Improved Algorithm for Proficient Storing and Retrieving of Medical Data Records in A Data Lake

Dilli Babu M.[1]
[1]Research Scholar, Department of Computer Science and Engineering,
Hindustan Institute of Technology and Science,
Chennai, Tamil Nadu 603103, India
deenshadilli@gmail.com

Sambath M.[2]
[2]Associate Professor, Department of Computer Science and Engineering,
Hindustan Institute of Technology and Science,
Chennai, Tamil Nadu 603103, India
msambath@hindustanuniv.ac.in

Abstract:

A data lake is a prevailing concept widely accepted nowadays by healthcare industry due to the widespread accessibility of information in diverse formats. Today's medical examination involves a lot of medical images to be viewed by the physicians. The physical medical film viewing is quite cumbersome and may not be accurate at times. So this research work provides ways for storing the medical images in electronic medium in a data lake thereby decreasing the storage for these records and makes the retrieval process much easier. The storage and retrieval of electronic medical images though challenging, the uniqueness in each medical image, the size of the image makes the process to proceed in the right direction. The proposed method is based on improving the storage of medical images and accurate retrieval of medical images from a data lake. This paper proposes methods for efficient improvement in storage of electronic medical images and access techniques related to Hadoop. The uniqueness in the medical images is taken into consideration for the integration of diverse medical images with easy access availability of medical image files. A hybrid algorithm involving the first come first serve (FCFS) and least recently used (LRU) algorithm is proposed for the effective retrieval of medical image files from a data lake.

Keywords: Data Lake, medical images, Hadoop, FCFS, LRU

Introduction

Data Lake is a rising technology to facilitate data management in data analytics. The data lake follows flat kind of architecture to store data in a different raw format [1]. The data that is stored in the data lake must have proper means for storing, retrieving and analyzing the data. If the data present in the data is not properly interpreted it will lead to enormous loss of information that is stored. Moreover data swamp may result due to the improper usages of data in the data lake [2]. Particularly in the analysis of medical records, the data need to be handled in a very intelligible manner. The medical records storage, retrieval and analysis is an important task for the data analyzer.

Good medical infrastructure is a significant element in a society. The storage of patient data, medical records, scanning reports play a vital role in the effective management of records in hospitals. Making these records to be in electronic mode is the current trend in almost all leading developed and developing countries. The main purpose of electronic medical records is to make its storage consume very less space and its retrieval to be fast. Huge electronic medical imaging figures offer dominant statistics sustainability designed for intellectual supplementary analysis algorithms. Furthermore it creates challenges intended towards multifaceted intricate storage of medical records and its retrieval [3].The medical system will be greatly improved by the digitization of medical records. The electronic medical images are used by doctors for easy diagnosis of diseases. These images can also be viewed by patients through application tools. Based on the size of the medical images and magnitude of the medical images, the storage capacity varies invariably. The number of medical images produced during a single medical test depends on the

type of scan. For example, large number of medical images is produced in the Computed tomography (CT) scan with too small image size. But in Computed radiography (CR) and digital radiography (DR) scan, little number of medical images is produced with too large image size. The Hadoop distributed file system (HDFS) can be used for the storage and management of very enormous data. But it doesn't work fine in case of huge arbitrary read and write.

The electronic health records (EHRs) include text transcriptions made by a number of doctors, structured data from sensors and measurements, and unstructured data from video and image (MRI, X-RAY). These resources necessitate the creation of methods for structuring unstructured data so that it can be integrated. In the textual portions of EHRs, a variety of information extraction techniques have been used. By encoding clinical notes into a feature matrix using medical terminologies, information extraction was employed to find signs of drug safety.

In the developed world, almost everyone has a medical history. Medical records gather sensitive and essential information about patients, their health, the ailments they have, and the treatments they get for those conditions.These details may be found in medical records:Social security numbers, addresses, and identification that is exclusive to the healthcare provider are examples of personal identifying information (PII), commonly referred to as private health information or PHI. A person's medical history may be kept on file without their knowledge. For instance, if a person hasn't had immunizations, the medical records may reflect this. Any diagnostic, current and prior medical issues, and potentially vital, life-saving information like allergies or drug sensitivity are all included in the medical history. Relevant medical issues that have been noted for immediate family members that may be congenital or have another effect on their wellbeing are also included in a person's medical records.Medication history is that substances a person has used, including prescription pharmaceuticals, over-the-counter medications, herbal remedies, and even illegal drugs. Treatment history is the person's past medical interventions and how they affected their health.Health records may also include a "living will," which is a set of instructions that a patient intends to convey to medical personnel in the event that they are unable to speak for themselves. Thirty years ago, the majority of medical records were written on paper and kept in hospital archives or doctors' filing cabinets. In many medical settings, patient records are still largely kept on paper. However, medical records are becoming more and more digital, and the healthcare sector currently manages three main categories of digital health information.

EMR (Electronic Medical Record):EMR is a technology that enables doctors to keep track of patient visits, to put it simply. The EMR contains a variety of information regarding the ailments or complaints of patients, the treatments they have undergone and the outcomes, as well as billing details pertaining to the medical services rendered.

EHR (Electronic Health Record) :EHR is a more comprehensive system that contains data from several medical facilities. By including data from wearable technology, public medical research, health-related databases, and demographic or survey information that can illuminate patient wellbeing, it enriches EMR data. EHR is designed to include patients and assist them in sharing more details about their lifestyle and circumstances, which can enhance care for both them and other people with comparable problems.

PHR (Personal Health Record) :PHR is a system that enables people to track their health information and is often run and offered by insurance companies. Usually, it comes in the form of a user-interfaced programme that allows for the storage and utilisation of medical data. This can comprise information entered by the user or personally obtained through wearable technology, medical records added by the user, and data directly imported from EMR systems.

The continual management of medical records is a field that calls for devoted personnel and knowledge in many contemporary healthcare institutions. Management of medical records includes:ensuring the availability, security, and safety of medical records,monitoring digitization projects to make sure they adhere to the organization's objectives, pertinent industry standards, and regulatory requirements,taking care of redundancy, backup, and disaster recovery to maintain the resiliency of medical records,directing IT technologies, including as EMR systems, used to direct and access medical records. EHR is a game-changer because it has made it feasible to mine a massive ocean of data, which opens the door to numerous benefits like precision medicine but also calls for a deep understanding of both fields. Precision medicine, which is regarded as one of the key benefits of data mining, is intended to deal with a course of care that is specifically suited to each patient. Finding increasingly targeted treatments using EHR data is another option, one that naturally fits into today's

practise paradigm, even though precision medicine includes a significant genomic component.

Related Work

The data lake is an economical storage methodology that works on Hadoop. The data lake architecture provides solutions in terms of software, hardware and abstract design feature [4]. Data Lake is not like a data warehouse. The data warehouse is based on relational database management system. But Data Lake is more associated with diverse data and supports variety of datasets [5]. Highly significant data need to be acquired in the healthcare segment. To assist for the effective utilization of medical data records, its storage and easy access need to be processed in a better way [6]. Network-based model was employed to extort information from data lake[7]. For the analysis of data in the Data Lake, application tools were used for storing the metadata in a central storage for easy access of the data [8].

Recently Hadoop have turned out to be one of the majority widely accepted distributed computing model meant for outsized data analytics. Hadoop can store up enormously huge files. The Hadoop Distributed File System (HDFS) works on master and slave architechture [9]. Hadoop is exceptional in handling data files of very large size. The HDFS separates the huge data into smaller blocks of data. The metadata for the blocks of data are stored in the NameNode. The large real input data are stored in the DataNodes. MapReduce is involved in the processing of the blocks of data. Amid the escalating file sizes the Hadoop becomes incompetent in managing several files of small size [10].

The abundant diminutive files put forward severe performance concern with Hadoop for the reason that the store up of several files of small size in the Hadoop turn out to be a memory overhead. These huge enormous files of small size occupy most of the main memory in the NameNode. The Hadoop becomes inefficient when handling files that interact between them less frequent. This is due to the fact that Hadoop is made up to analyze large big data instead of transmitting several files [11]. A SequenceFile offers an unrelenting data structure in favor of binary key-value pairs. The key is the file name and the value is the information available in the file. A sequence file can hold several small files. The MapReduce processes these sequence files [12]. The other kind of file is the MapFile. The MapFile is also a sequenceFile but it is sorted in order. In MapFile the key uses an index for its access operation. This MapFile is advantages due to fact that it need not explore the whole file instead it can lookout for a index using the key [12]. A MapFile is a type of sorted SequenceFiles with an index to lookup operation by key. It consists of two files, a data file and a smaller index file. All of the sorted key-value pairs are stored in the data file and the key location information is stored in index file. MapFile does not search for entire file when looking for a specific key.

The two main strategies for securing privacy in EMR are access control techniques and data protection measures. Data encryption, privacy anonymous processing, and access control are among the technical concerns[13]. Additionally, the privacy protection system for EMR systems is being steadily built with an emphasis on privacy and sensitive data. Additionally, due to the prominence of SDN technology, security concerns and associated problems have garnered a lot of attention [14–16].

Clinical notes, which are unstructured EHR data, can be processed to assist medical professionals in describing and rationalising potential adverse medication occurrences (ADEs). The authors in [17] used distributional semantics models, which are unsupervised techniques that take advantage of cooccurrence information to model, typically in vector space, the meaning of words and, in particular, combinations of such models to improve the predictive performance, to learn to recognise information about ADEs present in clinical notes. The relationship between diseases, the relationship between diseases and medical examinations, and the relationship between diseases and therapy are three categories into which the entity relations in EMR can be categorized [18]. Additionally, the entity relation is only allowed to exist between two named entities that are present in a sentence. Three widely used techniques are used in the medical industry to extract entity relations: cooccurrence-based [19], pattern-based, and machine learning approaches.

A suitable time layout of the sequence of important events should be compiled and used to generate a patient-specific timeline, which could further support medical staff in making clinical decisions, in order to make an accurate and valid assessment of patient discharge reports. A hybrid method [20], combines two approaches—one is rule-based, and the other is based on the maximum entropy model—to discover relevant temporal linkages between a pair of things. According to the experiment's findings, the proposed system's F-score of 0.563, which was at least 30% higher than the baseline system's, was reached. A

system that combines rule-based and machine learning approaches was presented in [21] to automatically extract temporal expressions and events from clinical narratives. Conditional random field models were trained for event and temporal recognition, while rule-based components were created to handle the identification and normalisation of temporal expressions. An open-source natural language processing system for information extraction from clinical free-text in electronic medical records was presented in [22]. They developed the open-source clinical Text Analysis and Knowledge Extraction System (cTAKES), a modular system of pipelined components that combines rule-based and machine learning techniques.

Monitoring the development of risk factors over time in EMRs could help medical professionals make clinical decisions and facilitate data modelling and biomedical research. The right associations might not always be provided by this strategy, though. As a result, the authors in [23] developed a context-aware method for assigning the time characteristics of known risk variables by reconstructing contexts that contain more trustworthy temporal expressions.

Healthcare data frequently has a wide range of data. The data lake is therefore considered to be a promising remedy for combining various medical data. To analyse the cost of care, for instance, the traditional data warehouse was frequently employed. Data from EHRs and claims must be integrated for advanced cost analysis. In a traditional data warehouse, the process started with the data being unloaded, followed by a new extract, convert, and load procedure. However, a data lake streamlines the process of adding a new data source or sophisticated activity (queries, algorithms, etc.) [24].Data lake is a developed Big data consolidation solution, according to Krause [25].

Methods

The proposed method focuses on improving the storage space for medical image files. The storage space of the medical files is effectively utilized using the proposed integration approach applied for diverse medical image files. The random access to the small medical image files are improved with the application of the proposed model which uses a derived hierarchical based indexing scheme.

In general, during a normal patient examination, the patient could be advised by doctors to take certain scan of their body parts. A patient could be subjected to different types of medical tests. The medical tests may involve a sequence of medical report generated. These medical reports in turn contain a number of medical images. Figure 1 shows the representation of hierarchy of medical images produced during the patient test. In Figure 1, 'IS' is the Image Sequence and 'Img' is the Image file.

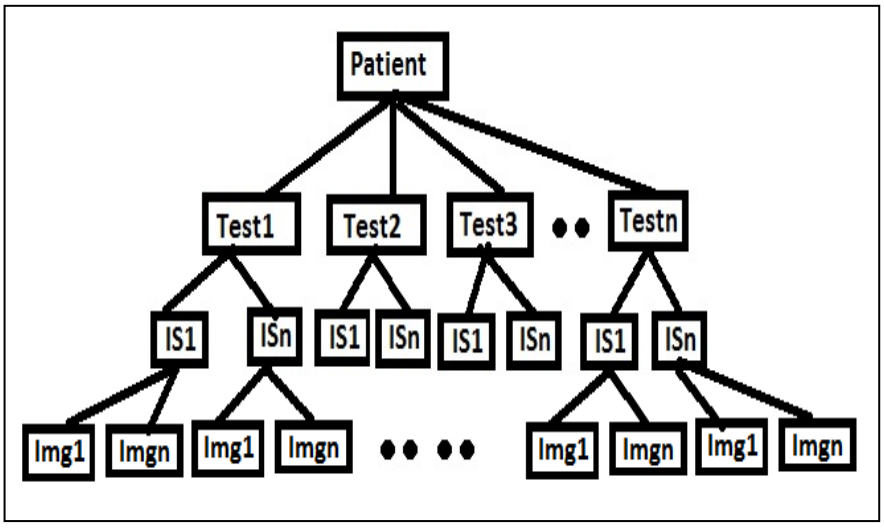


Figure 1: Hierarchical structure of patient record

In the first level, all the patient related documents will be stored and processed. The second level contains the patients test records. The third below level contains the Image sequences produced during the test taken by the patient. The beneath fourth level contains individual medical Images from the Image sequence. The different images under same Image sequences have well-built association between the them. There also exist strong bondage among the varied Image sequences under the similar test taken by a particular

patient. However this relationship between the Images and Image sequences will not be projected in the Hadoop architechture.So there is a need to propose methods to highlight this intergration. Hence a hierarchical based indexing is done that reflects the relationship among the medical images and Image sequences.

In the traditional Hadoop storage, the patient test file will be stored as two data blocks as a 128MB SEQ. Hence there arises a need to read the data across these two blocks for accessing a patient test file. The proposed method utilizes only one data block for its data storage. The medical images are grouped into two. One group of medical images constitutes the MR and CT scanned medical images. Here the Image sequences form the vital entity for doctors to proceed with their diagnosis. The other grouping constitutes the CR and DR scanned medical images. Here a single medical image file forms the vital entity. The integration approach utilizes the Medical data bucket (MDB) storage storing 128 MB patient testing data. The current patient test file will be added to check bucket (CB) storage. The size of the medical data bucket and the Check bucket is checked against a predetermined threshold. If the size exceeds the threshold, the data is stored in a new File bucket (FB). This integration approach mainly avoids the extra data block storage in HDFS.

Algorithm-Integration approach algorithm

MDB-Medical data bucket, MDIB-medical data index bucket,CB-check bucket, FB-file bucket

Step 1: Preprocessing

Step 2: Patient test file merged based on its file type

Step 3: Medical Image files removed from MDB

Step 4: The current patient test file added to CB

Step 4: Index generated for medical image file and stored in MDIB

Step 5: if (size > threshold), create new FB else store in HDFS

The integration approach starts with preprocessing where the patient test files are merged based on its file types. The merged patient test files are stored in the Medical data bucket (MDB). Then the indexes are generated for the particular current medical image file and stored in the medical data index bucket (MDIB). The size of the current patient test file is checked against a predetermined threshold. If the size is greater than the threshold, it is stored in a new file bucket (FB). If the size is lesser than the threshold it is stored in the HDFS.

The file access can further be improved by prefetching the caches and reading it through two diverse queues. One queue following the first in first out (FIFO) approach and the other queue following the least recently used (LRU) approach. The data is first stored in the FIFO queue until the FIFO queue becomes full. If the FIFO queue becomes full then the data will be deleted in the first-in-last-out approach. If there arises a need to access the data again in the FIFO queue, the FIFO buffer is shifted to the LRU queue. If there arises a need to access the data again in the LRU queue, the data will be moved to the end of the queue. If the LRU queue becomes full then the LRU queue will be deleted

Algorithm: Prefetch-Cache algorithm

Input: data D

Step 1: begin

Step 2: if FIFO queue not full

Step 3: add D

Step 4: if FIFO queue full

Step 5: delete bottom D from FIFO queue and insert it to LRU queue

Step 6: if LRU queue full

Step 7: delete data from LRU queue

Step 8: endif

Step 9: add D in LRU queue

Step 10: endif

Step 11: end

Experimental Results

The medical image files are collected from hospitals. The patient medical image file is a DICOM file. The attributes in the DICOM file is checked as a preprocessing step. The patient medical image files are arranged as per the hierarchical structure in figure 1. For the experimentation, we used 720,342 Computed tomography (CT) scan images, 5322 Computed radiography (CR) and 3643 Ultra-sonic (US) images.

Table 1: Files merged

Scan type	SEQ	Proposed algorithm
Computed tomography (CT)	2217	1093
Computed radiography(CR)	1090	87
Ultra-sonic (US)	2233	226

We determine the merging of the files for both the SEQ and the proposed technique. We check the possibilities of merging of large files among these two techniques. It was found that the proposed system was able to merge large files than compared with the existing SEQ technique. The merging factor can be determined using the Big File Ratio (BFR). Table 2 gives the BFR values of the SEQ and the proposed technique. The SEQ technique very negligibly merged very large files. In the case of Computed tomography images, the SEQ technique was able to merge large files only to a few percentages of nearly 20%. Whereas in case of merging large files in Computed radiography and Ultra-sonic images, the percentage is almost zero percentage. But the proposed algorithm was able to merge very large file size in all the three different scan types of Computed tomography, Computed radiography and Ultra-sonic medical images. In the case of Computed tomography medical report images, the proposed algorithm merged almost 74.24 percentages of very large files. Similarly in Computed radiography medical report images, the proposed algorithm merged almost 99.49 percentages of very large files. The merging of large Ultra-sonic image files was up to 96.12 percentages. This shows the effectiveness of the proposed algorithm in merging very large medical image files.

Table 2: BFR values

Scan type	SEQ	Proposed algorithm

Computed tomography (CT)	20.05%	74.24%
Computed radiography(CR)	0%	99.49%
Ultra-sonic (US)	0%	96.12%

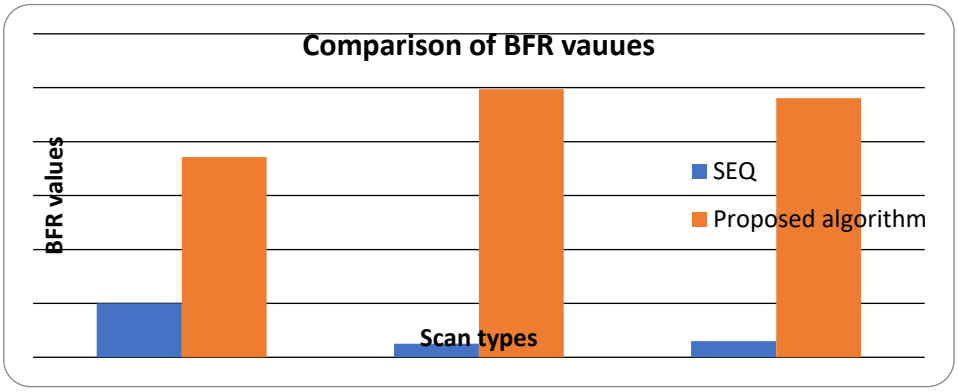
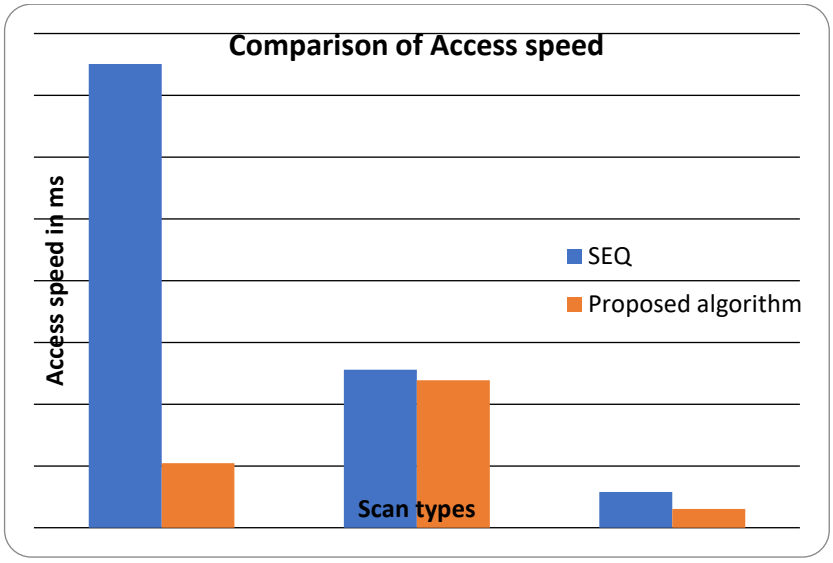


Table 3: File access speed

Scan type	SEQ	Proposed algorithm
Computed tomography (CT)	375.23 ms	52.11 ms
Computed radiography(CR)	128 ms	119.36 ms
Ultra-sonic (US)	29 ms	15.24 ms



We also compute the speed in accessing the Computed tomography, Computed radiography and Ultra-sonic medical images files. Usually the Computed radiography image files are of large size in megabytes and the CR file is about 5MB and the Ultra-sonic image fle is of smaller size in the range of kilobytes. Table 3 represents the access speed in milliseconds while reading the three types of scanned image files namely Computed tomography, Computed radiography and Ultra-sonic. The access speed for

reading Computed tomography in SEQ method is 375.23 ms and in the proposed technique it is 52.11 milliseconds. The access speed for reading Computed radiography in SEQ method is 128 milliseconds and in the proposed technique are 119.36 milliseconds. The access speed for reading Ultra-sonic in SEQ method is 29 milliseconds and in the proposed technique are 15.24 milliseconds.

For any process, its execution time E_t is given in equation (1).

$$E_t = IO_{access}(CPU_{time} + IO_{time}) \quad (1)$$

Where IO_{access} gives the count of Input output accessed, CPU_{time} gives the CPU time used by application and IO_{time} gives the time needed to fulfill the input output operation.

Let T_h be the time which hits the cache during a block read, T_k be the time for cache in which the block gets fetched from disk which is the latency involved during the block fetch and T_d be the time involved in buffer allocation, queuing operation, interrupt mechanism . The Tm be the input ouput access missed in the cache which is given in equation (2).

$$T_m = T_h + T_d + T_k \quad (2)$$

At point of time, the part of demand access to facilitate hit within the cache is known by means of the cache-hit ratio, CH(b), where 'b' denotes the buffers used in the cache. With the known CH(b), it is possible to determine the average time needed for an input output request service TL(b), given in equation (3).

TL□b□□= CH□b□Th+ □1 – CH□b□□Tm (3)

Considering the LRU aided buffer used in cache utilizing the LRU replacement policy outcome results in an enhancement of the input output time for the application to be serviced.

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□TL□b□
= TL□ b – 1□□– TL□ b□□

= □CH□
b□□□– CH□ b – 1□□□Tm – Th
□□□□□□□□□□□□□□□□□□□□□□□□□

Without the use of the buffer, the input output time serviced for the application will increase drastically.

Conclusion

This paper provided techniques for storing the medical images in electronic medium in a data lake. The proposed algorithms decreased the storage space of records and the retrieval process was made much easier. The proposed method improved the storage of medical images and produced accurate retrieval of medical images from the data lake. This paper proposed methods for efficient improvement in storage of electronic medical images and as well as access techniques related to Hadoop.. A hybrid algorithm involving the first come first serve (FCFS) and least recently used (LRU) algorithm was proposed for the effective retrieval of medical image files from a data lake. The proposed algorithm produced better access of files than the existing methods.

References

[1] Inmon B (2016), "Data Lake Architecture: Designing the Data Lake and Avoiding the Garbage Dump Technics Publications"

[2] Hai R, Geisler S, Quix C (2016), "Constance: an intelligent data lake system', International Conference on Management of Data, pp. 2097-2100.

[3] X Huang, F Yan, W Xu and M Li (2019), "Multi-attention and incorporating background information model for chest x-ray image report generation," IEEE Access, vol. 7, pp.154808-154817.

[4] Madera C and A Laurent (2016)," The next information architecture evolution: the data lake wave", International Conference on Management of Digital EcoSystems.

[5] Fang H (2015), "Managing data lakes in big data era: What's a data lake and why has it became popular in data management ecosystem", IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems.

[6] Westra Bonnie & Subramanian Amarnath & Hart Colleen & Matney, Susan & Wilson Patricia & Huff Stanley & Huber, Diane & Delaney, Connie. (2010), "Achieving Meaningful Use of Electronic Health Records through the Integration of the Nursing Management Minimum Data Set", The Journal of nursing administration, vol.40, pp. 336-43.

[7] Lo Giudice, Paolo & Musarella, Lorenzo & Sofo, Giuseppe & Ursino, Domenico (2018), "An approach to extracting complex knowledge patterns among concepts belonging to structured, semi-structured and unstructured sources in a data lake", Information Sciences. 478. 10.1016/j.ins.2018.11.052.

[8] Maccioni A and R Torlone (2017), "Crossing the finish line faster when paddling the data lake

with kayak", VLDB Endowment, vol. 10(12),pp. 1853-1856.

[9] Shvachko K, Hairong K, Radia S, Chansler R (2010), "The Hadoop Distributed File System", IEEE Symposium in Mass Storage Systems and Technologies (MSST).

[10] Dong B, Qiu J, and Zheng Q (2010), "A Novel Approach to improving the Efficiency of Storing and Accessing Small Files on Hadoop", IEEE International Conference on Services Computing, pp. 65-72.

[11] Jiang L, Li B and Song M (2010), "The optimization of hdfs based on small files", IEEE International Conference on Broadband Network and Multimedia Technology, pp. 912-915.

[12] White T (2009), The Small Files Problem. http://www.cloudera.com/blog/2009/02/the small files problem.

[13] F. Liu and T. Li, "A clustering K-anonymity privacy-preserving method for wearable IoT devices," Security and Communication Networks, vol. 2018, Article ID 4945152, 8 pages, 2018.

[14] H. Zhang, Z. Cai, Q. Liu, Q. Xiao, Y. Li, and C. F. Cheang,"A survey on security-aware measurement in SDN," Security and Communication Networks, vol. 2018,

[15] J. Xia, Z. Cai, and M. Xu, "An active defense solution for ARP spoofing in OpenFlow network," Chinese Journal of Electronics,vol. 3, 2018.

[16] Y. Li, Z. Cai, and H. Xu, "LLMP: exploiting LLDP for latency measurement in software-defined data center networks,"Journal of Computer Science and Technology,vol. 33, no. 2, pp. 277–285, 2018

[17] B. Tang, H. Cao, Y. Wu, M. Jiang, and H. Xu, "Recognizing clinical entities in hospital discharge summaries using structural support vector machines with word representation features," BMC Medical Informatics and Decision Making, vol. 13, article S1, Supplement 1, 2013.

[18] Ö. Uzuner, B. R. South, S. Shen, and S. L. DuVall, "2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text," Journal of the American Medical Informatics Association,vol. 18, no. 5, pp. 552–556, 2011.

[19] R. Jelier, G. Jenster, L. C. J. Dorssers et al., "Co-occurrence based meta-analysis of scientific texts: retrieving biological relationships between genes," Bioinformatics, vol. 21, no. 9, pp. 2049–2058, 2005.

[20] Y.-C. Chang, H.-J. Dai, J. C.-Y. Wu, J.-M. Chen, R. T.-H. Tsai,and W.-L. Hsu, "TEMPTING system: a hybrid method of rule and machine learning for temporal relation extraction in patient discharge summaries," Journal of Biomedical Informatics,vol. 46, pp. S54–S62, 2013.

[21] A. Kovačević, A. Dehghan, M. Filannino, J. A. Keane, and G. Nenadic, "Combining rules and machine learning for extraction of temporal expressions and events from clinical narratives," Journal of the American Medical Informatics Association,vol. 20, no. 5, pp. 859–866, 2013.

[22] G. K. Savova, J. J. Masanz, P. V. Ogren et al., "Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES):architecture, component evaluation and applications," Journal of the American Medical Informatics Association, vol. 17, no. 5, pp. 507–513, 2010.

[23] N.-W. Chang, H.-J. Dai, J. Jonnagaddala, C.-W. Chen, R. T.-H.Tsai, and W.-L. Hsu, "A context-aware approach for progression tracking of medical concepts in electronic medical records," Journal of Biomedical Informatics, vol. 58, pp. S150–S157, 2015.

[24] J. Roski, G. W. Bo-Linn, and T. A. Andrews, ``Creating value in health care through big data: Opportunities and policy implications," Health Affairs, vol. 33, no. 7, pp. 11151122, Jul. 2014.doi: 10.1377/hlthaff.2014.0147.

[25] D. D. Krause, ``Data lakes and data visualization: An innovative approach to addressing the many challenges of health workforce planning," Online J. Public Health Informat., vol. 7, no. 3, p. 7, Dec. 2015.doi: 10.5210/ojphi.v7i3.6047.