Enhanced Data Lake Clustering Design based on K-means Algorithm

Jabrane Kachaoui[1], Abdessamad Belangour[2]
Hassan II University, Faculty of Science Ben M'sik
Casablanca, Morocco

***Abstract*—In recent years, Big Data requirements have evolved. Organizations are trying more than ever to accent their efforts on industrial development of all data at their disposal and move further away from underpinning technologies. After investing around Data Lake concept, organizations must now overhaul their data architecture to face IoT (Internet of Things) and AI (Artificial Intelligence) expansion. Efficient and effective data mapping treatments could serve in understanding the importance of data being transformed and used for decision-making process endorsement. As current relational databases are not able to manage large amounts of data, organizations headed towards NoSQL (Not only Structured Query Language) databases. One such known NoSQL database is MongoDB, which has a high scalability. This article mainly put forward a new data model able to extract, classify, and then map data for the purpose of generating new more structured data that meet organizational needs. This can be carried out by calculating various metadata attributes weights, which are considered as important information. It also processed on data clustering stored into MongoDB. This categorization based on data mining clustering algorithm named K-Means.**

Keywords*—Big data; Data Lake; NoSQL; MongoDB; K-means; metadata

I. Introduction

Around the world, organizations are looking for a complete data analytics solution to cut costs, accelerate development cycles, and provide valuable information to solve certain of their biggest organizational problems. They view their data assets as an engine driving economic activity for competitive edge. Yet, the cost of running analytical solutions increases dramatically while deployment speed stands a primary challenge. There are vast amounts of data across multiple businesses resources, but suffering from difficulties for obtaining relevant and significant information within these resources [1].

Currently, most corporate data is stored in Data Lake, which holds all data (volume, variety) available in a fully centralized storage [2]. One of the problems of relying on Data Lake information is that the large amount of information stored in different sources makes it hard and complex to know its placement, meaning and source that it comes from. In other ways, it is extremely complex to comprehend data context for data consumers. Therefore, it becomes difficult to place confidence in its accuracy and veracity as well as to use it carefully [3] [4].

To solve this problem, organizations have implemented systems with a clustering strategy. This strategy consists of partitioning data, it splits data asset toward various subsets; these subsets termed clusters or groups. They are characterized by a high similarity inside and a high dissimilarity meanwhile other groups members [5]. This process aims to determine data internal structure, without any acquaintance of data features [6]. In this background, various approaches have been provided, the most known is K-means, it guarantees simplicity and capacity of processing large data sets [7]. This paper proposes an approach to achieve a clustering combined with metadata information. This can be accomplished through an edited variant of k-means clustering algorithm. It takes steps toward advancing the synergy between metadata and k-means clustering algorithm, and identifies pathways for developing a more cohesive metadata research agenda in data management.

This paper concentrates on various data sources centralized in Data Lake and analyzes them based on a common targeted schema [8]. These data are collected and mapped into NoSQL database named MongoDB. MongoDB is able to store larger amount of data compared to SQL (Structured Query Language) databases [9]. On the entire data collected from Data Lake and stored in MongoDB, K-Means algorithm is applied for data classification and clustering.

A. Research Motivation

The main motivation of this research is to propose a new strategy to extract the hidden valuable information from unstructured data and to classify it into clusters to make intelligent decisions and for predictive analysis.

B. Objectives and Contribution

This study is conceived for data predictive analysis and classification, by supplying a metamodeling classification for unstructured data. A classification algorithm has been set up for the purpose of emphasizing unstructured data preprocessing. Thus, the contributions of this research study can be identified as:

- A manual strategy was defined to extract data from various data sources as source towards MongoDB as target. The purpose of this study is to present a method that transform unstructured to structured data.
- An approach for analyzing Big Data metadata, which allows selecting blocks of information amongst heterogeneous sources and data repositories. Then, applying unsupervised learning skills of data mining for data clustering.

II. Related Works

Data classification generally deals with large data sets and many attributes. In real life, classification does not always handle homogeneous data. Most often, heterogeneous ones are involved [10]. On the one hand, MongoDB data can be considered as homogeneous data, if only the types are taken into account. Nevertheless, on the other hand, documents can be treated as heterogeneous data if the metadata attributes are also taken into account.

As regards the heterogeneous clustering data sets, an eminent approach was adopted by Modha and Spangler [11] for obtaining an appropriate data grouping which integrates several subsets of heterogeneous attributes in k-means clustering algorithm, they calculate weights attributed to different subsets attribute, which minimize simultaneously the average within a cluster spread and maximize cluster average spread along all subsets attribute [12]. Yet, the influence control in this case specified by a subset of attributes on others is lost. In a particular case, a subset of attributes in a metadata field include fewer attributes than subset coming from data description. Thus, metadata subset influence have to be checked and calculated separately.

Recently, many researches have been presented in machine learning database improvement area and data base storage techniques. The presented research in [13] and [14] prove the power of machine learning algorithms to deal with storage and retrieval using document-based or relational data storage. Other researches concentrate on enhancing data storage performance for data sets, they illustrate the necessity of distributed computing systems improvement. Researches in [15] and [16] improved the ability of processing large data sets thanks to distributed computing processes.

MongoDB has promoted the introduction of NoSQL databases. They are able to solve problems cannot be resolved with relational databases. They can manage large amount of data coming for different sources, several researches have worked in this field for proposing methods and approaches to deal with relation databases issues. Saran Raj proposed a model to categorize and store data into MongoDB using hashing algorithm [17]. Colombo and Ferrari have proposed a model operating integration at document level into MongoDB, and established an access operating control model at field level able to support context- and content- based on access control policies like to those of Oracle VPD (Virtual Private Database) [18].

Actually, according to authors' knowledge, most of researches contributions in the field of data classification and storage treat this subject separately. They consider MongoDB to improve unstructured data storage into structured format. Otherwise, others researches focus on k-means algorithm for data clustering of structured data without taking into account a common schema. Besides, they have not considered combining the two mechanisms to deal with effectiveness data classification and management. This work sheds light on the described processes above for the aim of giving a global view of the completed process from storage to clustering.

III. K-Means Algorithm using Metadata Analysis

A. Metadata Analysis

Metadata is a data description (data about data). It affords information on certain element content. For example, an image can contain metadata that describes image size, color depth, image resolution, image creation date, and other data. The metadata of a text document can describe information about document length, its author, its creation date and a document brief summary [19].

Researchers often use clustering or classifications for metadata analysis. Supervised classification is applied to group data in the basis of a preexisting classification. Clustering, furthermore, is a data unsupervised classification into groups based on internal characteristics or attributes similarities.

K-means is one of the most used algorithm in clustering, particularly, text clustering [22]. Due to the diversity of fields in which clustering is used and specific conditions to every application, there are many variants of standard algorithm proposed in literature. However, it is necessary to develop metadata founded on a standard diagram. While generating metadata, using DCME (Dublin Core Metadata Element) [20], 15 elements were proposed to develop metadata for unstructured data. These elements are described in Table I.

TABLE I. 15 elements in DCME

Title	Explanation	Date
Identifier	Creator	Publisher
Type	Source	Subject
Contributor	Format	Language
Relation	Location	Rights

DCME was recommended as a standard for metadata development in this study for these reasons:

- Focus on important element only;
- Easy and simple to use;
- Smoothly understandable by all users;
- Can be used for several domains and topics;
- Used elements are understandable and meaningful;
- Element reused is allowable.

B. K-means Algorithm

In Big Data and machine learning world, there are lot of algorithms in supervised, semi-supervised and unsupervised learning. They are used for several goals such as prediction, pattern recognition, classification and clustering. The use of these algorithms is endless and can include any field of study [21]. This paper concentrates on unsupervised learning, particularly clustering. The clustering concept in Big Data handles by identifying alike information without previous knowledge about data classes. One issue of clustering is that it is field of several interpretations and does not have an explicit definition.

K-means is one of the widely used algorithm for clustering, its aim is to divide data set into k distinct pieces. The objective of k-means is to determine similarities between elements of

data set and to group them according to a similar aspects function. The most applied function is the Euclidean distance function, but different function expressing similarity can be used. K-means is a useful and an interesting feature since it is simple and speed. It is essentially an optimization problem where an optimal local grouping can be achieved, while a global one is not guaranteed [22]. Historically, researchers introduced k-means from several disciplines. The most known researcher to have been the author is Lloyd (1957, 1982), with Forgey (1965), Friedman and Rubin (1967) and McQueen (1967).

C. Basic K-means Algorithm

K-means is one of the easiest unsupervised learning algorithms solving clustering problem. The process follows an easy and simple manner to group a specific data set across a number of fixed (presumably R clusters) a priori. The flowchart is shown in Fig. 1. The main goal is to determine k centers, one at each cluster. These centers must be located cleverly since diverse location leads to diverse result. Therefore, the best option is to place them separately and in different locations. The following step is to take each point pertaining to a specific data set and link it with the proximate center. When no point is outstanding, the first step is achieved and a precocious group age is performed. At this point, we require recalculating k new centroids as clusters barycenter ensuing from the preceding step. Once getting these k new centroids, a new link must be made among the same points in the data set and the nearest new center. This mechanism generated a loop. As a result, it can be perceived that the k centers modify their location step by step until no more transformation are made. In other terms, the centers no longer move [23].

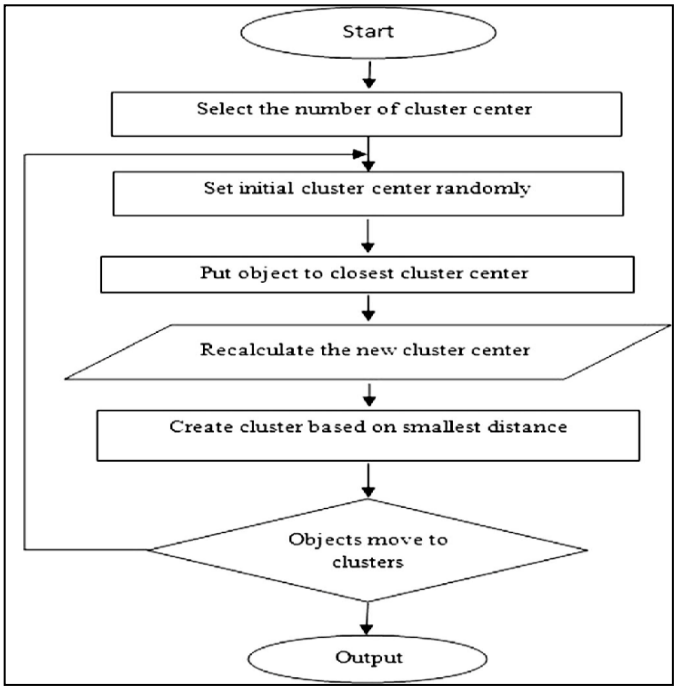

Fig. 1. Flow chart of k-means Clustering Algorithm.

The standard K-mean clustering Algorithm is as below:

Input:

$U = \{n1, n2, n3, \ldots\ldots\ldots\ldots n\}$

// Set of n number of data points.

Value of *R*

// Number of desired Cluster.

Get random centroid or Define initial centroid.

Output:

A set of *R* Clusters.

// Number of desired Cluster.

IV. Specific Objectives of Proposal System

The main purpose of the presented system concentrated to offer provided task to users. It stores huge amount of unstructured data into MongoDB database. It highlights below the major elements performed during this work:

1) Extracting unstructured data from the source.
2) Storing unstructured data into MongoDB.
3) Managing data clustering and categorization requests using K-means algorithm.

A. Architecture Design

Today, databases have taken lot of forms - from managing structured data to unstructured data. Many advanced applications are mainly web-based and need a scalable database to manage various types of data such as email, images, newspapers and video files.

MongoDB is one of the popular database management system conceived for Internet infrastructure and web applications. It is designed for high read and write speed and can be easily upgraded. When applications require a single or dozens of database nodes, MongoDB offers better performance and stability compared to relational databases [24].

Java, furthermore, is an open source software language used by developers to build and deploy scalable applications using Java development services. In the proposed system, MongoDB is integrated with Java technology for storing data is a first step, a second step illustrates K-means Java program algorithm development for data clustering.

Fig. 2 depicts the proposal system architecture using already discussed technologies.

Fig. 2 explains the task of obtaining unstructured data from data sources and analyzing them by the system provider. Then storing them into MongoDB using Java MongoDB driver. Once the storage process has completed, all data is collected in MongoDB will be categorized using K-means algorithm Java program with combination of metadata attributes. After categorization method is achieved, all data are clustered into various clusters based on a chosen metadata attributes [25].

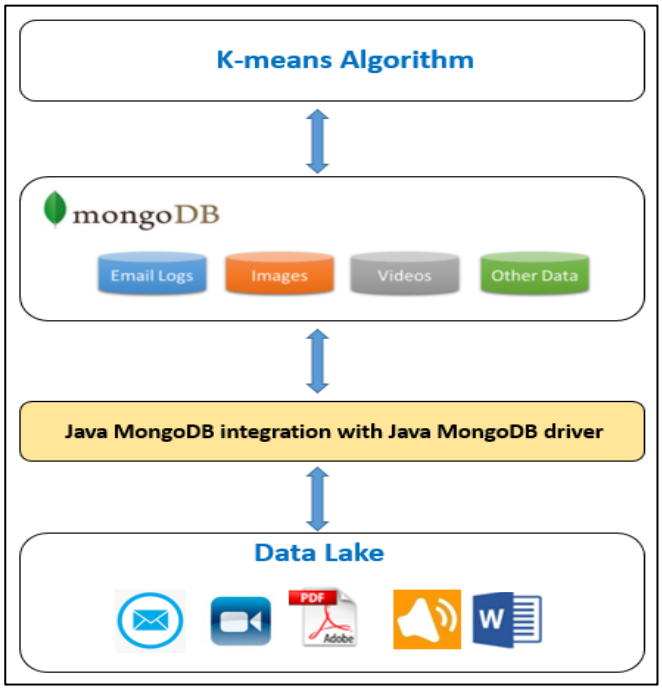

Fig. 2. Proposal Architecture Integration & Development Design.

B. Data Flow Diagram

DFD (Data Flow Diagram) is a type of graphical representation of data flow through an information system. This tool is often used as a preliminary step in designing a system to create an overview of this information system. In addition, it is used to visualize data processing (structured design).It shows what type of information enters (input) or leaves (output) the system, where it comes from and where it is stored. However, it does not indicate the timing of data transmissions, nor order in which data flows [26].

DFD uses defined symbols as rectangles, circles and arrows, as well as labels with short labels, to represent data inputs, outputs, storage points and paths between each destination. From simple overviews, even hand-drawn processes, to complex diagrams on several levels which progressively deepen data processing. It can also be used to analyze an existing system or to model a new one. Like all quality diagrams and charts, a data flow diagram can often visually "say" things that might be difficult to explain in words.

1) DFD level 0: The DFD level 0 illustrates input, system and output of this study. Since this project uses unstructured data, it should be stored in MongoDB database. Fig. 3 presents an easy way to explain project. Two system functions are launched to improve project work. Firstly, obtaining source from the origin and send it for storage in MongoDB database systems. In this ingestion method, many analyzers are used to convert data to text format. Second system function consists of clustering process using K-means for obtaining data groups based on metadata categorization.

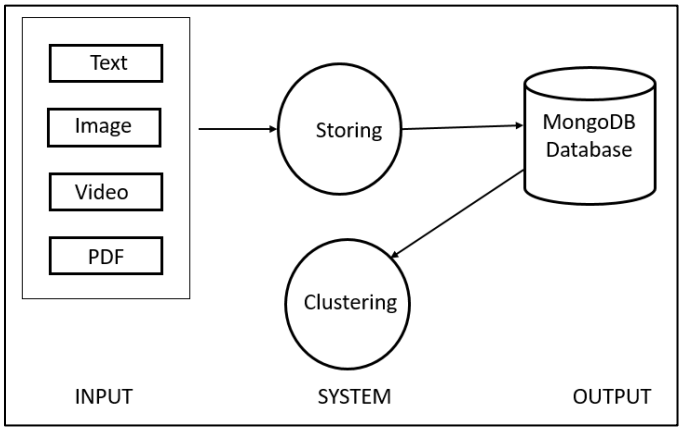

Fig. 3. DFD Level 0.

Not all unified storage systems are created equally. To successfully unify storage transition, and gain investment and risk, the solution must be built for organization. Virtualized storage and intelligent file exhaustion must be integrated to guarantee an availability close to 100%.

2) DFD Level 1: It is noted that when creating DFD level 1, the relationship between system and environment is not eliminated. In other words, system data incoming and outgoing flow must be the same compared to data provided in DFD level 0. Thus, data flow created in DFD level 1 have to be added in DFD level 0.

Fig. 4 describes system functioning mechanism and various function setting it up. First of all, data is routed toward MongoDB database using **Storage of unstructured data** function. After storage process, a parallel process consists of categorizing data while storing titled **Data clustering**, this latter is launched to categorize ingested data using metadata information.

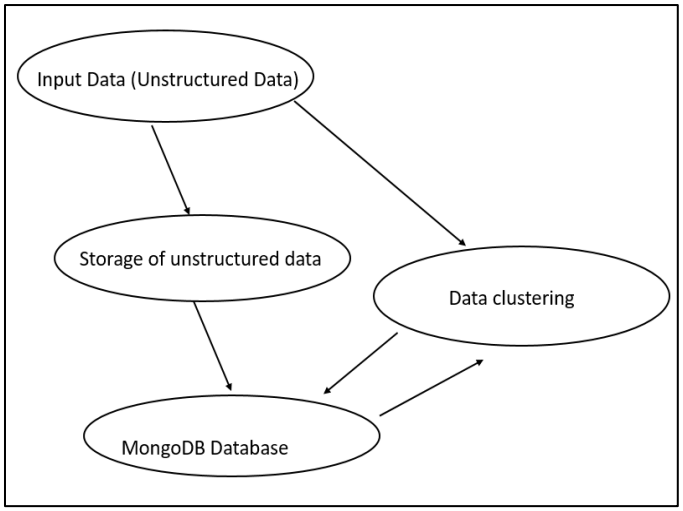

Fig. 4. DFD Level 1.

C. Servers Availability Process

Supposing that one or two servers are unavailable, the metadata cluster becomes read-only. Read and write functions can be applied on data from fragments, but no splits or block migration will happen before all servers' availability. In this case, clusters can still be used just if Mongo instances are available, until after the configuration servers are reachable. If mongo instances are restarted before configuration server's availability, servers will not be able to send reads and writes. Clusters become unusable without metadata cluster. To guarantee that configuration servers continue being intact and available, backups of servers are essential. Configuration server data is small compared to data stored in a cluster, and the configuration server has comparatively low load process. These properties make it easier to find a way to back up configuration servers. One of the known processes to ensure systems availability is load-balancing notion. It is the operation of spreading network traffic through multiple servers [27]. This ensures no single server supports too much demand and increases applications and databases availability for users.

V. Implementation and Evaluation

To validate our K-means method for clustering unstructured data stored into MongoDB, a system was developed with Java programming language. For the constants used in K-means algorithms, the following values were chosen: C1= 0.7; C2=0.2

This system was designed and tested under a server configuration defined bellow:

Hardware Requirements:

System: Quad core processor 2.80GHz.
Hard Disk : 500 GB.
RAM : 8 GB.

Software Requirements:

Operating system: Windows 10
Coding Language: Java 8
Database: MongoDB
IDE: Intellij Idea 2017.2.2
The project is available in github under the link:
https://github.com/jabrane2005/MongoDBClustering

A. Running MongoDB

For tests, samples extracted documents from local hard disc were used. The files fields available are: filename, aliases, chunkSize, uploadDate, length, _id, contentType and md5.

From these fields have been used for tests only: contentType for the k-means metadata algorithm. The Fig. 5 illustrates a MongoDB Java program application for data connection:

The large amounts of unstructured data available in hard disc are used in this program to storing into MongoDB database.

Fig. 6 illustrates a Java program for saving different data:

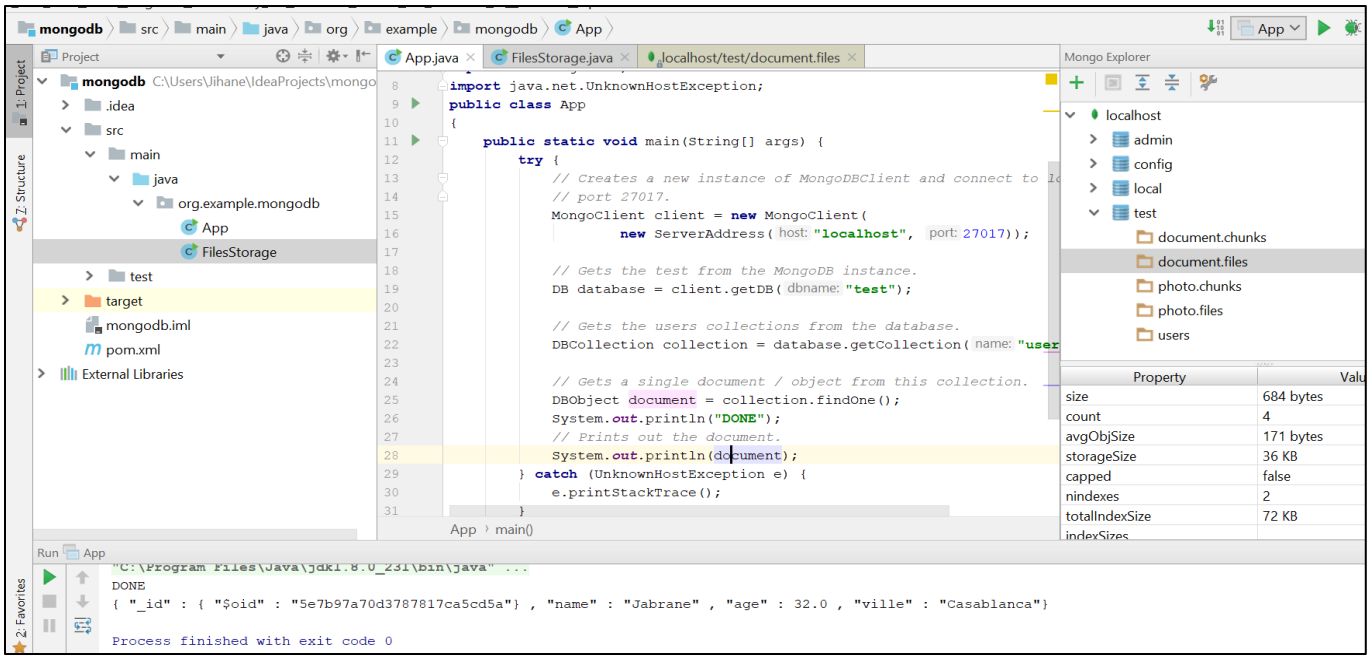

Fig. 5. Java Program for MongoDB Connection.

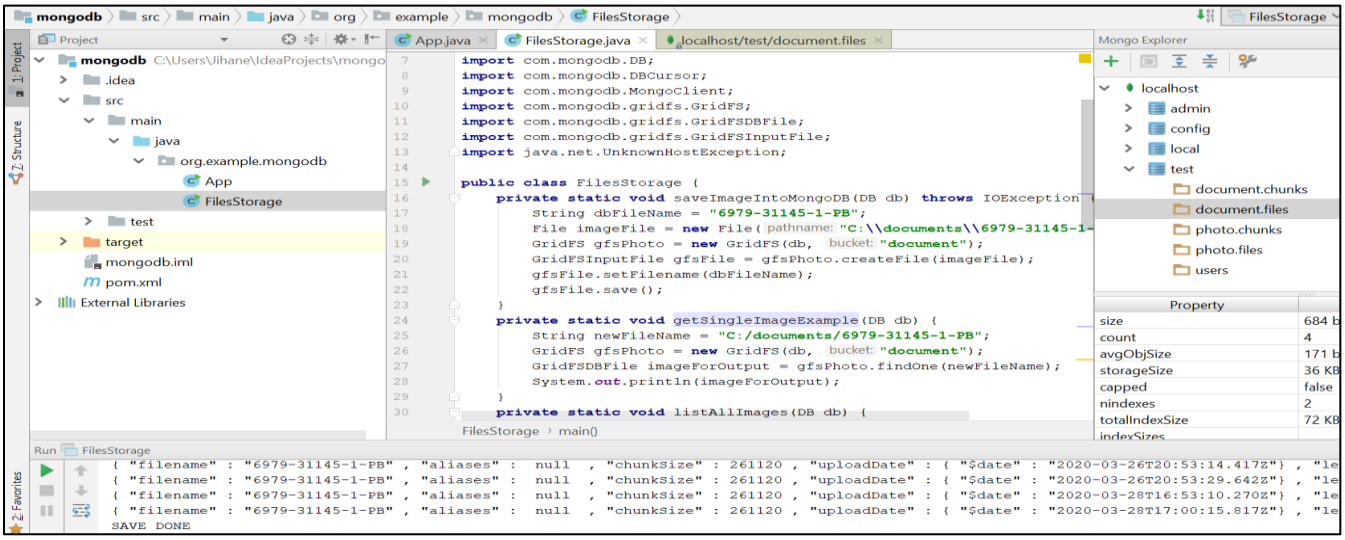

Fig. 6. Java Program for Saving Data into MongoDB.

B. Running K-means Algotithm based on Metadata

Fig. 7 illustrates java program of K-means algorithm. Three classes were developed for that purpose :data, cluster and the main class AlgoKMeans that contains the executable method. This program creates a Kmeans cluster with a given clusters number and iterations. The internal random generator is based of current system time. For the distance, the Euclidean n-space distance is used.

Various tests have been performed and presented as described below for representative tests only. For each test case, two clustering were executed. First, a K-means clustering with the standard model was executed (without taking into account the metadata), then a K-means clustering based on this study developed model was executed (where the metadata were taken into account).

For every type of clustering, 10 runs were executed and the best grouping result was taken for analysis. For measuring each grouping quality, the pairwise average similarity was used in each algorithm. To compare tests quality results of the two clustering techniques, F-mesure is used. After clustering using the two algorithms, the F-mesure values from Table II were resulted.

The both of algorithms had a success rate as shown in Table II. As noticed, in clusters in which metadata was applicated, average result for F-mesure is closer to 1, that means that the distribution of documents in clusters is good. It can be concluded that documents distribution in clusters is nearly obtained as the same as in data source.

The purpose of this clustering, however, is not obtaining identical clusters of classification which already exists, but to obtain groups of documents with similar content. F-measure is used to compare results quality of both clustering algorithms, using the same reference: MongoDB. The second test was executed on a sample involving five clusters. After clustering, values of Table III have been obtained using F-measure.

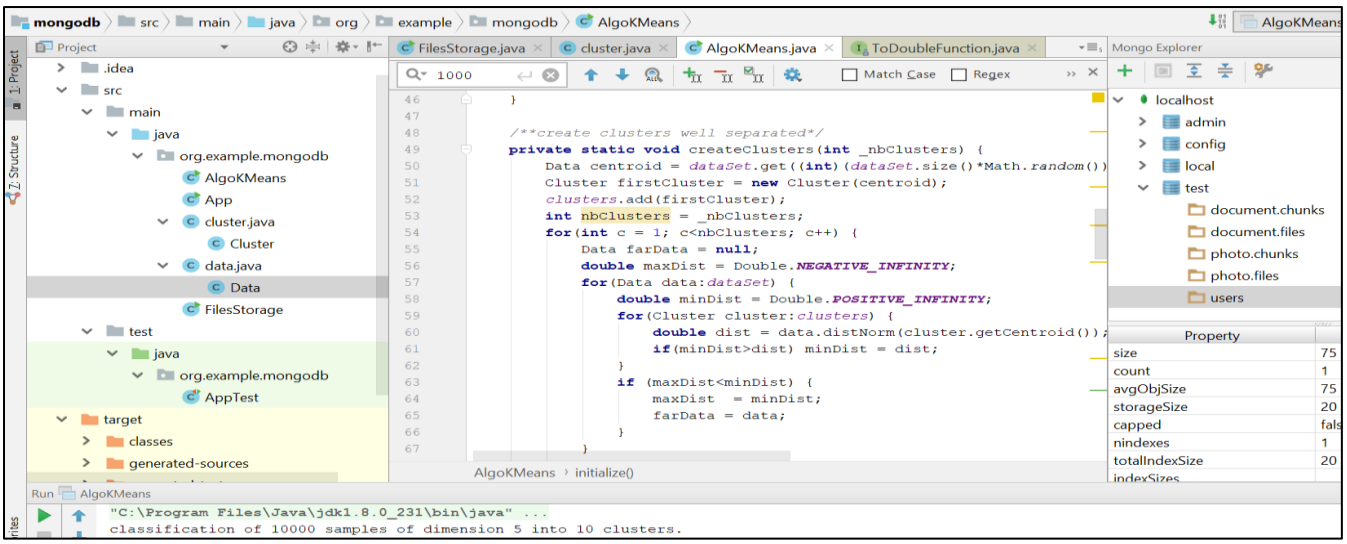

Fig. 7. Java Program for K-means Algorithm.

TABLE II. F- MESURE VALUES FOR 10 RUNS WITH K=2

Run	K-means without Metadata	K-means with Metadata
1	0.9989665446774567	1
2	1	1
3	0.9994701063598654	0. 9987668654568893
4	0.9989665446774567	1
5	0.9984566755334567	1
6	0.9994701063598654	1
7	1	1
8	0.9994701063598654	0.9987997769063541
9	0.9989403089137634	1
10	0.9973456789000877	0. 9989648949426297

TABLE III. F- MEASURE VALUES FOR 10 RUNS WITH K=5

Run	K-means without Metadata	K-means with Metadata
1	0.9584647272670947	0.9387867656481974
2	0.7544678488490202	0.7148788909088644
3	0.7191948473652238	0.7953588600765327
4	0.7009874232267489	0.6367996432456745
5	0.5563527902983746	0.8045432578967342
6	0.7646252782993902	0.8165879899065689
7	0.7073629289736697	0.5589651267975943
8	0.9649297646829207	0.6246865379965346
9	0.8127264940498373	0.7134778987449867
10	0.7493837363636892	0.7642457543457756

The comparison of the two results of F-measure values from Table III illuminates the standard clustering algorithm highest value is 0.951, comparing by that of clustering using metadata is 0.938. It is also noticed that if the averages of the F-measure values from each results are respectively: 0.7688 and 0.7408. That means that both of clustering algorithms produce practically good results.

By parsing 500 files with both K-means algorithms, result in Table IV is obtained.

From the results above, the two algorithms values are nearly the same, but the number of documents in clusters using K-means algorithm with metadata are less than those without metadata, wich means that the clustering is more efficiant using metadata. It is concluded that it is benefit to use clustering with metadata for grouping documents by content.

TABLE IV. NUMBER OF FILES IN EACH CLUSTER

Number of files in Cluster	K-means without Metadata	K-means with Metadata
1	341	366
2	95	82
3	40	37
4	23	15
5	1	0

VI. CONCLUSION AND FUTURE WORK

The main contribution of this research is the transformation of Data Lake unstructured data into structured data using a NoSQL database. MongoDB database has been chosen for this research. Authors have firstly proposed a method to extract and store data into MongoDB using Java programming language. The second step focuses on setting up a clustering technique, in particular K-means, to create clusters responding to specific necessities. Experiments are performed using Java system proposing these functionalities. Various tests are made to evaluate approach effectiveness. The obtained results confirm that clustering techniques benefits from using metadata to increase performance of decisional queries clustering. In future work, authors look forward considering all these proposed approach for the aim of combining these clusters with Data Warehouses to take advantages of existing reports and dashboard for a decision-making efficiency.

REFERENCE

[1] P.Lake and R.Drake, “Information Systems Management in the Big Data Era”. London: Springer, 2014.

[2] J. Kachaoui and A. Belangour, “Challenges and Benefits of Deploying Big Data Storage Solution”, Proceedings of the New Challenges in Data Sciences: Acts of the Second Conference of the Moroccan Classification Societ, Article No.: 22, pp 1–5, 2019.

[3] J. Kachaoui and A. Belangour, “An Adaptive Control Approach for Performance of Big Data Storage Systems”, International Conference on Advanced Intelligent Systems for Sustainable Development, pp 89-97, 2019.

[4] J. Kachaoui and A. Belangour, “A Multi-criteria Group Decision Making Method for Big Data Storage Selection”, International Conference on Networked Systems, pp 381-386, 2019.

[5] P. Arora1, Dr. Deepali and S. Varshney, “Analysis of K-Means and K-Medoids Algorithm For Big Data”, International Conference on Information Security & Privacy (ICISP2015), 11-12 December 2015.

[6] M. Wria, M. Salih and M. Saraee, “Mining Semantic Web Data Using K-means Clustering Algorithm”, British Journal of Mathematics & Computer Science? 13 (1), pp 1-14, 2016.

[7] M. Boussahoua, O. Boussaid and F. Bentayeb, “Logical Schema for Data Warehouse on Column-Oriented NoSQL Databases”, International Conference on Database and Expert Systems Applications, pp 247-256, 2017.

[8] J.Kachaoui and A. Belangour, “From Single Architectural Design to a Reference Conceptual Meta-Model: An Intelligent Data Lake for New Data Insights, unpublished.

[9] A. Chauhan, “A Review on Various Aspects of MongoDb Databases”, International Journal of Engineering Research & Technology (IJERT), Vol. 8 Issue 05, pp 90-92, 2019.

[10] Y. Al-Sharo, G;Shakah, Mutasem Sh. Alkhaswneh, B. Naeidi and M. Alazzam, “Classification of Big Data: Machine Learning Problems and Challenges in Network Intrusion Prediction”, International Journal of Engineering & Technology, 7 (4) ,pp 3865-3869, 2018.

[11] D. Modha, S.W. Spangler, “Feature weighting in k-means clustering. Machine Learning”, 52 (3), 2003.

[12] P. Berkhin, “A Survey of Clustering Data Mining Techniques. In: Grouping Multidimensional Data”, pp. 25–71, 2006.

[13] J. Manurung, H. Mawengkang, and E. Zamzami, “Optimizing Support Vector Machine Parameters with Genetic Algorithm for Credit Risk Assessment,” Journal of Physics: Conference Series, vol. 930, no. 1, p. 012026, Dec. 2017.

[14] N. Venkateswaran and S. Changder, “Simplified data partitioning in a consistent hashing based sharding implementation,” presented at the TENCON 2017 - 2017 IEEE Region 10 Conference, 2017, pp. 895–900.

[15] C. Krome and V. Sander, "Time series analysis with apache spark and its applications to energy informatics," Energy Informatics, vol. 1, no. 1, p. 1, 2018.

[16] Q. Huang, H. Gudmundsdottir, Y. Vigfusson, D. A. Freedman, K. Birman, and R. van Renesse, "Characterizing Load Imbalance in Real-World Networked Caches", New York, USA: ACM, 2014, pp. 8–7.

[17] R. Saran, "Storing of Unstructured data into MongoDB using Consistent Hashing Algorithm", Thesis 2015.

[18] P. Colombo and E. Ferrari, "Fine-Grained Access Control Within NoSQL Document-Oriented Datastores", DOI 10.1007/s41019-016-0015-z,2016.

[19] F. Qayyum and M.Afzal, "Identification of important citations by exploiting research articles' metadata and cue-terms from content", Scientometrics ,https://doi.org/10.1007/s11192-018-2961-x, 2018.

[20] A. Haraty,1 M. Dimishkieh,1 and M. Masud, "An Enhanced k-Means Clustering Algorithm for Pattern Discovery in Healthcare Data", International Journal of Distributed Sensor Networks,Article ID 615740, 2014.

[21] T.N. Gnanaraj, K.R. Kumar and N.Monica, "Survey on mining clusters using new k-mean algorithm from structured and unstructured data", International Journal of Advances in Computer Science and Technology, Volume 3, No.2, 2014.

[22] A. Fadaei and S. Khasteh, "Enhanced K-means re-clustering over dynamic networks", Expert Systems With Applications 132 ,pp 126–140, 2019.

[23] H. Abbes and F. Gargouri, "Big Data Integration: a MongoDB Database and Modular Ontologies based Approach", 20th International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2016, 2016.

[24] J.Kachaoui, J. Larioui and A. Belangour, "Towards an Ontology Proposal Model in Data Lake for Real-time COVID-19 Prevention Cases", International Journal of Emerging Technologies in Learning (iJET), unpublished.

[25] J.Kachaoui, J. Larioui and A. Belangour, "MQL2SQL: A Proposal Data Transformation Algorithm from MongoDB to RDBMS", unpublished.

[26] I.Rosziati and Y.Siow, "Formalization of the Data Flow Diagram Rules for Consistency Check", International Journal of Software Engineering & Applications (IJSEA), Vol.1, No.4, 2010.

[27] B.Mallikarjuna and A.Doddi, "The Role of Load Balancing Algorithms in Next Generation of Cloud Computing", Journal of Advanced Research in Dynamical and Control Systems 11(7):1715-1733,2019.