International Journal of Advanced Trends in Computer Applications
www.ijatca.com

Big Data Compression for Aadhaar Storage through Reduced Bit Level Ordering

[1]Kulanthaivel. G , [2]Ezhilarasu. P, [3]Ulagamuthalvi.V
[1]Professor, NITTTR, Chennai,
Tamil Nadu, India.
[2]Associate Professor,
Department of Computer Science and Engineering, SRMIST, Uttar Pradesh, India.
[3]Associate Professor,
Sathyabama Institute of Science and Technology,
Chennai ,Tamil Nadu, India.
[1]gkveldr@gmail.com,[2]prof.p.ezhilarasu@gmail.com, [3]ulagamv@gmail.com

Abstract: *In this paper explores on compression of Aadhaar number storage through the concepts of reduced bit level ordering. The Aadhaar number is an unique number used in various government schemes. It is a twelve digit number. The population of our country is more than 135 crores. It means more than 135 crores of Aadhaar numbers will be available. The uniqueness of beneficiary is ensured by avoiding duplicate beneficiary. The uniqueness needs Aadhaar number comparison with minimal amount of time. Hence there is a need for reducing the Aadhaar storage for minimizing the search time. The numbers are represented by using various number representations with their needed storage. The compression of Aadhaar storage implemented by using the concept of bit level ordering which takes sorted integer as an input. The expected saved space shown in graphs and tables. The input from various bit level taken and output obtained as index bit and content bit. The obtained saved space depicted through table and graphs.*

Keywords: Big data, Compression, Aadhaar, Bit level ordering, Space reduction**.**

I. INTRODUCTION

1.1 Big Data

The impact of social media leads to exponential growth of digital data. It is growing exponentially daily. This collection of digital data named as Big Data. The data are categorized as structured data, semi structured data and unstructured data. Incase of structured data sometime it is possible to process the data manually. But in case of semi structured and unstructured data we need an analytic tool for processing of data. The prime objective of big data analysis is to process data of high volume, velocity, variety, and veracity using various traditional and computational intelligent techniques [1].

1.2 Compression

The system of reducing the size of a data file referred to as data compression [2]. Data compression uses the trade-off in the form of space and time complexity. If the data stored in its original form, then there is no need to perform compression and decompression of the taken input data which require huge storage. By applying compression space can be reduces. This efficiently minimize space require and time for transfer. Data transfer rate is calculated by eq.1.

Data transfer time = Input data /

Data transfer rate (1)

Data transfer time required depend on the data rate and size of data. Depend on the storage of the destination files will be stored in storage. The space and time

complexity based on compression ratio. It can be calculated by using the following equation 2.

Compression ratio = Actual uncompressed data / Compressed Data (2)

The data compression also has specific limitations for compression and decompression. If processing time is more then the need for performing compression practically not needed. Limitations can be Compression and Decompression cost, time and quality of data. Lossy and lossless compression can be used.

1.3 Aadhaar

The United Nations, Department of Economic and Social Affairs, Population Division released world population prospects 2019[3]. According to this report the population of India is 136.6 crores. Aadhaar is a verifiable 12-digit identification number issued by UIDAI to the resident of India for free of cost [4]. This number is similar to nine digit Social Security number issued by Social Security Administration in United States of America. The Aadhaar number used in many government schemes to identify beneficiary. In many bank Aadhaar enabled Payment System (AePS) is a bank led model which allows online interoperable financial inclusion transaction at PoS (MicroATM) through the Business correspondent of any bank using the Aadhaar authentication [5]. The fake beneficiary removed from the database using Aadhaar number in many government schemes. This helps to strengthen the economy of India. This Aadhaar number is based on population of India. The population of India is more than 135 crores means the available Aadhaar number is more than 135 crores. The Aadhaar number has many details like Enrolment number, Aadhaar number, Date of Birth, Address, QR code, Gender. The state and central government has many schemes based on Aadhaar number.

II. RELATED WORK

Claude Elwood Shannon who is known as father of Information Theory discussed about discrete random variable and its information content [6]. Inverted indexes , which is the most common data structure used by search engines to index their data, are made of lists of increasing integers corresponding to the documents of the collection[7]. Importance of security to save data in big data is done by various techniques [8,9]. If we compare Elias-Fano encoding space requirement with the theoretical lower bound we realize that this structure is close to the bound, so it has been epithet quasi-succinct index [10]. The Elias-Fano integer encoding of monotone sequences of integers solved using succinct data structures [11, 12, 13]. Compression of inverted indices saves disk space, but more importantly also reduces disk and main memory accesses [14], resulting in faster evaluation. An inverted index is a collection of sorted sequences of integers [15, 16, 17]. Compressing such sequences is a crucial problem which has been studied since the 1950s; the literature presents several approaches, each of which introduces its own trade-off between space occupancy and decompression speed [20, 21, 22, 23]. Elias-Fano coding for various length of sorted integers shows that input with 3 bits length (1 highest bit) cannot produce compression. The 10 bit highest bit provides 73% of space savings. Hence for storing 100 bits we need only 27 bits. The increased number of highest bit (>10 bits) results in more amount of space savings (>73%)[21].

III. AADHAAR NUMBER REPRESENTATION

The Aadhaar number can be stored using the number representation methods such as

- ASCII CODE REPRESENTATION
- NORMAL BINARY REPRESENTATION

3.1 Number Representation

The ASCII code range is from 0 to 255. So we need eight digits (2 powers 8 = 256) for representing each number. The numbers from 0 to 9 represented in ASCII using the numbers 48 to 57. In normal binary representation the maximum number 9 has the size 4. So we need four digits for representing each number. The number representation for both number representations is given in the table 1.

Table 1: ASCII code representation for numbers (0-9)

S.NO	***NUMBERS***	***ASCII REPRESENTATION***	***NORMAL BINARY REPRESENTATION***
1	0	00110000	0000
2	1	00110001	0001
3	2	00110010	0010

4	3	00110011	0011
5	4	00110100	0100
6	5	00110101	0101
7	6	00110110	0110
8	7	00110111	0111
9	8	00111000	1000
10	9	00111001	1001

3.2 Storage Need

The storage needed for storing 135 crores Aadhaar numbers using the above representation is given in Table 2 and Figure 1.

Table 2: Needed storage for 135 crores Aadhaar numbers

S.NO	REPRESENTATION	EXAMPLE AADHAAR NUMBERS	TOTAL DIGITS AFTER PAIRING	SIZE OF INDIVIDUAL PAIRS	SIZE OF SINGLE AADHAAR NUMBER	SIZE OF 135 CRORES AADHAAR NUMBERS
1	ASCII	9999 9999 9999	12	8	96	15.08 GB
2	NORMAL BINARY	5555 5555 5555	12	4	48	7.54 GB
3	2-COUPLE	99 99 99 99 99 99	6	7	42	6.60 GB
4	3-TRIPLE	453 453 453 453	4	10	40	6.28 GB
5	4-QUADRUPLE	1060 1060 1060	3	14	42	6.60 GB
6	6-SEXTUPLE	555555 555555	2	20	40	6.28 GB

Figure 1: Needed storage for 135 corers Aadhaar numbers

IV. AADHAAR COMPRESSION THROUGH BIT LEVEL ORDERING

The possible range of Aadhaar number is from 000000000000(12 Zero's) to 999999999999(12 Nine's). The available Aadhaar numbers are sorted in ascending order. Then they are compressed. The compression of sorted integers is implemented using eliasfano coding by splitting the numbers into high and low level bits. In this work only sorted integers are used. Instead of splitting as high level and low level, here bit level ordering is used. In bit level ordering the bit level and its occurrence are used. The twelve digit Aadhaar number is considered as decimal number. Then decimal to binary number conversion is performed. The size of binary number is termed as level. Ex. 123412341234 is converted into binary as

11100101110111111100101110000111110010 and its level as 37.

V. RESULTS AND DISCUSSION

If the total number of Aadhaar numbers are 150 and there consolidated count in reduced bit level are as given in the table 3, then it is possible to store and search Aadhaar numbers using index bit and storage bit with contents of index and storage bits is calculated and storage decided according to that. Spaced saved for storing all Aadhaar card is shown in table 3. The range from 000000000000-001349999999 denotes first 135 cores possible Aadhaar numbers. The needed bit for this range using reduced bit level ordering is 38352516324 bits. Average bits used to save are 28.41 and 39 bits after bit ordering.

Table 3: Saved space for various number representation using 150 Aadhaar numbers

S.NO	NUMBER REPRESENTATION	CHARACTER TYPE	CONSOLIDATED TOTAL BIT	SAVED SPACE
1	ASCII	Fixed - 8 bits	150 * 96 = 14400	0%
2	NORMAL BINARY	Fixed - maximum 4 bits	150 * 48 = 7200	50%
3	2-COUPLE	Fixed - maximum 7 bits for 2 characters	150 * 42 = 6300	56.25%
4	3-TRIPLE	Fixed - maximum 10 bits for 3 characters	150 * 40 = 6000	58.33%
5	4-QUADRUPLE	Fixed - maximum 14 bits for 4 characters	150 * 42 = 6300	56.25%

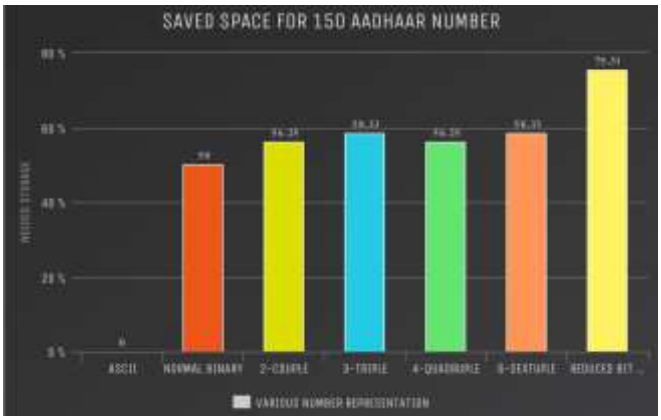

Figure 2: Space used for storing 150 and 135 cores Aadhaar numbers

VI. SEARCHING

The reduced bit level for the given number is taken. The consolidated previous bit can be calculated using multiplication of previous bit level and its occurrence. Ex. If the given number is 137438953490(**00000000000000000000000000000000010100**) its reduced bit level is 37, then consolidated previous bit is 1*1+2*2+3*3+...+35*6+36*4=2469. The occurrence for 37th reduced bit level is 2. Hence we need to search from 2470to 2506 then 2507 to 2543. If match found then the number is present else the given number is not present. Here the given number 137438953490(**00000000000000000000000000000000010100**) matches the bit from 2507 to 2543. Hence the given number is present. It is explained below.

100110001011110000010010110111110000000010
110010000000000010000100000110010000110111110
000111010000000000000110000110001001000101000
001110111010001101001111001011101001111111000
000000011001001001110010101100101011011111111
101111111100000000000000101000010100000101100
100101111101001000101100001101111010100000000
000000010101001101100010111101110011111111111
000000000000000001101101111010001011111111111
000000000000000000000011000000000011100000110
101000111000101001000101001000000000000000111
111111111100001001101010000110010100101101010
100101100000000000000000000000011101010101011
011011010111111111111110000000000000000000000
000111010010010101111100001001011111100100100
111110111101001000000011101000100101111110000
100010011111001101110101111111111011100100000

00000011001011111010000100100010111110111101
00001000000000000000000001111111111111111111
00000000000110101010000011001000111000100010
11010110100010010010110101111100101011101010
10111001001011101010111111010011110101101000
10111011000000000000000110001011011010101101
11110011111010000100100000010000000000000000
00000010011111111111111001010100000000000000
00000000011000000000000000100011001010101001
10001010110001111001011100100000001010000000
11111010111100001000001010000110111000111000
01001110110010000101110011101011110111100010
10010000010000000000000000000001001010000100
11100000011000001111001111101011110000100000
00101101110011010110010100000100000100011101
00011010111001111101111110100100100000000100
00010001110100011010111001010011110111100100
00110101011000111001011011111100100111001110
10011111011010111000111010110111001101011100
10100000001011111111111111111111111111111000
00001011111011111100000001000000001110000011
10010010010010000001111100000010100100000100
00010001110100011010111001001111111100011011
10011110000001000011001011101110110010010001
11010001010100000010111110010000000000101001
01100100011000011001110010000010010110100110
10110111110010101110100101101001101011100001
01010111001011101000010010110000100101110111
01111110000100011101011000000001011001111100
10111011110000111001001111000000010011000000
10111100100001111000010110100100100110101010
11100000000000000000000000000000000000001010
10000001011111001000000000010001100010111100
10100101101000000010100111011110101111010000
00011100101111100001011010010010011010101011
00111110001111101111011010000011100110000000
00000000000000000000000000010100000000000000
00000000000000011001000000000000000000000000
00000101110100000000000000000000000000001000
01010 (**bit 1 to bit 2469**)

000000000000000000000000000000000000 (**bit 2470 to bit 2506**)
0000000000000000000000000000000010100 (**bit 2507 to bit 2543**)
00000000000000000000000000000000000000000000
00000000000000000000000000000100000000000000
00000000000000000000000000000000000000000000
00000000000000000000100000000000000000000000
00000000000000100.

VII. CONCLUSION

The Aadhaar number is used as an index for the remaining Aadhaar details for an individual. The government scheme is based mostly on Aadhaar database. The uniqueness of beneficiary ensured by merging the Aadhaar database followed by checking. We need to check the Aadhaar number with their successor. If both are same then it is fake beneficiary. The data used in Aadhaar card for entire nation comes under Big data. Hence compression plays big role in storage of Big data. This Aadhaar number compression and storage concept can be used in many applications related with government scheme. The saved space provided by reduced bit level ranges from 59.33% to 70.41% for 135 crores Aadhaar number. The range differs based on the number of total number of Aadhaar numbers. If the Aadhaar number is lesser than 135 crores then the starting range will be minimum (<59.33%) and ending range will be maximum (>70.41 %.). In reverse case if the Aadhaar number is greater than 135 crores then the starting range will be maximum (>59.33%) and ending range will be minimum (<70.41 %.).

REFERENCES

[1] M. K.Kakhani, S. Kakhani and S. R.Biradar, Research issues in big data analytics, International Journal of Application or Innovation in Engineering & Management, 2(8),pp.228-232, 2015.
[2] Available online at: http://en.wikipedia.org/wiki/Data_compression
[3] United Nations, Department of Economic and Social Affairs, Population Division (2019). World Population Prospects Highlights. ST/ESA/SER.A/423.2019.
[4] Available online at: https://uidai.gov.in/
[5] Available online at: https://www.npci.org.in/product-overview/aeps-product-overview
[6] C. E. Shannon: A Mathematical Theory of Communication, The Bell System Technical Journal(27): 379-423, 623-656, July, October, 1948.
[7] Justin Zobel and Alistair Moffat. 2006. Inverted files for text search engines. ACM Comput. Surv. 38, 2, Article 6, 2006.
[8] S.R.Khonde and V. Ulagamuthalvi, "Fusion of Feature Selection and Random Forest for an Anomaly-Based Intrusion Detection System", Journal of Computational and Theoretical Nanoscience**, v**ol. 16, pp. 1–5, 2019.
[9] S.R.Khonde and V. Ulagamuthalvi, "Hybrid Architecture for Intrusion Detection System", Ingenierie des Systemes d'Information, vol. 24, no. 1, pp. 19-28, 2019.
[10] Sebastiano Vigna, "Quasi-succinct indices", In Proceedings of the sixth ACM international conference on Web search and data mining (WSDM '13). ACM, New York, NY, USA, 83-92,2013.
[11] P. Elias, "On binary representation of monotone sequences", Proceedings of the 6-th Princeton Conference on

Information Sciences and Systems: 54-47, Princeton University, 1972.
[12] P. Elias, "Efficient storage and retrieval by content and address of static files", J. ACM 21(2): 246-260, 1974.
[13] R. M. Fano, "On the number of bits required to implement an associative memory", Memorandum 61, Computer Structures Group, Project MAC, Massachusetts Institute of Technology, 1971.
[14] S. Büttcher and C. L. A. Clarke. Index compression is good, especially for random access. In M. J. Silva, A. H. F.
[15] S. Buttcher, C. L. A. Clarke, and G. V. Cormack. ¨ Information retrieval. "Implementing and evaluating search engines", MIT Press, Cambridge, Mass., 2010.
[16] C. D. Manning, P. Raghavan, and H. Schulze, "Introduction to Information Retrieval", Cambridge University Press, 2008.
[17] J. Zobel and A. Moffat, "Inverted files for text search engines", ACM Comput. Surv., 38(2), 2006.
[18] D. Lemire and L. Boytsov, "Decoding billions of integers per second through vectorization", Software: Practice & Experience, 2013.
[19] A. Moffat and L. Stuiver, "Binary interpolative coding for effective index compression",. Inf. Retr., 3(1), 2000.
[20] D. Salomon,"Variable-length Codes for Data Compression", Springer, 2007.
[21] A. A. Stepanov, A. R. Gangolli, D. E. Rose, R. J. Ernst, and P. S. Oberoi. "Simd-based decoding of posting lists", In CIKM, pages 317–326, 2011.
[22] H. Yan, S. Ding, and T. Suel, "Inverted index compression and query processing with optimized document ordering", In WWW, pages 401–410, 2009.
[23] Ezhilarasu P, Mahapatra RP, Senthil R, "Analysis and Interpretation of Elias-Fano coding for Sorted Integers", LAP LAMBERT Academic Publishing, 2019.

Author Profile

Dr. G. Kulanthaivel,Professor & Head of the Department of ECE, NITTTR, Chennai, India received his Ph.D. degree in Information and Communication Engineering from Anna University, Chennai. He completed his Master's degree in Microwave and Optical Engineering from Madurai Kamaraj University and Bachelor's Degree in Electronics and Communication Engineering from University of Madras. He is having experience of more than 27 years out of which more than 24 years in training of technical teachers in India and abroad.His area of interest includes Telemedicine, Image Processing, Communication, IoT, Virtual Instrumentation, ICT applications in Teaching/Learning. He has published/presented many papers in the National/International Journals/Conferences.

Dr.P.Ezhilarasu received the B.E degree in Computer Science and Engineering from Bharathiar University Coimbatore. Then M.E degree in Computer Science and Engineering from Anna University,Chennai. He commended his Ph.D in Faculty of Information and Communication from Anna University Chennai .He is having two decades of academic experience. He guided so many UG and PG students in his career. He handled more than 30 subjects in his career and also having more than 80 international/national publication in conference/journal. His area of interest includes Image Processing, Theory of Computation and Data science.

Dr.V.Ulagamuthalvi, Associate Professor, department of Computer Science and Engineering, Sathyabama Institute of Science and Engineering received her Ph.D degrees in Medical Image Processing from sathyabama University. She is having experience of more than 20 years in teaching. Her area of interest are Data Mining, Image Processing, Big Data Analysis.