An Efficient Indexing Technique for Big Data Analytics using Ranking Information

Bassey, Aniefiok Tom, Bennett, E. O., Emmah, V. T.
Department of Computer Science,
Rivers State University,
Port Harcourt,
Nigeria
basseyaniefiok551@yahoo.com, bennett.okoni@ust.edu.ng, victor.emmah@ust.edu.ng

Abstract—The exponential growth of semi structured and unstructured data in the world today which is known as Big Data requires an efficient indexing technique for fast and easy retrieval of information. Individuals and organizations that utilize this data find it difficult to extract these information(s) at a low latency and this results in inefficiency of the system. In this research, two major techniques are combined to achieve the research objective. However, before the combination of these two techniques, the unstructured data are first converted to structured data. The method used in the conversion is called Bag of Words (BoW). After the conversion, an indexing technique called inverted index method is used to index the data in the document or file. Secondly, the indexed file is ranked using vector space model of ranking. Languages used to achieve objectives are PHP and MySQL. When a query is sent, the similarity between the query vector and the document vector determines the retrieval of information expressed by the user as well as the frequency of occurrence of index key.

Index Terms—*Analytics, Big Data, Big Data Analytics, Efficient, Indexing, Ranking Information, Technique*

1 INTRODUCTION

Everyday, trillion, quintillion bytes of data are produced in the world today. The speed of growth is exponential and it is necessary to analyze such massive data effectively and efficiently. Apart from being massive, this data is heterogeneous in nature as well [6]. Big data has been described by some Data Management Scholars as "overwhelming, huge, and uncontrollable amounts of information." In 1663, John Graunt dealt with "overwhelming amounts of information" as well, while he studied the bubonic plague, which was currently ravaging Europe. Graunt used statistics and is credited with being the first person to use statistical data analysis. In the early 1800s, the field of statistics expanded to include collecting and analyzing data. With the massive explosion in volume of available data, the challenges are how to separate the signal of data and the valuable information. Unfortunately, at this point, a lot of companies, establishments have difficulties in identifying the right data and determining how best to use it. The fight against unprompted data and data quality is a crucial problem. Therefore it is necessary for Companies to think out of the box and look for revenue models that are very different from traditional business.

Also, the most obvious challenge associated with big data is simply storing and analyzing all that information. Extraction of useful information from all the available online information can be difficult due to the volume and variety (structured/unstructured) of data. However, the foundation of good analytical framework relies totally on the quality and quantity of data. The need for techniques which can access and search data item speedily becomes important and indexing strategy is used to design an access method to a searched item. Ranking information (information retrieval) is sorting of information according to some conditions so that the best result appears early in the result list displayed to the user. Classically, ranking criteria or condition are phrased in terms of relevance of documents with respect to an information need expressed in the query [1].

Ranking Information has been widely used in a number of massive and complex data intensive fields such as search engine, cloud computing, social media, etc. These techniques

provide possible solutions to mine the information hidden in the data. Most traditional ranking information is designed for data that would be stored in an index for efficient retrieval [2]. However, the overall aim of the ranking process is to return the best results for the user based on the underlying intent.

2 RELATED LITERATURE

Big data is a wide notion. It was first proposed by a famous consulting company, McKinsey. In 2001, big data was described by META Group (now Gartner) analyst, Doug Laney, as being three dimensional:

i. Volume (amount of data),
ii. Velocity (speed of data in and out), and
iii. Variety (range of data types and sources), [6].

The "3V" (Volume, Velocity, and Variety) definition was passionately used by many Firms. After, Gartner updated it by noting that "Big data has high volume, high velocity, and high variety information assets that require new method of processing to enable enhancement of decision making, insight discovery and process optimization [9].

In [8] a new behaviour named "Veracity" was added to it. Today, research on big data has sheltered nearly every area, such as economic productivity, genomics, and biological researches. The processing scheme of big data can be divided into six parts: Acquisition and pre-processing, storage and management, computing, analyzing and data mining, visualization, privacy and security.

At the moment, big data is attracting more concerns because with the high development speed of information technology, massive amount of data produced every minute from different sources like internet, network communication, financial institution, biological industries etc has become a latest and vital factor of production. Even with these opportunities, it also brings new challenges in technology and changes in subjects. Since 1980, the storage space of data has increased by double every 40 months [4]. As at 2012, 2.5 exabytes (2.5×10^{18}) bytes of data were created on daily basis and it is predicted that the traffic flow of data over the internet will attain 667 exabytes (6.67×10^{20}) per annum in 2023 [5]. The idea of using computers to search for relevant information was popularized by Vannevar Bush in 1945. Bush was inspired by patents for a statistical machine file invented by Emanuel Goldberg in 1920s and '30s that searched for documents stored on film.

The first description of computer searching for information was described by Holmstrom in 1948, detailing the Univac computer.

Automated information retrieval systems were introduced in the 1950s which one was featured in 1957 romantic comedy desk Set. In the 1960s, the first large information retrieval research group was formed by Gerard Salton at Cornell. By 1970s, numerous different retrieval techniques were shown to perform well on small text corpora such as the Cranfield collections (several thousand documents) [11].

The aim here is to research and come up with a technique that can index big data and rank it based on criteria for easy retrieval and with a low latency. Implementation of encryption and decryption for a secure index construction was completely done with attractive performance [10]. After index construction is done, it gets compressed and is stored in .cfs file format. When single keyword query is fired, user gets documents that hold the specified keyword. The advantages of this system are it protects data privacy by encrypting documents before outsourcing ranked documents, and it is easy to access the encrypted data by multi keyword rank search using keyword index. The Disadvantages of the system are single keyword search without ranking, Boolean keyword searching without ranking, single-keyword search with ranking, rarely sorting of results i.e. no index creation and ranking, Single User search. A fresh construction of a public key searchable encryption scheme using inverted index was proposed [12]. This scheme overcomes the one-time-only search limitation. In this work, probabilistic trapdoor generation algorithm was used to check the cloud server from linking the trapdoors. The disadvantages of the system are first, keyword privacy is compromised once a keyword is searched. This cause the index to be rebuilt for the keyword searched. Such solution is counterproductive as the operating cost is high. Secondly, the existing inverted index does not support conjunctive multi-keyword search, which is the most common form of queries now a days. The advantages are, it explores the problem of building a searchable encryption scheme based on the inverted index, it achieves secure and private matching between the query trapdoor and the secure index, and it designs a new trapdoor generation algorithm so that query related inverted lists are combined together secretly without letting the cloud server know which inverted lists are retrieved.

3 METHODOLOGY

This type of research is common in computer science research procedures. The constructive approach is simply problem solving through the construction or use of models, diagrams, plans, etc. This pattern of research is generally used in technical sciences, operations analysis, mathematics, clinical medicine and in operations research [3]. The word "construct" frequently used in research is indicating a new knowledge build. It can be a new theory, model, software, algorithm, or a framework. Mathematical algorithms and new mathematical entities present theoretical examples of constructions. This type of technique requires a form of test that does not need to be silent as experimentally based type. The results should be objectively defined as this may include assessing the "construct" developed systematically against some predefined conditions. Constructive method solves practical problems

and also produces an academically valued theoretical contribution [7]. However, the elements of constructive method used in this research include:

Relevant Problem: The massive growth in volume, velocity, and variety of data generated by mobile devices and cloud applications etc, has cause abundance of data or 'big data.' The World Wide Web (WWW) is a useful and communicating device of information like hypertext, multimedia etc. On searching for any information on the Google, many Unified Resource Locators (URL's) are opened. Because of large amount of information online, the users find it difficult to extract and filter the relevant information at ease.

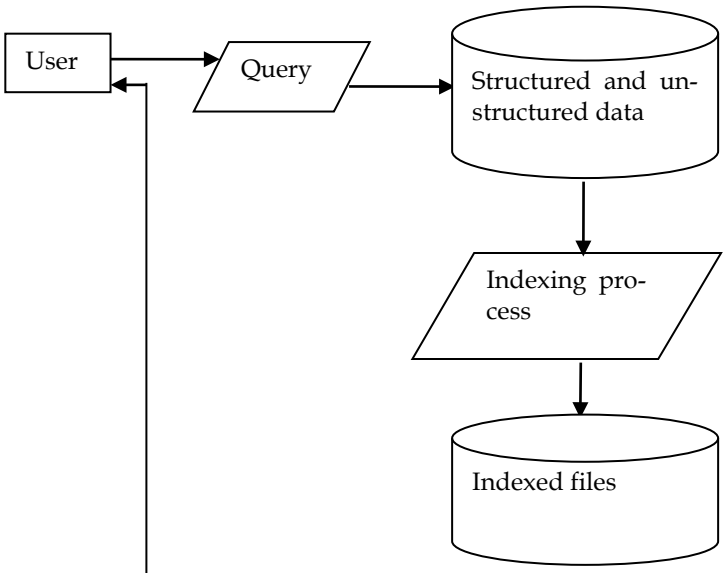
Problem Solution: This research intends to solve the problem of difficulties in information retrieval in big data by developing an efficient indexing technique using ranking information technology as criteria which will help in fast retreival of information, sorting and management of information based on ranking.

Practical Relevance: This research will be useful as a model for understanding indexing techniques. However, organizations, businesses, etc will benefit from this resesarch as it will help them organise and retrieve information based on their content.

Theoretical Relevance: This is formed to explain, predict, understand phenomena and, even challenge previous knowledge within the areas of assumptions. The theoretical framework of this research will help future reseachers to have a deeper understanding of indexing techniques for quick and easy information retrieval.

4 System Architecture

It shows a deeper description of structure and activities of a system. Figure 1 shows the proposed system architecture.



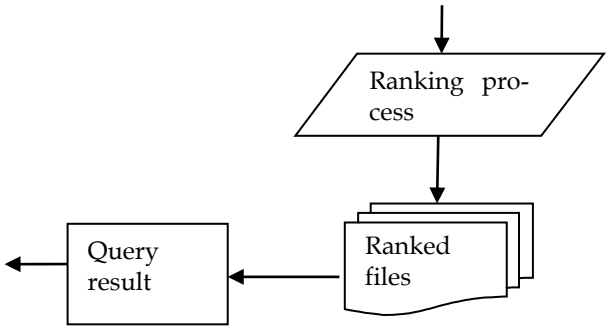


Figure1. Proposed System Architecture

System architecture consists of system components and subsystems urbanized, to work together in other to implement the overall intention of the proposed system. The major components here are the user, search engine/ranking system, indexer, query pre-processing and database which is responsible for the storage of information.

In the proposed system, from the input point, unstructured data need to be converted to structured data before indexing for efficient retrieval of information when a query is sent.

CONVERSION PROCESS:

There are different techniques used in converting the unstructured data to structured data for easy retrieval of information in big data. However, the conversion model used in this research is Bag of Words (BoW).

In Bag of words, a sentence or a document is seen and considered as a 'Bag' containing words. It takes into account the words and their frequency of occurrence in the sentence or in the document without minding semantic relationship in the sentences.

This technique uses pattern matching style to find out the order in text. It assists to transform the unstructured text form into structured text form. Example of how Bag of words work:

Suppose there are four documents named:

Doc1. I love Cricket
Doc2. I love Football
Doc3. I love Hockey
Doc4. I love Golf

Each document is treated as a standalone entity and it makes a list of all words from all the four documents excluding punctuations. This result to,

"I", "Love"," Cricket"," Football"," Hockey"," Golf"

The next step is to create vectors. Vectors convert text that can be used by the Machine Learning. So, for the first document it is as follow:

I = 1

Love = 1

Cricket = 1

Football = 0

Hockey = 0

Golf = 0

So, following documents are represented as follow:

1. I love Cricket: [1,1,1,0,0,0]
2. I love Football: [1,1,0,1,0,0]
3. I love Hockey: [1,1,0,0,1,0]
4. I love Golf: [1,1,0,0,0,1]

INDEXING PROCESS:
The documents crawled by the search engine are indexed for proficient retrieval. The documents are first parsed, and then tokenized; stop word(s) are removed and stemmed. After all, it is stored in an inverted index. The practice is discussed below:

Tokenization
This is extracting word tokens from running text. By illustration, given a piece of text: 'I Love Football' it will output 'I', 'Love', 'Football' as tokens.

Stop-word remover
Stop words are words with no potential power. Common examples of stop words are articles, prepositions etc. It is the ability to remove these words in a running document or sentence from the list of tokens. By example, if a document is given as 'hotels are places of luxury', it reduces it to: 'hotels', 'places', 'luxury'.

Stemming
This is reducing tokens to the root form of it. For example, the token 'hotels,' becomes 'hotel'. Note this happens after stop words are removed.
The indexing technique used in this research is inverted indexing.

Inverted index in big data is an index data structure storing a mapping from content, to its locations in a database file, or a set of documents.
Inverted Indexing improves the speed of data retrieval operation in database. The main idea of having index is to speed up search queries. The different between the ordinary index and the inverted index is that in ordinary index, for each document, only the index term is found while for inverted index, the list of documents and its location is seen for each term. It is of paramount advantage to use the inverted indexing in the research since the system is interested in searching for documents that contains the index terms in the query. Inverted index may contain additional information like how many times the term appears in the document.

For example, there are four documents.
Doc1. I love Cricket
Doc2. I love Football
Doc3. I love Hockey
Doc4. I love Golf

After stop word removal and stemming to the root form of the token, inverted index that looks like what is publicized in table1takes place.

Table 1: Words and Frequency of Occurrence

S/N	Words	Frequency of occurrence in document
1.	Cricket	1
2.	Love	1,2,3,4
3.	Football	2
4.	Golf	4
5.	Hockey	3
6.	I	1,2,3,4

Ranking Process:
The indexed files are consulted to get the document most similar to the query when user gives a query. The most similar words are ranked according to the degree of similarity and relevancy. The proposed system uses vector-based model of ranking to determine the degree of similarity and importance.
In Vector Based Model, documents are regarded as vector of index terms

$$dj = \{aij , aij , \ldots , atj\} \quad (1)$$

(*dj* is vector document of *j*,*t* is total number of index term in document *j* and *i* is index term in document *j*). '' (1)''
Queries are regarded as vectors in similar way like the documents. The similarity among the query vector and document

vector becomes the determinant of relevancy of the document that is used as ranking score. However, the similarity among document vector and query vector is calculated as cosine similarity among them. Note that if the similarity is greater than a predefined margin, the document is retrieved.

Vector Based Ranking Process:

Vector based model of Information Retrieval (IR) tries to cluster set of documents into two namely: document related to query and document unrelated to query. For this balance to be fixed and kept, the term weights in the document and query vector which represent importance of term for expressing the meaning of the document and query are defined. However, there are two factors used universally in calculating term weights.

i. Term Frequency, (*tf*)
ii. Inverse document frequency, (*idf*)

Mathematically, the term-frequency of a term **i** in document ***j,*** is given by:

$$tf_{ij} = \frac{\text{freq}_{i,j}}{\max_l(freq_{i,j})} \quad (2)$$

. The inverse document frequency (*idf*) of a term *i*, is given by:

$$idf_i = \log \frac{N}{n_i} \quad (3)$$

(Where ***ni*** is the number of documents that ith term occurs and ***N*** is the total number of documents). '' (3)''

After fixing the term-weights, the document and query vectors in **h** dimension (where **h** is number of index terms in vocabulary), becomes necessary to find their similarity. The degree of similarity used is called the cosine similarity. Cosine similarity between two vectors **dj,** (the document vector) and **q,** (query vector) is given by:

$$Similarity\ (\vec{d}_j, \vec{q}) = \cos\theta \frac{\vec{d}, \vec{q}}{\left|\vec{d}\right| \left|\vec{q}\right|} \quad (4)$$

(Here, ***θ*** is the angle between the two vectors,*d,q* are vectors of documents and queries). '' (4)''

5 Results and Discussion

The system was implemented using tools earlier mentioned and data was gotten from a raw source file as shown in figure 2. Results of system are tabulated in Table 2.

Table 2: Indexed and Ranked Result

Figure 2: Raw Source File

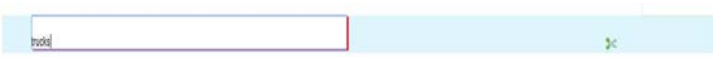


Figure 3: First Extracted Document with Index Key 'Trucks'.

This screenshot shows the extracting content of one document that is indexed with an index key ''trucks'' and the result is as

seen in the screenshot. It appears 6 times in line 3,5,6 and line 7. The chart below shows specifically how the indexed key appears; it appears 1 time in line 3, 2 times in line 5, 2 times in line 6 and once in 7. See the chart below with index key "trucks".

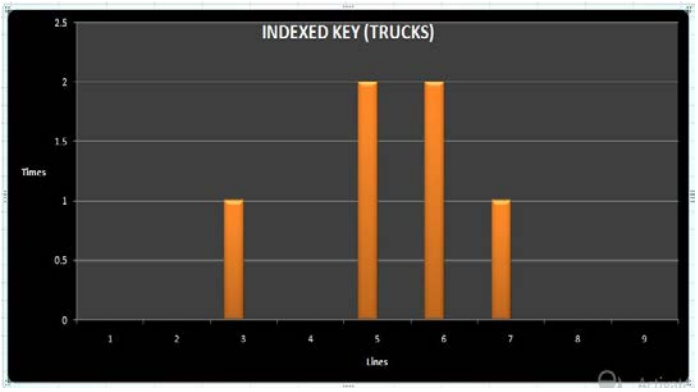


Figure 4: Chart, First Extracted Document

Data mining applications need partitioning of data into uniform clusters which interested groups has discovered. To do such analyses, the following problems must be solved; efficient partitioning of a large data set into uniform groups or clusters, and effective interpretation of the clusters. The system when tested was able to perform the aforementioned aim "textual data indexing and retrieving".

Figure 2 which is the source file page shows all data and records in the database where user can extract any to perform the indexing technique. The page contains actions like 'Read', 'Download', and 'Extract document contents'. However, the size of a file, the date of loading, the title of the file and the description of the file is indicated. Figure 3 shows the extracted contents of particular documents using index key 'trucks'. Once a file is extracted from the source file, all data (tokens) automatically are indexed while figure 4 shows the chart of figure 3 result. At this point, the user types in the index key on the text area provided and it filters the word showing the frequency of occurrence of that particular index key and the line(s) it appears. It was tried on four different documents and the result is represented in table 2 which carries the index key, file name, frequency of occurrence of the token, the lines it appears and the size of the file. Figure 5 shows the graph of time (ms) against size (k) and its correlation. The bigger the size of the file, the more time it takes to index and rank.

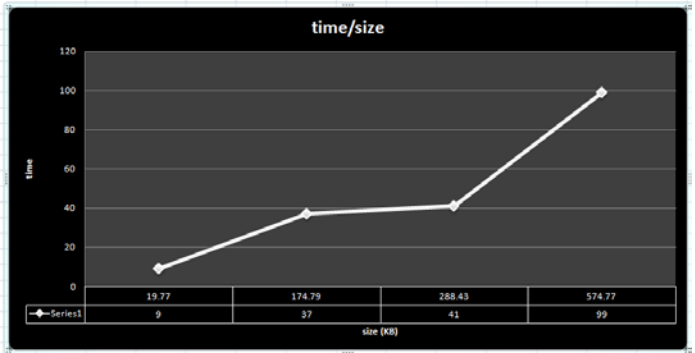


Figure 5: Time Graph against Size

6 Conclusion

Although structured and unstructured indexing system for information retrieval has been considered for years, it is still tedious for most commercial search engines to index multimedia data. It is as a result of many elemental limitations in statistical method used in information retrieval technologies when applied to large scale unstructured data. Here a proposed framework for indexing and ranking are combined to achieve the aim of the proposed system. This framework is used for any type of unstructured textual data using variety of queries (metadata, keyword) to attain suitable results. This system is therefore a good starting point to researchers interested in developing more efficient indexing technique for Big Data analytics.

References

[1] Achlioptas, D. (2001,) Using Language Models for Information Retrieval. Netherlands TaaluitgeverijNesliaPaniculata.

[2] Chen, G., Wong, A. and Yang C.S. (2014). A Vector Space Model for Automatic Indexing. Communications of the ACM, Vol. 18, No. 11, pp 613-620.

[3] Eero,O., Shanghai,L., Hugei, M., and Antony, X. (2009). Designing a Strategic Information Systems Planning Methodology for Malaysian Institutes of Higher Learning (isp- ipta), *Issues in Information System*, 6(5).

[4] Hilbert, O. (2011). The World's Technological Capacity to Store, Communicate, and Compute Information, American Association for the Advancement of Science, 1200 New York Avenue NW, Washington, 554-560.

[5] Jacobs, J, and Croft W.B. (2009) A Language Modeling Approach to Information Retrieval. In Proceedings of the 21st International Conference on Research and Development in Information Retrieval, pp.275–281.

[6] Laney, D. (2001). 3-D management: Controlling data volume, velocity and variety, 121-130.

[7] Liisa, J., Pavlo, A., Tu, S., Stonebraker, M., &Zdonik, S. (2013). Anticaching: a new approach to database management system architecture. In Proceedings of the VLDB Endow-ment (1942–1953).

[8] Manyika S.E, Walker S, Jones S, Hancock-Beaulieu M, and Gatford M. (2011) Okapi at TREC-3. Proceedings of the Third Text Retrieval Conference, Gaithersburg, US.

[9] Mukherjee, S., & Shaw, R. (2017) Big Data–Concepts, Applications, Challenges and Future Scope. International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 2.

[10] Raneet.al, B.S. and Jung, J. (2015) A Novel Ranking Model for a Large-Scale Scientific Publication Mobile Networks and Applications, Vol. 20, Issue 4, PP 508-520.

[11] Singhal. P., Gupta. A. and Dixit A, (2001). Comparative Study of

HITS and PageRank Link Based Ranking Algorithms. International Journal of Advanced Research in Computer and Communication Engineering, Vol 3, Issue 2, PP. 5749 - 5754.

[12] Wang, Y. Tong, Y. and Zeng, M (2014) Ranking Scientific Articles by Exploiting Citations, Authors, Journals, and Time Information. Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, Bellevue, Washington, USA, PP. 933 -939.

S