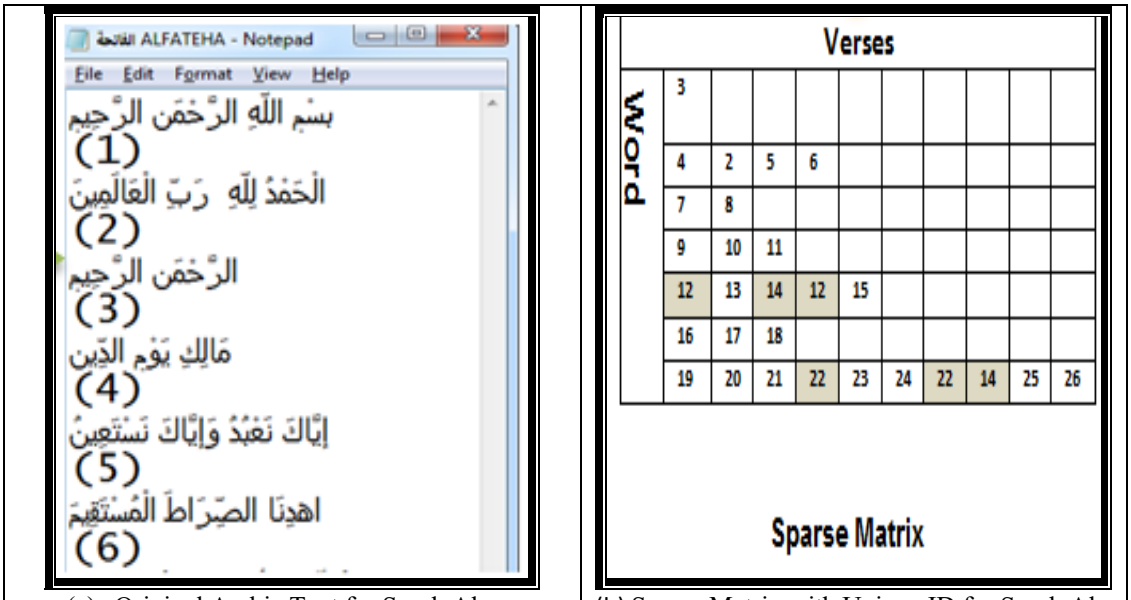
Figure 5: Quran Representation with Unique ID in a Sparse Matrix for Surah Al-Fatihah

The storage of words is being optimized due to using one memory space for that word instead of one memory space for each Arabic character in the particular term. For example, **الحمد**

has five letters meaning 2 Bytes x 5 letters = 10 B memory space where each letter is stored in 2 bytes (16 bits). If a unique ID represents the word, it only requires 2 bytes for storage. This reduces 80% of the space needed, as illustrated in Table 1.

Table 1: Comparison of storage size الحمد and its unique ID

Letter	د	مـ	حـ	لـ	ا	الحمد	Unique ID = 4
Storage Size (Byte)	2	2	2	2	2	10	2



(a) Original Arabic Text for Surah Al-Fatihah

(b) Sparse Matrix with Unique ID for Surah Al-Fatihah

Figure 6: Comparison of Al-Fatihah Representation

Table 2: Summary of the Comparison Words Size Upon Using Unique ID in Lookup Table

SURAH	Number of Letters	Size of Hex Text (Bytes)	Number of Words	No Unique Words	Size (Bytes)	Total Size Reduction (%)
Surah Al-Fatihah	143	81	28	19	38	53.09%

Figure 6 shows the comparison between the different representations of Surah Al-Fatihah. The content structure with the unique ID can be easily decoded into its original Arabic text when it needs to be displayed. This is in contrast with [4] and [8] that the concatenation of the

hexadecimal of each letter thus forms a string of concatenated hexadecimal characters. It does not optimize space, nor does it handle repeating words of the Quran. Table 2 shows the reduction in space for surah al-Fatihah.

Table 2 shows how using a unique ID for each unique Quranic word will reduce the file sizes for surah Al-Fatihah as shown in Eq 2. The text size for surah Al-Fatihah is 81 bytes, whereas the unique ID for this surah is 19 x 2bytes= 38 bytes with a 53.09% reduction in the storage size.

$$reduction(\%) = \frac{size\ of\ original\ text\ with\ duplications - size\ of\ text\ with\ unique\ ID}{size\ of\ original\ text\ with\ duplications}$$
$$= \frac{(81-38)}{81} \times 100 \qquad (2)$$
$$= 53.09\ \%$$

3.0 CONCLUSION

This paper presented a new lightweight version of the Digital Quran model by handling the duplication of Quran words. Each unique Quranic word is given a unique ID and listed in a lookup table. This approach optimizes the memory space, as discussed, due to the presentation using numbers for the unique ID compared to the concatenation of Unicode in Hexadecimal representation for each letter in a particular word.

Acknowledgement
The author would like to thank the Universiti Sains Islam Malaysia and the Ministry of Higher Education for the Niche Research Grant Scheme 2014-2020 titled Knowledge Reservoir Through the Integration of Naqli And Aqli Knowledge from Heterogeneous Expertise and Sources that have sparked the ideas on storage requirements for Quranic and Hadith text as part of the needs of the study.

REFERENCES

[1] L, Oliver. (2006). The Qur'an: an Encyclopedia. Routledge, 30–31.

[2] Zakariah, M., Khan, M. K., Tayan, O., & Salah, K. (2017). Digital Quran computing: review, classification, and trend analysis. *Arabian Journal for Science and Engineering*, 42, 3077-3102.

[3] Larsson, G. & Hoffman, T. (2012). The impact of the Internet on the Qur'ān. Muslims and the New Information and Communication Technologies, Springer, Berlin.

[4] Almazrooie, M., Samsudin, A., Gutub, A. A. A., Salleh, M. S., Omar, M. A., & Hassan, S. A. (2020). Integrity verification for digital Holy Quran verses using cryptographic hash function and compression. *Journal of King Saud University-Computer and Information Sciences*, 32(1), 24-34.

[5] Hilal, T. A., & Hilal, H. A. (2019). Arabic text lossless compression by characters encoding. *Procedia Computer Science*, *155*, 618-623.

[6] Foda, K. M., Fahmy, A., Shehata, K., & Saleh, H. (2013). A Qur'anic Code for Representing the Holy Qur'an (Rasm Al-'Uthmani). *2013 Taibah University International Conference on Advances in Information Technology for the Holy Quran and Its Sciences*, 304-309.

[7] Quran Statistics. (n.d.). Quran Analysis. https://qurananalysis.com/analysis/basic-statistics.php (accessed on Jun. 26, 2016).

[8] Hakak, S., Kamsin, A., Tayan, O., Idris, M. Y. I., Gani, A., & Zerdoumi, S. (2017). Preserving content integrity of digital holy quran: Survey and open challenges. *IEEE Access*, 5, 7305-7325.

[9] Mouratidis, H., Islam, S., Kalloniatis, C., & Gritzalis, S. (2013). A framework to support selection of cloud providers based on security and privacy requirements. *Journal of Systems and Software*, 86(9), 2276-2293.

[10] Saada, B., & Zhang, J. (2015). Vertical DNA sequences compression algorithm based on hexadecimal representation. *Proceedings of the World Congress on Engineering and Computer Science*, 2.

[11] Law, H. H. C., & Chan, C. (1996). N-th order ergodic multigram HMM for modeling of languages without marked word boundaries. *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

[12] Tripathi, A. (2012). Problems and prospects of Hindi language search and text processing, 59, 219-222.

[13] Sharma, S., Bora, N., & Halder, M. (2012). English-Hindi transliteration using statistical machine translation in different notation. *Training*, 20000(297380), 20000.

[14] AlMaayah, M., Sawalha, M., & Abushariah, M. (2014). A proposed model for Quranic Arabic WordNet. *Proceedings of the 2nd Workshop on Language Resources and Evaluation for Religious Texts, 31 May 2014, Reykjavik, Iceland*, 9-13.