RSECM: Robust Search Engine using Context-based Mining for Educational Big Data

D. Pratiba
Dept. of CSE, RVCE
Bangalore, Karnataka

Dr. G. Shobha
Dept. of CSE, RVCE
Bangalore, Karnataka

***Abstract*—With an accelerating growth in the educational sector along with the aid of ICT and cloud-based services, there is a consistent rise of educational big data, where storage and processing become the prime matter of challenge. Although many recent attempts have used open source framework e.g. Hadoop for storage, still there are reported issues in sufficient security management and data analyzing problems. Hence, there is less applicability of mining techniques for upcoming search engine due to unstructured educational data. The proposed system introduces a technique called as RSECM i.e. Robust Search Engine using Context-based Modeling that presents a novel archival and search engine. RSECM generates its own massive stream of educational big data and performs the efficient search of data. Outcome exhibits RSECM outperforms SQL based approaches concerning faster retrieval of the dynamic user-defined query.**

Keywords—Big Data; Context; Cloud; Educational Data; Hadoop; Search Engine

I. Introduction

There is a revolutionary change in the educational system in the modern times, where ICT plays a crucial role right from primary to advance technical education [1]. At present, it is not possible for an educational institution to provide extra knowledge and skills required for getting through any competitive examination or cracking the interviews for top notch Multinational organization. In order to cater up to the educational need, various enterprises have come forward to furnish educational knowledge and skill. This system is also called as online learning or e-learning system [2]. The conventional online learning system is not interactive, and it is one way communication dominantly, where students have to listen to the instructions presented in the form of the webinar. However, with the changing requirement of education, the needs of the students are also changing that demands the Learning Management System to be highly interactive [3]. Students requires the Learning Management System to be more real-time and more interactive to feel them a virtual presence:-

e.g. i) omnidirectional communication system among students-instructors, instructor-instructor, and student-student,

ii) availability of more offline assessments,

iii) exchange of course materials,

iv) sharing of students or instructor-centric study materials,

v) application to support a wide variety of plugins and various add-on features,

vi) access policy to be controlled by user owing to the collaborative educational network. Another trend observed in the present era is the migration of the majority of the enterprise applications over the cloud in order to give pervasiveness to the data and services. Our prior studies [4][5][6] has showcased an existing system towards digital learning along with a novel framework for providing security to big educational data. It also considers the design of interactive digital learning framework. However, the existing studies were quite analytical and need much more focus on query handling, complexities handling, and ensuring faster response rate for the stream of educational big data. Our prior framework [5][6] have discussed a system that can generate the big data efficiently. However, there is a presence of tradeoff found in our prior work introduced most recently in the research community and the actual need. When we conducted a thorough review, we found that there exists various industrial standards and tools for analyzing big data e.g. Hadoop, Hive, Pig, Cassandra, etc. We potentially felt that our existing framework have the better scope of enhancement and could bring the most additional feature that can potentially enhance the teaching and learning experience. It was also explored that till now there exist a massive archival of search engines on conventional data but never on big educational data. From any existing archival or digital repository, the outcome of the search for some specific content is either time consuming or gives irrelevant outcomes. The prime factor behind this is that existing search engines are not capable enough to identify the exact data that is valuable to the user [7]. Apart from this problem, the existing data mining algorithms are also not applicable for analyzing big educational data owing to the problem of high dimensionality and problems in the extraction of appropriate feature vector [8][9][10].

This paper reviews the existing system specifically about the mining techniques over the cloud and the techniques that are based on context mining. A significant problem statement is derived and then it proposes a novel technique that performs efficient, secure, and faster mining operation over educational big data using context-based approach. The technique introduced is quite simple and cost effective. Section II discusses the existing review of literature followed by problem identification in Section III. Discussion on proposed model is carried out in Section IV followed by research methodology discussion in Section V. Section VI illustrates the implementation techniques focusing on the algorithm and their respective operations. Outcomes being accomplished from the study are discussed on Section VII while the conclusion of the paper is made in Section VIII.

II. Related Work

This section provides the glimpse of various significant studies that are being carried out by different researcher and the different mechanisms used by them to evaluate the performance of their work. Since the evolution of dataset for an educational sector is still in infancy stage; therefore different disciplines of application are considered for studying the mechanisms of approach with a common factor called Big Data. There has been remarkable work and literature available in the field of medical analytics on big data, one of such work by Belle et al. [11], where medical image analysis, physiological signal processing, and genomics data processing is discussed. The author recommended building a reconstructed network with close co-operation among clinicians, computational scientist, and experimentalists.

This can be mimicked in the case of digital learning framework to have a collaborative learning – teaching intelligent mechanism. Such collaborative platforms of learning require an efficient mechanism of context retrievals, which needs robust aggregation techniques, one of such work in the context of image data is carried out by Cao et al. [12] for web-scale image retrieval, which uses distributed learning for ranking. It supports for billions of images. Data mining is an integral part in BigData where less informative words are ignored, and only significant words are considered. One such work is carried by Chen et al. [13] by reviewing data mining in IoT (Internet-of-Things). The authors have used data mining to enhance the performance as well as to save storage. The storage of ever growing data for the analytical purpose requires a large storage which is achieved using the cloud. One such work towards using the cloud as storage of analytical data was carried out by Khan et al. [14] about the smart cities. The prototype work is developed using Hadoop and Spark. The proposed work also makes use of cloud for storage using HBase in Hadoop framework. An increase in data is also associated with tedious work of managing, analyzing and integrating the data using big data. Reviewing the challenges and future perspective in managing, analyzing and integrating big data in the field of Medical Bioinformatics is carried out by Merelli et al. [15]. The system uses data management, performs data analysis along with the integration of data from various actors like students and faculties. Usage of deep learning can contribute the effective implementation of any work. One such work is carried out by Najafabadi et al. [16] who have implemented the concept of deep learning of big data analytics. The efficiency of any system depends on the effectiveness of the algorithms or techniques used in the system. One such kind of work is carried out by Oyana [17] where disease identification and analytics of visual data was achieved using Fangled FES-k means clustering algorithm. Different algorithms are used for various functionalities like multiple logins or in the case of search operations. In any educational system, the credibility of the system depends on the quality of the information posted by the system as well as its source. Refinement of Quality Information was carried out by Ramachandramurthy et al. [18]. This is achieved by using fuzzy Bayesian process which helps in improving the truthfulness in big data.

Effectiveness and efficiency of the system are dependent on the nature of its design modeling, and performance enhancement is based on optimization. One such work is carried out by Slavakis et al. [19] in signal processing domain, wherein the authors have presented encompassing models which capture huge range of signal processing relevant data for analytics (includes Principal Component Analysis (PCA), Dictionary Learning (DL) and Compressive Sampling (CS)). In an educational system, the queries submitted by the user or clients need to be appropriately addressed. This addressing requires individual search mechanism to provide a suitable solution. One of such work is carried by Cataldi et al. [20] wherein the author's present context-based search and navigation system based on the KBC (keyword-by-concepts) graph. Santini and Dumitrescu [21] have presented a search system which provides the search result considering the context related to certain activities. Fisher and Hanrahan [22] presented a mechanism for the context-based search of the 3D scene. They also proved that context-based performs better than the keyword-based search. Lane et al. [23] presented a local search technology which in comparison to other context-based searches also uses various factors such as location, time, user activity as well as weather. It also constructs a behavioral model and provides the result of the basis of personal requirement making using of similarity among users rather than the conventional result. Adomavicius [24] has emphasized on the significance of relevant contextual data in a recommendation system. Authors have introduced different framework such as contextual pre-filtering, post filtering, as well as modeling to facilitate the incorporation of contextual data in recommendation system. Li et al. [25] have formulated context-based people search using a grouping-based mechanism in a labeled social networking. Silva et al. [26] have introduced a context-based system which allows the evaluation of related data that is associated with a concept. It performed by certain distinct contextual information considering certain facts like user domain as well task, perspective and task intended to use the data. Thangaraj and Gayathri [27] have proposed searching techniques which consider not only context but also synonyms in comparison to the conventional keyword search engines. Vasnik et al. [28] presented a semantic and context-based search engine for the Hindi language. It is developed by lexical variance, user context as well as a combination of these two mechanisms. Gupta [29] has proposed a focused searched based on contextual data where the search engine is devised to such that it can decide the relevance of the data by certain context of user query keyword.

Depending on the quality as well as desired relevance, the result set size is minimized by eliminating irrelevant documents. Rahman et al. [30] have demonstrated a context-aware meta-search engine based on Eclipse IDE. Authors have taken advantage of the API's provided by different search engines like Bing, Google, and Yahoo as well as Stack Overflow site to provide the basic solution to errors and exceptions. The search engine works by content relevance; popularity as well context relevance. In the system, a search operation is carried in two ways in a conventional way as well as advanced way whereas the conventional way uses conventional keyword search whereas advanced search is performed by context.

Hence, it can be seen that there are various studies towards developing search engines using typical mechanism. Each mechanism has its advantages as well as pitfalls. The next section discusses the problems identified after reviewing the existing literature.

III. Problem Identification

After going through the contribution of existing literature from prior section, it was seen that conventional relational database management system (RDBMS) is adopted for reporting as well as archiving the data. However, Hadoop is deployed to deposit a massive quantity of data over a distributed cluster nodes and at the same it processes it as well. Performing data mining or analysis over the structured data can be easily done over RDBMS as it structures the enterprise data in the patterns of rows and columns. One interesting finding is that there is a myth that conventional data mining approach cannot be applied as the data is unstructured. However, the statement is not completely true. It is because, whenever there is a need for analyzing a part of the data from the big data, RDBMS may be a perfect choice. However, there are various challenges posed by the big data to be working on conventional RDBMS system. The inherent characteristics of big data (e.g. high dimensionality, heterogeneity, data veracity, data velocity, etc.) make the RDBMS system out of the picture when it comes to deploying mining approaches. Adoption of Hadoop mitigates this problem as it has various characteristics that support processing big data. With HDFS (Hadoop Distributed File System), it supports processing big data with complete optimization of the unused space too. HDFS is the frequently used approach even for storing and processing (mining) big data. However, there are certain issues in it. The problem identifications of the proposed study are as follows:

- The SQL-based database system cannot be used to store and process educational big data owing to its incapability of handling unstructured data. Existing data mining algorithms are not applicable directly over the massive streams of big data owing to the unstructured pattern of data.
- The security features of Hadoop are highly challenging and potentially complex to be configured. It doesn't have any form of standard encryption mechanism over its storage as well as network levels giving rise to authentication-based attacks. Hadoop is designed in Java, and same Java is also used by cyber criminals to launch an attack.
- The frequently used HDFS is not good enough for handling smaller files and incapable of performing an effective data compression. The design principle of HDFS is even capable of handling random reads of smaller files.
- The biggest problem is that although Map Reduce, another frequently used software framework, supports batch-based architecture, which will eventually mean that it will never permit itself to be deployed as the use case that will require real-time access to data.

Owing to the problems as mentioned above, existing data mining algorithms cannot be integrated over Hadoop and will thereby pose a challenge to generate a precise search of specific data in the cluster node. Therefore, after briefing the problems above, the problem statement of the proposed study can be stated as – "*It is the quite computational challenging task to evolve up with a mechanism that supports highly relevant search engine over complex and unstructured educational big data."* The prime goal of the proposed system is thereby to overcome this problem.

IV. Proposed System

The prime aim of the proposed system is to develop a technique that can perform both archival and search of the relevant data from the massive stream of educational big data. The technique is termed as RSECM i.e. Robust Search Engine using Context-based Mining for educational big data. The contributions of the proposed system are as follows:

- To develop an efficient archival process that can store the educational big data.
- To develop a precise search engine that can perform faster retrieval of the outcome with better accuracy.
- To show that proposed system is the better approach than conventional SQL-based approach.

In order to achieve 1st research objective of RSECM, the proposed system first develops three different actors associated with the educational domain and develops a prototype application for generating real-time educational big data. The big data is designed in the highly unstructured way as it consists of multiple forms of data with higher dimensionality. The next part of the RSECM technique is to provide enough security features to address the problems of security from Section III. Supported by a recent version of the cryptographic hash function, SHA3, the proposed system ensures higher secrecy in data transaction as well as system security. The next part of the functional operation is to perform conversion of unstructured data to structured data using Hbase. All the problems of frequently used HDFS highlighted in Section III can be addressed potentially using Hbase. The next functional operation of RSECM is to perform a search from the education data over the cloud. A novel search engine is designed using context based mining approach that initially preprocesses and then it applies post processing to generate the feature vector. The feature vector is stored in the cluster node. The proposed system will also introduce a simple query system using simple search mechanism and contextual search mechanism. The architecture of RSECM is highlighted in Fig.1

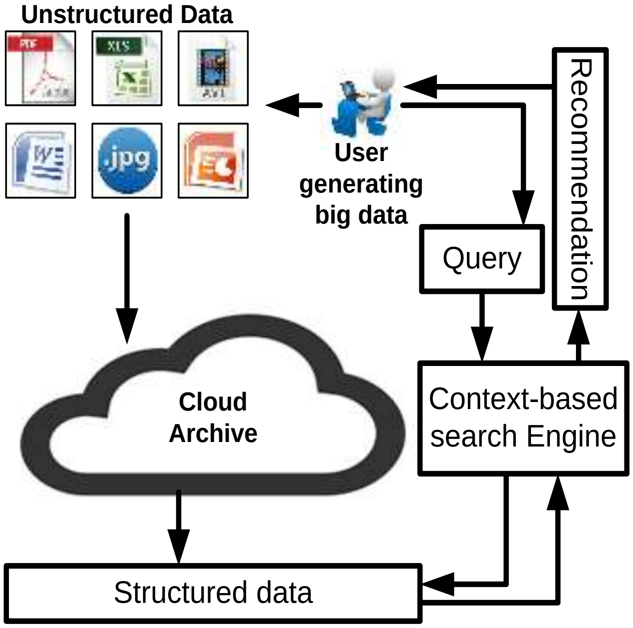


Fig. 1. RSECM Architecture of RSECM

V. Research Methodology

The proposed research work has adopted the analytical as well as experimental research methodology. Analytical modeling was developed for designing the context-based search, while experimental approach was considered to validate the outcomes on machines mapping with the cloud infrastructure. RSECM has the capability to generate real-time educational data rather than using existing available database. This section discusses the various significant task handled by the proposed RSECM for educational big data.

a) Actor Modelling

The proposed RSECM possess multiple forms of an actor of the cloud-based educational domain as well as learning management system to leverage teaching and learning experiences. RSECM has been developed over Java using a common interface protected by a novel authentication system. The interface supports accessibility to three types of actors i.e. i) students, ii) instructors and iii) policy makers. Fig.2 shows the system architecture associated with the actor modeling of RSECM. It also shows three actors associated with the interactive educational system. Initial state consists of profiling the student that records all the necessary information e.g. student's name, contact information, educational history, professional experience, etc.

A conventional database system can be used to store all profiling information. Upon successful profiling operation, the student is provided with access id and a static password originated from the cloud server. RSECM assumes that the static password is secured by the security patches practiced by the enterprise. As the existing security patches over cloud runs over the internet, we also assume that it is not safe and it requires to be protected too.

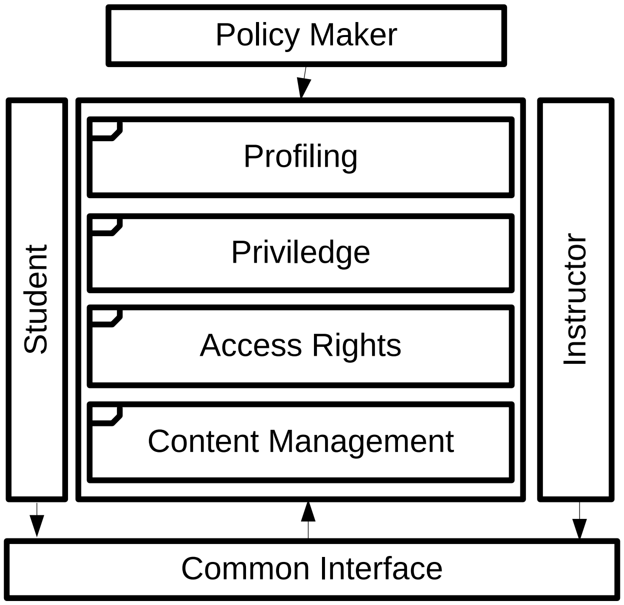


Fig. 2. System Architecture of RSECM Actor Modelling

The protection mechanism of RSECM is given by a novel authentication policy (discussed in next sub-Section B). RSECM possessed the feature of single as well as multiple authentication mechanisms, where the verification of the legitimacy of the user is done over cloud environment. The static password is only used for single authentication operation while in multiple authentication system prompts for multiple interfaces to validate the user multiple times over the internet. After successful authentication, the student actor is provided with feature to opt for an online educational system. The features about student actor are as follows:

- Can make the selection of the available list of course materials.
- Can visualize the skills and professional experience of their online instructors.
- Can download the study materials for the specific course enrolled.
- Can upload the study materials in the form of office files, PDF files, image files, audio files, video files.
- Can upload a particular file of new extension permitted by policy makers.
- Can take up the online test to assess their skill set on particular course.
- Can check their ratings provided by the instructors as an outcome of the test.
- Can communicate and share heterogeneous data with other members (students) as an open discussion forum.
- Can able to perform contextual-based search mechanism on appropriate educational data.

The second actor of RSECM modeling is an instructor, who is responsible for knowledge delivery services to the students using the remote connectivity applications in ICT. Before starting using their individual feature, the instructor will need to authenticate themselves on the same uniform interface (even used for authentication by the student actor). Similar profiling features will also be used to store the unique identity-based information of the instructor. The features pertaining to instructor actor are as follows:

- Can create a course on a particular subject and stream.
- Can upload various forms of study materials of multiple file formats permitted by policy makers.
- Can communicate with students by sharing and exchanging various forms of study contents.

The third actor of RSECM modeling is policy-maker, who is responsible for supervising the complete operations by instructors and students. RSECM emphasizes on the policy-maker as it is the only actor that monitors the activities as well as the privilege of student and instructors. The system also provides various forms of features in order to assist proper operations of the users e.g. it is policy-maker who can configure about the allowances of upload and download of the study contents. It can also configure the type of file extension (.doc, .xls, .ppt, .pdf, .avi, .jpg. WMV, .mp3 etc). As the policy-maker acts almost like the administrator of RSECM, hence it is skipped from the profiling process. However, they will be required to get themselves authenticated in order to execute their privileges.

Hence, the existing database consists of the high dimensional data which is massively growing and also is highly unstructured data. Before attempting to create a search, it is essential that such data should be stored efficiently as existing data mining approaches cannot be applied. RSECM uses Hbase data management, which is elaborated in next sub-section.

b) Security Management

The proposed system of RSECM uses the SHA3 algorithm in order to secure the authentication mechanism. Our work is the first attempt to implement the most recent cryptographic standards introduced in the year 2014. The schematic architecture of the security modeling of RSECM is discussed in Fig.3. The static input password of the user is considered as seed, where the SHA3 algorithm is applied. The system then generates two arbitrary functions for generating pseudonym A and B. The SHA3 is subjected to the pseudonym some iteration for the seed i.e. SHA3 (seed)A in order to generate 512 bits of fingerprint that is again subjected to the MD5 algorithm. This operation further generates 128 bit of print that is again split into two sub-keys of equal size in order to re-encrypt with AES. The final encrypted key is hidden in QR barcode, while one of the sub-keys generated from the 128-bit fingerprint is forwarded to user's email using email APIs in Java.

The proposed framework is now able to perform secure authentication of the user (student and instructors); however, as the proposed study relates the user to student actors mainly. The next process relates to an effective data management using H-base.

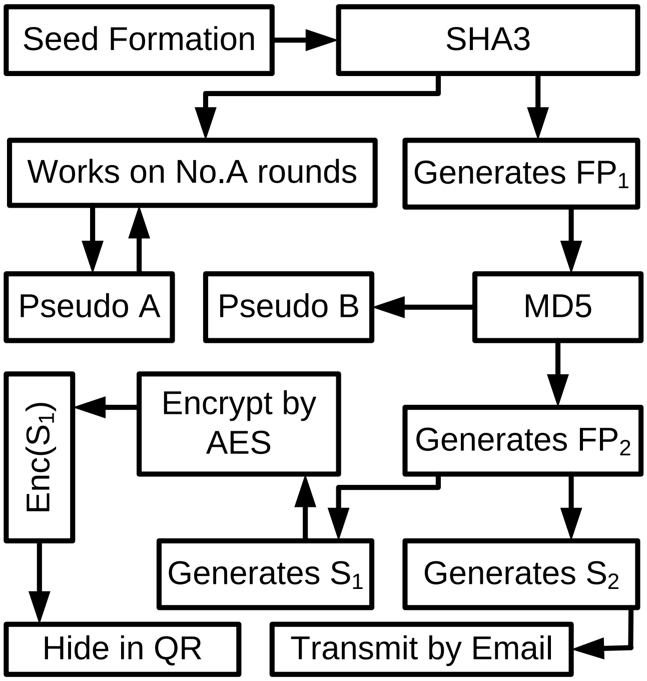
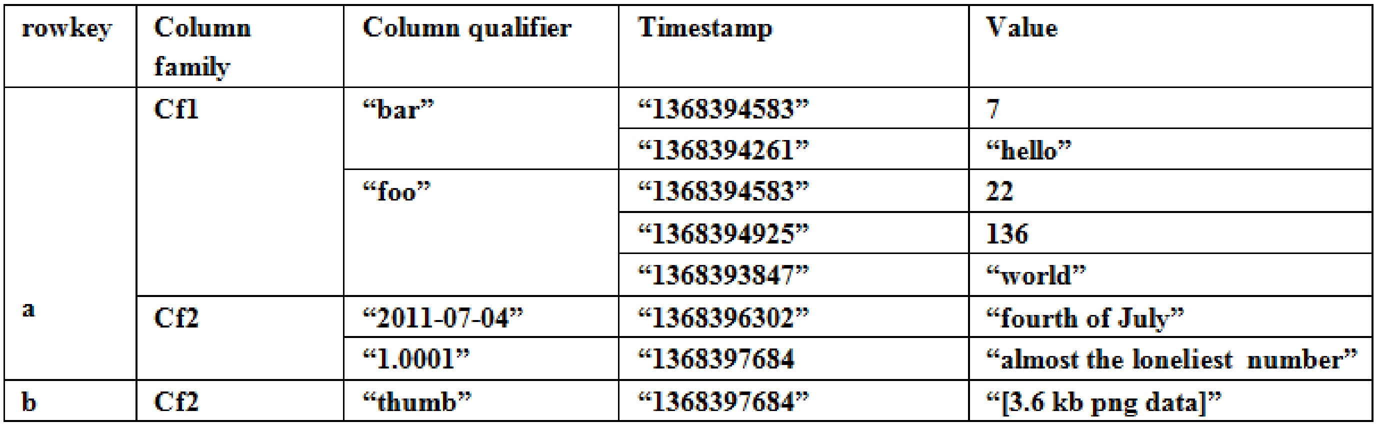


Fig. 3. RSECM Security Management

c) H-base Data Management

The proposed system of RSECM essentially generates a massive stream of random educational data which is highly unstructured and is not possible to store in the conventional database management system. However, at present, Hadoop is said to be made up of i) file system i.e. HDFS and ii) computational framework i.e. Map Reduce. Although HDFS permits repositing massive amount of extensive data in highly parallel as well as distributed manner, it doesn't support arbitrary read and write features being a file system. Although such feature is quite suitable for HDFS to access sequential data, in reality, it is not necessary that massive streams of data should be sequential in order. That's why the design principle of RSECM doesn't use HDFS but uses Hbase as it strongly supports random data with read and writes features on it. In a nutshell, Hbase has strong support to process and store unstructured data. Adoption of Hbase has another advantage of faster data access. Owing to storage characteristics of data in columnar pattern considering data as keys, Hbase can process and store the data quite faster than HDFS. Another interesting point about HBase usage is its potential for replica management with the aid of the feature termed as "region replication." According to this framework, it is possible to store and access multiple replicas over the region servers. An added advantage of replica management using HBase is its flexibility where it can manage multiple replicas in one region. The system uses a distinct replication identity to complete available replica initializing from 0. Hence, we call the primary region for any region whose replication identity corresponds to 0. Similarly, there are secondary replication regions too. All the significant writers are possessed by primary replication region along with all the updated alternations on the data. Therefore, with this feature, HBase provides a better data consistency.

Although, there is certain search engine like ElasticSearch [31] that can perform the search over cloud using graphical exploration tool, the proposed system chooses to stick on to the usage of HBase owing to its better editable properties of HDFS architecture. The data modeling in RSECM is designed in order to facilitate the unstructured data that significantly differs in size of the field, data type, and columns.



rowkey	Column family	Column qualifier	Timestamp	Value
a	Cf1	"bar"	"1368394583"	7
			"1368394261"	"hello"
		"foo"	"1368394583"	22
			"1368394925"	136
			"1368393847"	"world"
	Cf2	"2011-07-04"	"1368396302"	"fourth of July"
		"1.0001"	"1368397684	"almost the loneliest number"
b	Cf2	"thumb"	"1368397684"	"[3.6 kb png data]"

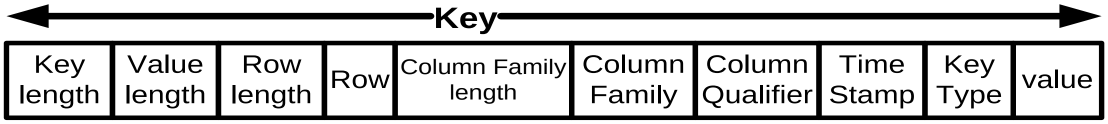


Fig. 4. Components of Hbase used in RSECM

The data modeling supported by RSECM possess multiple components are as shown in Fig.4. The functions of each component are briefly discussed below:

- *Tables*: All the unstructured data is organized in tables which are coined by string-based name and is consist of multiple collections of rows reposited in discrete partitions that are called as regions.
- *Rows*: It is a position where the data is reposited and is identified using the respective key. There is no data type for a row keys. It is also called as the byte array.
- *Column families*: It is group under which row each data resides. It is a string and is made of characters that are safe for deployment.
- *Column Qualifier*: It is a type of flag that is used for addressing the column family. It doesn't possess any form of data type and is often considered as a byte array.
- *Cell*: A cell is a combination column family, a key of a row, and column qualifier. The value of the cell is used to flag the data stored in it. It doesn't possess any form of data type and is often considered as a byte array.
- *Version*: It is a number that is used for signifying the time of generation of the cell. The default number of version of cell in HBase is 3.

Therefore, the storage framework of the RSECM will consist of the database with columnar pattern and table with rows. There is a primary key for each row also called as row keys. It is also possible to have multiple columns in each row with timestamp data stored in each cell. The method to access the data is only through the row key, table, family, timestamp, and column. The system has also used the simple APIs e.g. get, put, scan, and delete. This the way how unstructured data is converted to structured data and stored in Hbase storage. We choose to use Hbase as it supports transactional data exchanging mechanism in Hadoop that is essentially required by the user to get the updates and do personalization of their educational data.

d) Mining with Contextual Search Engine

The proposed system carries out a novel design of data analytics by carrying out mining operation as well as contextual-based search operation. The proposed system offers the generation of a query through HBase that already uses Hadoop metrics framework. The system then inherits the features as well as its classes. The mining operation of the proposed system is carried out by a simple statistical method. The presented method aggregates the statistics of a varied relationship among the words from the educational data in order to provide query specification and comparison of the objects. A contextual database is built by initially compiling the relationship between the words in order to encapsulate the statistics of similarity measures among the words. During the data retrieval, we compute a search matrix of keywords given.

The schematic diagram for this is shown in Fig.5.

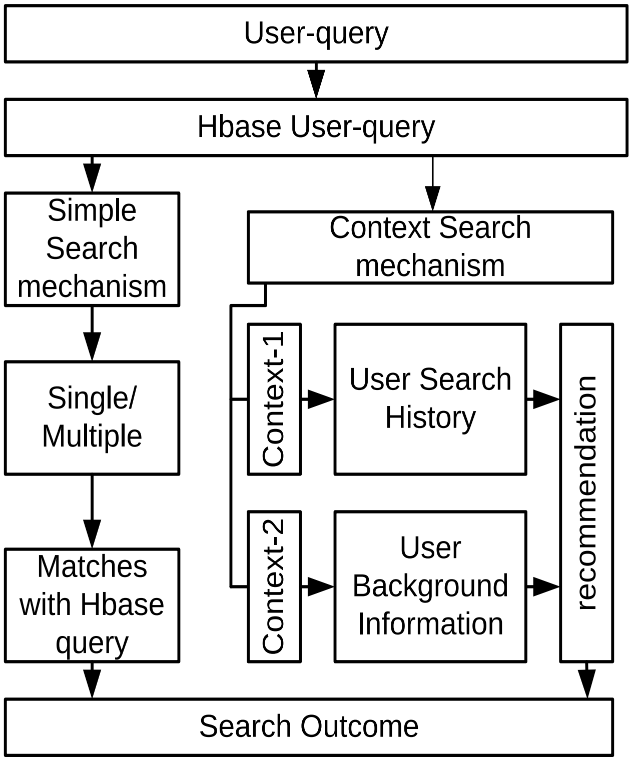


Fig. 5. Mining with Contextual Search Engine

Fig.5 shows that the transactional education big data is initially an unstructured data for which reason; it is not feasible for carrying out data mining technique. The problem is mitigated after directing the stream of unstructured data to the Hbase that performs self-indexing of the data. The unstructured data is first processed to become compatible with the Hbase index. This operation is carried out using data mining technique that performs both preprocessing as well as post processing. The preprocessing operation includes identification and removal of noise in the data, redundant data, etc., while post-processing operation involves feature extraction operation. Finally, a significant feature is extracted and stored in cluster node (or Hbase). The next process is about the search technique implementation that is initiated using user-based query keywords. The search technique can be both single as well as multiple searches. The second step is to convert the user query to the Hbase query, where depending on the number of the words, it generates multiple HBase queries. As per the number of the words in a user-based query, the loop will work for generating some Hbase query. Finally, the Hbase query returns results in sequential form for each HBase query. Finally, it matches queries from the source of files and matching sentences. The entire process is called as simple search design that is functional when the users access the common interface of RSECM. The proposed system develops context-based search in two ways,

- *Historical Context*: The proposed system considers the search history as the prime factor of developing context. If the user-centric search history, as well as user query, are found similar than the result of the Hbase query will act as recommended result of the context-based search.
- *Background Context*: The background of the user is also included for forming the context-based search. If the information pertaining to the background of the user (from profiling data) as well as the user-centric query is found to be similar than the result generated by the Hbase query will be considered as recommended results of the context-based search.

VI. Implementation

As the proposed system considers the experimental approach, so it is essential to have a standard and perfect prototyping of the experimental test-bed. RSECM experiments over Oracle VirtualBox in Linux machine that provides a standard virtualization with a supportability of guest operating platform. For this purpose, a virtual machine is created and configured with the necessary parameters as well as settings for creating a cluster node (or Hbase). Cloning of this reference virtual machine is carried out multiple times for an experimental purpose. The inclusion schema used of VirtualBox integration is shown in Fig.6.

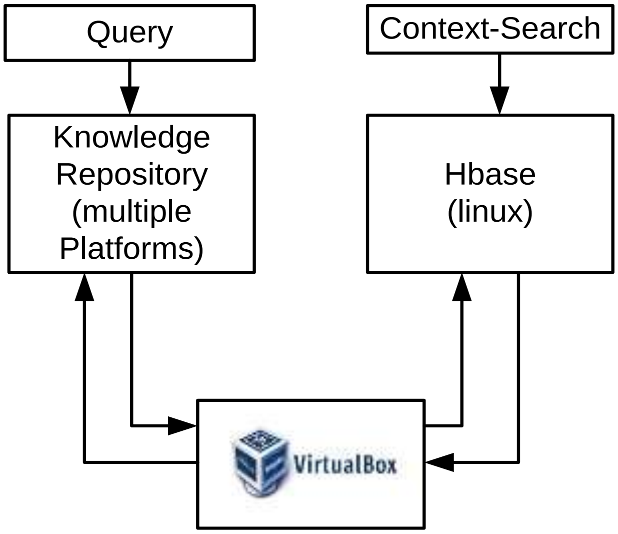


Fig. 6. Inclusion Schema of VirtualBox

The above Fig.6 shows the use of multiple forms of operating system, where the knowledge base is created. It means that user application interface lies on the 1st block where the operating system and machine configuration may quite differ from each other. It is the block that is responsible for the generation of big educational data, which is characterized by unstructured and massive streams. The user-based search is also generated from this block. The second block consists of the virtual environment where VirtualBox is configured to create Hbase in Linux machine. The contextual-based search is carried out in this block. Both the blocks are connected via cloud environment and query processing in order to accomplish better data synchronization. The implementation of the RSECM is carried out in 32/64 bit operating system consisting of 8GB RAM, and a storage capacity of 1TB. The interface designing, as well as HBase work, is carried on Java platform using JDK 1.7, using NetBeans IDE and Oracle Virtual Box. A laboratory prototype is a setup performs this implementation using the cloud. The proposed system is supported by essentially four types of algorithms actor modeling, security management, Hbase data management, and mining with the contextual search engine.

TABLE I. ALGORITHM FOR GENERATING UNSTRUCTURED DATA

```
**Algorithm-1: Generation of Unstructured Data**
**Input**: n (number of actors), p (number of characteristics)
**Output**: unstructured data generation
**Start**
1. Define type of actors [$i_n$ | $n = 3$]
2. Define characteristics of actors [$i_n=C_p$]
3. $i_n \rightarrow files(m)\ \forall\ m \in 14$
4. Stream data to Hbase
**End**
```

Table 1 highlights the algorithm that has used in the generation of the unstructured data. The algorithm takes the input of *n* number of actors where *n* is considered to be three i.e. i) student, ii) instructor, and iii) policy maker. It is also feasible for enhancing some actors depending on the type of application. Each actor is defined a particular set of characteristics p to represents their privilege. Any event of p will lead to generation of m files (.doc, .docx, .xls, xlsx, .ppt, .pptx, .pdf, .jpg, .bmp, .tif, .png, .avi, .mpeg, wmv). The higher limit of m is controlled by policy-maker. Line-3 of the algorithm-1 represents the generation of transactional data by the users leading to a generation of highly unstructured data that is now streamed to cluster node (or Hbase). The security features responsible for maintaining authentication for the actors are as discussed next.

TABLE II. ALGORITHM FOR SECURITY MANAGEMENT

```
**Algorithm-2: Security Management**
**Input**: pswd (password)
**Output**: Key generation and authentication
**Start**
1. str (pswd) $\rightarrow$ S
2. $g(x)=( Z_1, Z_2)$
3. Apply hash $\rightarrow$ SHA3(S)$Z_1 \rightarrow FP_1$
4. Apply hash $\rightarrow$ MD5($FP_1$)$Z_2 \rightarrow FP_2$
5. Div($FP_2$) $\rightarrow$ [$S_1$(64 bit), $S_2$(64 bit)]
6. Enc($S_1$) $\rightarrow S_{e1}$
7. Embed $S_{e1}$ in QR barcode
8. Forward $S_2$ to user-email
9. Extract $S_{e1}$ and $S_2$
10. Authenticate using $S_2$
6. Allow access
**End**
```

The working of the Algorithm-2 is illustrated as follows. Initially, the unique data i.e. confidential data such as password from the user profile is retrieved. This unique data is used to generate the secured key for authentication. The unique data undergoes two kinds of the hashing function. The first hashing function is denoted by Z_1 and is performed using the SHA3 algorithm. The next hashing function is denoted by the Z_2 and is carried on the unique data using the MD5 algorithm. The SHA3 encrypted output resulting Z_1 acts on the unique data for x iteration, whereas the MD5 encrypted data resulting Z_2 acts on unique data, and the result of SHA3 encrypted unique data represented by Z_1 for *y* iterations. These results in the generation of the desired secured key required for authentication of the login process.

After the generation of the secure key, the authentication process continues as follows. The secured key resulted from the multi-login Algorithm is 128 bit. The 128-bit key is then split into two parts by dividing it using a simple division. This results in a generation of two subsets of secured key denoted by S_1 and S_2 each of which is of 64-bit long. The first subset S_1 is sent to the user mail. These subset keys are further subjected to encryption using the AES algorithm, i.e. The Subset key act as data and plaintext for the encryption algorithm. Here the first subset key is used as data/plaintext and the second subset key acts as an encryption key. The first subset key is encrypted by using the second subset key using AES algorithm. The output of this operation results in a bit data which is converted into a string. This string is used to generate a QR Barcode. This QR Barcode consists of the encrypted string. In order to generate the authentication key, we require two sets of keys; the two keys are one which is present in the user mail (first subset S_1), and the other key is hidden in the QR Barcode. This QR Barcode is scanned using a user developed the application. The decryption operation is performed using the key. In simple words, the authentication needs the two key one in the user mail and the key in the barcode. This feature provides robust security since the chances of comprise of both the keys are negligible. As the secured authentication requires both keys, a user having single key will never be successful in log in to the system. This multi login algorithm is applicable for both faculty and student who have multi login option. On successful login, the user will be directed to a page containing details about courses which will facilitate various operations such as download information; syllabus and other queries like FAQ will also provide details related to faculty. The student with the privileges will be provided different options like, upload, download, search, thread, start a new thread so on.

- *Upload*: Here if the student is granted privilege from the admin staff, a student can upload his own file.
- *Search*: Allows the student to perform context based search.
- *Thread*: Provides platform for discussion for students. This can be used for knowledge sharing.
- *Start a new thread*: Allows the user to start a new discussion. It should be noted that all these threads are used in the background for context generation.

After the RSECM provides enhanced security capabilities using the most recent version of the cryptographic hash function, the next step is to perform data conversion using Hbase management. The algorithm used for big data conversion is shown in Algorithm-3.

TABLE III. EXTRACTING STRUCTURED DATA USING HBASE

```
**Algorithm-3: HBase Data Management**
**Input**: m (unstructured data)
**Output**: structured data
```

```
Start
Generate db (m)
Classify $j$ [db (m)]
Configure $p$
For $p$ =1…..$k$
    $q \subseteq 1, 2, …., 2k$
    $r \subseteq k$
Pass $j$ to reducer
Reducer $j$ forwards to Hbase ($j_{hbase}$)
Represent $j_{hbase} \rightarrow$ dis(row, col)
Stream $j_{hbase(row, col)}$.
End
```

Table 3 shows the process of conversion of the unstructured educational data *db(m)* to the structured one. Although from the Algorithm-1, it was discussed that we choose to experiment with 14 file types; however, for further challenges, we also consider multiple forms of files or types of educational data. The variable *j* represents such forms of education data which are normally log files, streams of events, and various forms of educational course materials. Although there can be many more categorizations, RSECM chooses only these forms of educational file and subject it to Hadoop. The variable *p* basically represents *HRegion*, and *q* represents *MemoryStorage* and *HFile* of Hbase. The variable *k* represents the number of blocks of *HRegions* in Hbase, while *r* represents *HLog*. The data *j* is then finally passed on to the reducer that forwards the data to Hbase and thereby converting the entire data to structured one.

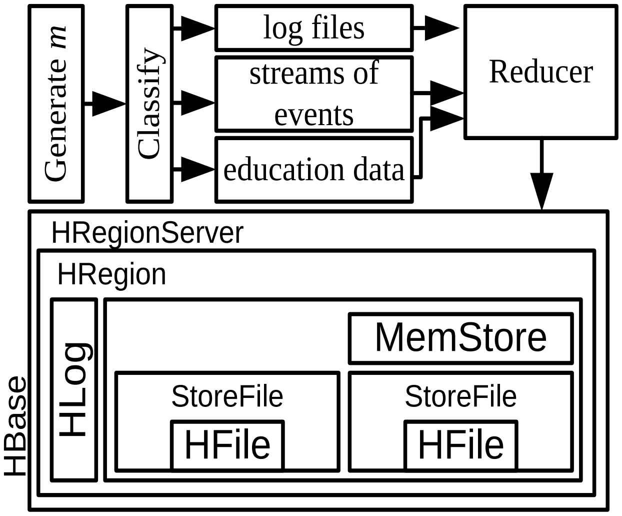


Fig. 7. HBase Architecture

HBase architecture essentially acts as the filter to eliminate the complexities associated with unstructured data (Fig.7). The context-based search in the secure educational interactive digital library provides various features to the instructor and student. The privileges are controlled by the Policy maker. The student and instructor are provided with single as well as multiple authentication schemes. The policy maker performs the action of enrolling of instructor and students. The instructor and students are entitled to different kind of privileges such as downloading, uploading or both as provided by the policy maker who acts as administrator. The instructor and student are required to satisfy certain requirement to access the uniform interface successfully. The policymaker will provide a single or multiple authentication schemes to the instructor and student. Each time a student or instructor is enrolled into the digital library, he/she is required to provide certain information's, for instance, such as his credentials, background interests, prior knowledge and so on. All these information is used to generate context and are saved in the form of simple text in the cloud, which will contain all other information such as credentials like passwords, blog history and so on. In order to have access to the digital library, instructor and students must provide certain credentials such as username/email and password for single login. In a case of multiple authentication schemes, the procedure is illustrated in the form of the algorithm. The algorithm for performing data mining using the contextual search engine is highlighted in Table 4.

TABLE IV. DATA MINING WITH CONTEXTUAL SEARCH ENGINE

Algorithm-4: Mining with Contextual Search Engine
Input: $j_{hbase(row, col)}$ (Structured data)
Output: Matched data

```
Start
$j_{hbase(row, col)} \rightarrow$ preprocess
Extract FV
Store FV in $j_{hbase(row, col)}$
For Simple Search
$query_{user} \rightarrow query_{hbase}$
For l=1:size($query_{user}$)
  Match $query_{hbase}$ with source file
  Match $query_{hbase}$ with matching contents
  Return $r_{hbase}$;
For Contextual Search
  If ($U_{SH}$, $query_{user}$) = = $r_{hbase}$
       $recom_{cont}$=$r_{hbase}$
  If ($U_{BI}$, $query_{user}$) = = $r_{hbase}$
       $recom_{cont}$=$r_{hbase}$
End
```

The output of Algorithm-3 i.e. structured data from Hbase is considered as input for Algorithm-4. A simple preprocessing is applied to the structured data and feature vector *FV* is extracted that is ultimately stored back to the same cluster node i.e. Hbase. The entire algorithm works in two methods e.g. i) simple search and ii) contextual search. In simple search method, the size of the user-defined query is considered to be the highest limit of *for* loop in order to perform an iteration of similarity match in the search mechanism of RSECM. The user-defined query is matched with source file as well as matching contents to finally return a result from Hbase i.e. r_{hbase}.

In contextual-based search, the system checks if the search history of the user U_{SH}, as well as their query $query_{user}$, is found equivalent to outcomes of Hbase i.e. r_{hbase} than the system considers r_{hbase} as recommended outcome against the user-defined the query. Similarly, if the background information UBI and user $_{query}$ are found equivalent to r_{hbase}, that the system considers r_{hbase} as recommended outcomes against the user search. Therefore, the search operation supports two kinds of search; one is the simple search whereas the other is contextual search. These two search operations are illustrated using algorithms. The search operation is preceded by the streaming of data, wherein the data is received, mined, and stored in the HBase in order to perform the search operation, data mining of the unstructured data which is collected as plaintext from the various context and stored in the cloud is used. The mining operation is as follows.

- Reads the unstructured data from the cloud
- Perform operations like preprocessing which include removal of stop word, URL, HTML content, removing spaces, noisy data, special characters and less informative or less significant words.
- Feature Extraction performs the tokenization or vectorization of string; extracts featured data, and stores these featured data into HBase.

A simple keyword-based search is initiated by the user. The system performs tokenization on this basis and searches for relevant data based on a keyword is searched. Here the user authentication is not needed. Be it simple or advance search mechanism, the system keeps a consistent check of the statistical information i.e. frequencies of the word and word proximity for all the input objects. Simple inferential statistics could be used for better inference mechanism in order to extract better knowledge after every query processing. Algorithm-4 uses the advantage of statistical computation carried out by Hbase in order to perform discrete query processing. In advanced search, the algorithm differs slightly from the earlier one used for the simple search. Here also the search begins with the keyword provided by the user, and the same principle of tokenization is also followed here. The search here differs from prior step in the sense, here the search performed is based on the context basis, i.e. the search here considers different parameters like the user's search history, context related to his background, context from his profile, and various context related to the user available within the cloud. The important thing to be noted here is the user authentication is needed here; the user needs to be in active session, so it has to access the user profile. One more difference between the simple search and the advanced search is the output of the advanced search is recommended whereas the simple search provides relevant data. The recommended data is obtained by checking the user profile history and other parameters based on the context. When this context matches with the keyword for search, these are generated as recommended search and provided to the user. When the search query is not matched with the result of user background or user profile history, the result returned is simple search result without recommendation. In case of multiple keywords each of the keyword is checked within the user profile history and when matched with the result are provided as recommended output. Compared to simple search, the advanced search gives more precise and more accurate data required by the user as it involves the searching of the user profile history, it also searches the most searched content or data by the user during the process of recommending advanced search result. The complete operation of the novel contextual-based search is highlighted in Fig.8. Hence, the proposed system uses quite a novel and simple process for extracting the features in order to provide a relevant search outcome for the users of this framework. The search engine is designed to provide both extensive search outcomes as well as narrowed and highly relevant search outcomes.

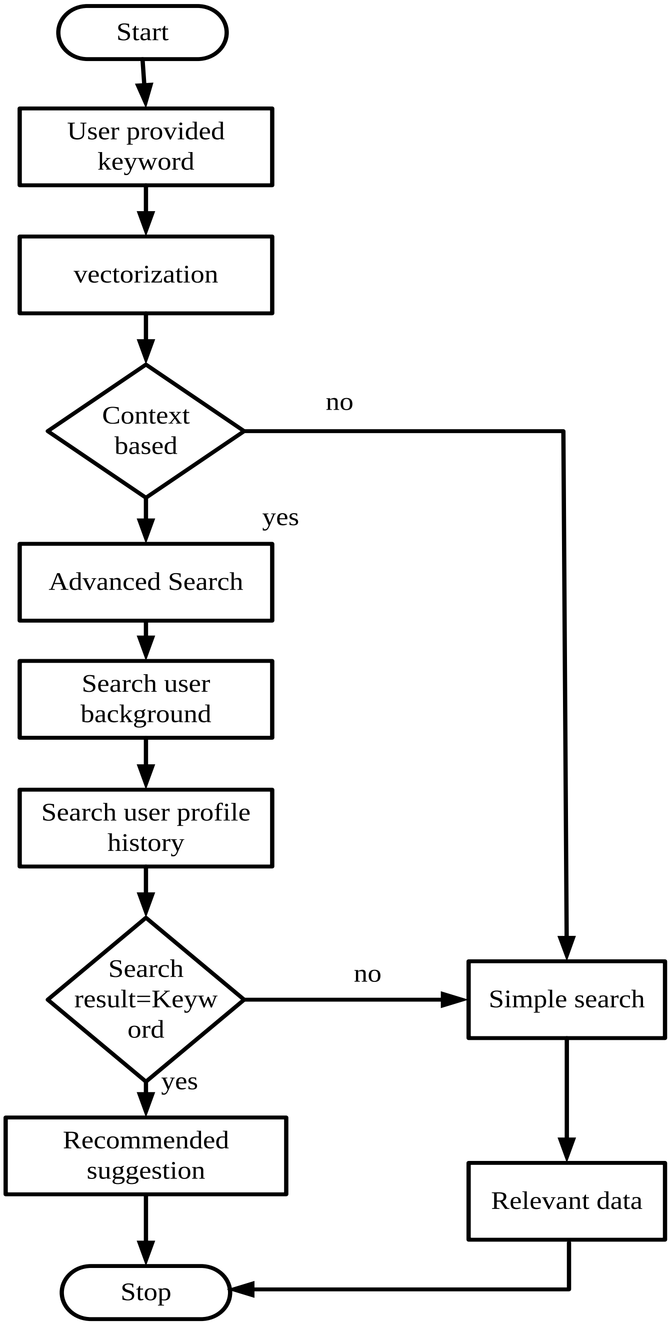


Fig. 8. Process flowchart of search operation

VII. Result and Discussion

This section discusses about the outcomes of the study that is compared with the conventional data mining technique using the SQL-based search engine. In order to talk about the results, it is necessary that we generate data on real-time basis. We have already developed an experimental prototype over Java to generate educational big data [6] along with security features incorporated in it [5]. Hence, the output of unstructured or semi-structured data is fed to Hbase and proposed algorithm to design and develop contextual-based search engine. The software testing is done using black-box testing, unit-testing, integration testing, and system testing. The formulations of text-cases are done on the basis of evaluating the core objectives of proposed system i.e. accuracy of search outcomes. Owing to the nature of novelty in the proposed study, we choose to craft novel performance parameters in order to gauge the effectiveness of the study. Following are the performance parameters:

- *Processing Time (PT)*: It is the total time consumed to process the entire algorithm processing right from query origination to recommendation generation from Hbase cluster node.
- *Actually Recognized Context (ARC)*: It is a measure of actual context being correctly matched with that of user-defined query.
- *Falsely Identified Context (FIC)*: It is a measure of outcome for context that doesn't completely match with the query of the user.
- *Missing Context (MC)*: It is a measure of an outcome that doesn't have any matching context for the user query.

The technique uses manual approach for evaluating the Total Context (TC) where the Context Identification Rate (CIR) can be computed as:

$$CIR=ARC / TC$$

TC can be initialized to a number corresponding to some search history as well as background information of the user. And the error in the context identification rate (ECIR) is evaluated as,

$$ECIR=FIC / [ARC+FIC]$$

The system also has computation of Non-Context identification rate (NCIR) as

$$NCIR=MC/ARC$$

The numerical outcome of the study is shown in Table 5. In order to make the analysis easier, we consider the equal number of TC. A closer look into the outcomes will show that proposed study offers significantly lesser rate of error in contextual search as witnessed by the values of ECIR and NCIR as compared to the SQL. The prime reason behind this is SQL uses relational structuring of the database while proposed RSECM is based on wide column data storage on Hbase. This results in maximum processing time for SQL which is never recommended for analyzing big educational data. Moreover, SQL based mining approach is based on tabular data stream, JDBC, and ODBC, while the RSECM is designed based on Java API with supportability of Hbase resulting in an efficient conversion of unstructured to structured database. Another interesting part is the inclusion of MapReduce that allows processing the distributed file system.

TABLE V. COMPARATIVE ANALYSIS OUTCOME

	TC	ARC	FIC	MC	CIR	ECIR	NCIR
RSECM	500	393	17	19	0.786	4.146341	4.834606
	500	349	14	11	0.698	3.856749	3.151862
	500	251	27	22	0.502	9.71223	8.76494
	500	458	18	25	0.916	3.781513	5.458515
	500	350	17	19	0.7	4.632153	5.428571
	500	485	27	21	0.97	5.273438	4.329897
SQL	500	175	87	79	0.35	33.20611	45.14286
	500	161	50	35	0.322	23.69668	21.73913
	500	251	94	94	0.502	27.24638	37.4502
	500	188	39	55	0.376	17.18062	29.25532
	500	167	55	70	0.334	24.77477	41.91617
	500	105	38	37	0.21	26.57343	35.2381

Map Reduce is highly essential to be used for data mining process as the data is actually stored in distributed nodes and Map Reduce can process the data in highly distributed manner on the cluster nodes. This capability is quite missing in the SQL based data analyzing techniques. Every node has the capability to process the data potentially rather than simply moving the data on the network. Apart from the conventional database (SQL or any other RDBMS) system, RSECM enables to carry out the querying of the data in real-time by building the data over columnar fashion using Hbase. Originally, this columnar pattern in Hbase acts like a hash table resulting in faster query processing as compared to conventional SQL. Fig.9 is the evidence of it.

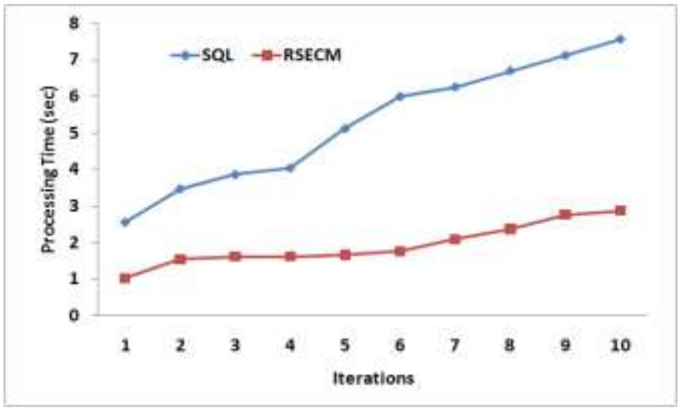


Fig. 9. Processing Time of RSECM and SQL

After reviewing the cumulative outcomes, it can be said that RSECM is highly suitable if there is a need to process massive educational data which lies in idle servers in educational institutions. The approach is also quite cost effective as RSECM doesn't require any particular modeling, however, SQL based approach of context mining is 100% dependent on data modeling. The system also offers an enormous scale of processing capabilities as well as storage facilities at a comparatively lower cost as compared to any SQL-based search engines. Another interesting point of optimistic outcome is RSECM also supports parallel processing and can even run various set of data in parallel and yield the outcomes as faster as possible.

VIII. CONCLUSION

From the study of existing literature as well as observing the industrial trends, it is found that Hadoop has been familiarized too much. One of the most challenging facts explored is that the existing technique based on Hadoop for storing and processing the big data is just beginning. The fact of familiarization is only because of the open source nature. There are multiple sources which say advantageous features of Hadoop as well as disadvantageous features of Hadoop not only for data storage but also for mining approaches. The phenomenon of inapplicability of conventional mining technique is another reason to boost various research attempts. This paper has focused on one of the rising arenas of education system where the entire data management of educational system will give rise to big data, which is not possible for any conventional RDBMS framework or data mining technique to normalize it and make it valuable for us. Hence, this paper has developed a technique with the aid of Oracle Virtual Box where the unstructured data is converted to structured data, and then a context based mining is used to extract the knowledge of the data. The proposed system has been tested on multiple platforms as well as multiple machines in order to cross-check its performance of processing the educational big data. The outcomes are quite optimistic and highly encouraging. We gave more emphases on computational speed as it is an only problem that many researchers are encountering to achieve in developing data mining techniques over cloud.

For the future research prospective considering scope of educational big data, the proposed framework can be used to improvise the students learning skills. This work gives the hints for future researches to improve further more computational speed along with the futuristic real time educational data.

REFERENCES

[1] H. Kanematsu, D. M. Barry, STEM and ICT Education in Intelligent Environments, Springer, 2015.

[2] G. Vincenti, A. Bucciero, C. V. Carvalho, E-Learning, E-Education, and Online Training, Springer International Publishing, 2014.

[3] L. Uden, J. Sinclair, Y-H Tao, D Liberona, Learning Technology for Education in Cloud - MOOC and Big Data, Springer, 2014.

[4] D. Pratiba, G. Shobha, “Educational BigData Mining Approach in Cloud: Reviewing the Trend”, International Journal of Computer Applications, Vol. 92, no.13, April 2014.

[5] D. Pratiba, G. Shobha, “S-DILS: Framework for Secured Digital Interaction Learning System Using keccak”, Springer-Software Engineering in Intelligent System, Advances in Intelligent Systems and Computing vol.349, pp.175-187, 2015.

[6] D. Pratiba, G. Shobha, “IDLS: Framework for Interactive Digital Learning System with Efficient Component Modelling”, Springer-Proceedings of Third International Conference on Emerging Research in Computing, Information, Communication and Applications, 2015.

[7] M. Levene, “An Introduction to Search Engines and Web Navigation, John Wiley & Sons”, 2011.

[8] I. Koch, “Analysis of Multivariate and High-Dimensional Data”, Cambridge University Press, 2013

[9] J. Leskovec, A. Rajaraman, J. D. Ullman, “Mining of Massive Datasets”, Cambridge University Press, 13-Nov-2014.

[10] S. E. Ahmed, “Perspectives on Big Data Analysis: Methodologies and Applications”, American Mathematical Society, 20-Aug-2014.

[11] A.Belle, R.Thiagarajan, S. M. Reza Soroushmehr, F.Navidi, D.A. Beard, and K. Najarian, “Review Article Big Data Analytics in Healthcare”, Hindawi Publishing Corporation BioMed Research International, pp. 16, 2015.

[12] G. Cao, I. Ahmad, H. Zhang, W. Xie, and M. Gabbouj, “Balance learning to rank in big data", Proceedings of the 22nd European in Signal Processing Conference (EUSIPCO), pp. 1422-1426, 2014.

[13] F. Chen, P. Deng, J.Wan, D. Zhang, A. V. Vasilakos, and X. Rong, “Review Article Data Mining for the Internet of Things: Literature Review and Challenges”, Hindawi Publishing Corporation International Journal of Distributed Sensor Networks, Article ID 431047, 2015.

[14] Z. Khan, A. Anjum, K. Soomro, and M.A. Tahir, "Towards cloud based big data analytics for smart future cities", Journal of Cloud Computing, Vol. 4, No. 1, pp.1-11, 2015.

[15] I. Merelli, H. P-Sánchez, S. Gesing, and D.D. Agostino, “Review Article Managing, Analysing, and Integrating Big Data in Medical Bioinformatics: Open Problems and Future Perspectives”, Hindawi Publishing Corporation Bio-Med Research International, 2014.

[16] M.M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic, "Deep learning applications and challenges in big data analytics", Journal of Big Data, Vol. 2, no. 1, pp.1-21, 2015.

[17] T.J. Oyana, “Research Article A New-Fangled FES-k -Means Clustering Algorithm for Disease Discovery and Visual Analytics”, Hindawi Publishing Corporation EURASIP Journal on Bioinformatics and Systems Biology, pp. 14, 2010.

[18] S.Ramachandramurthy, S.Subramaniam, and C. Ramasamy, “Research Article Distilling Big Data: Refining Quality Information in the Era of Yottabytes”, Hindawi Publishing Corporation Scientific World Journal, 2015.

[19] K. Slavakis, G.B. Giannakis, and G. Mateos, “Modeling and Optimization for Big Data Analytics”, IEEE Signal Processing Magazine, 2014.

[20] M. Cataldi, C. Schifanella, K. S. Candan, M. L. Sapino, and L. D. Caro, "Cosena: a context-based search and navigation system", Proceedings of the International Conference on Management of Emergent Digital Eco Systems, pp. 33, 2009.

[21] S. Santini, and A. Dumitrescu, "Context based semantic data retrieval", Proceedings of JSWEB, 9th Sep, 2015.

[22] M. Fisher, and P. Hanrahan, "Context-based search for 3D models", ACM Transactions on Graphics (TOG), Vol. 29, No. 6, pp. 182, 2010.

[23] N.D. Lane, D. Lymberopoulos, F. Zhao, and A. T. Campbell, "Hapori: context-based local search for mobile phones using community behavioral modeling and similarity", Proceedings of the 12th ACM international conference on Ubiquitous computing, pp. 109-118, 2010.

[24] G. Adomavicius, and A. Tuzhilin, "Context-aware recommender systems", Recommender systems handbook, Springer US, pp. 217-253, 2011.

[25] C-T. Li, M-K. Shan, and S-D. Lin, "Context-based people search in labeled social networks", Proceedings of the 20th ACM international conference on Information and knowledge management, pp. 1607-1612. 2011.

[26] C.F.D. Silva, P.Hoffmann and P.Ghodous, “Improve Business Interoperability through Context-based Ontology Reconciliation”, International Journal of Electronic Business Management, Vol. 9, no. 4, pp. 281-295, 2011.

[27] M. Thangaraj, and V. Gayatri, "A new context oriented synonym based searching technique for digital collection", International Journal of Machine Learning and Computing, Vol. 1, no. 1, 2011.

[28] N. Vasnik, S. Sahu, D. Roy, “TALASH: A Semantic and Context based Optimized Hindi Search Engine”, International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol.2, no.3, 2012.

[29] P. Gupta, "A novel approach for context based focused search engine", PhD diss., JAYPEE Institute of Information Technology, 2013.

[30] M.M. Rahman, S. Yeasmin, and C.K. Roy, "Towards a context-aware IDE-based meta search engine for recommendation about programming errors and exceptions", IEEE Conference of Software Maintenance, Reengineering and Reverse Engineering (CSMR-WCRE), Software Evolution Week-, pp. 194-203, 2014.

[31] Gormley, Clinton, and Zachary Tong. “Elasticsearch: The Definitive Guide", O’Reilly Media, Inc.", 2015.