JST: Smart Systems and Devices
Volume 33, Issue 2, May 2023, 009-017

Building and Implementation of a Big Data Mining Model for Businesses

Do Tran-Truong, Nguyen Xuan Dung, Phuong Xuan Quang, Vinh Tran-Quang*
Hanoi University of Science and Technology, Ha Noi, Vietnam
**Corresponding author email: vinh.tranquang1@hust.edu.vn*

Abstract

Today's businesses are facing a number of difficulties in statistics, analysis, and processing of their data sources to make appropriate business decisions. The reason is that the data source of the enterprise is stored discretely in many file types with different structures and is not unified. In this paper, we design and build a model of a centralised storage system, using big data mining to provide business data analysis functions according to their business requirements. To test and evaluate the effectiveness of the proposed model, we use the input data which is the actual business data set of an accessory business. The results when the model is applied show that the source data sets are organised, stored, and analyzed with many different criteria, and are displayed on the charts in an obvious and detailed way. Furthermore, we also compare the proposed model with some existing models. The results show that the proposed model is easy to use for end-users. It has high scalability and fault tolerance, and a faster processing speed compared to traditional models.

Keywords: Business intelligence, data warehouse, data lake, big data, distributed storage and processing.

1. Introduction

The digital era has meant that the availability of appropriate information and knowledge has become critical to the success of the business. The next information revolution is about information content and its purpose. However, organizations must adapt in order to survive and succeed as their business domains, processes, and technologies change in a world of increasing environmental complexity [1]. Enhancing the performance and competitive position by improving the ability to respond quickly to rapid environmental changes with high-quality business decisions can be supported by exploiting technologies such as data warehouse and Business intelligence (BI) analytical tools. However, ensuring business intelligence basically requires having relevant, reliable, accessible, accurate, timely, complete, coherent, and consistent quality information on the decision at hand. Hence, business intelligence through decision making improvement is a major concern for business managers nowadays. BI is a process that includes two primary activities: getting data in and getting data out [2], as the model in Fig. 1.

The amount of data generated every day is expanding dramatically. Big data is a popular term used to describe the data which is in zetta bytes. Government companies and many organisations try to acquire and store data about their citizens and customers in order to know them better and predict the customer behaviour. Social networking websites generate new data every second and handling such data is one of the major challenges companies are facing. Data which is stored in data warehouses is causing disruption because it is in a raw format, proper analysis and processing are to be done in order to produce usable information out of it [3].

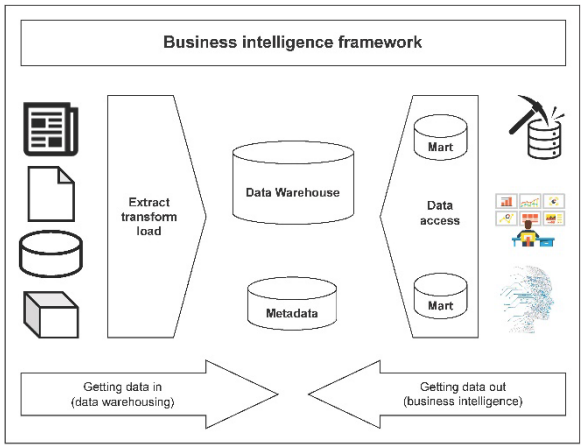


Fig. 1. Business intelligence framework.

Due to such a rapid increase in data, commercial business is an increasingly complex and competitive environment. Businesses must compete with each other because the needs of customers are changing unpredictably. Businesses must come up with appropriate solutions in a timely manner to keep up with market changes. To overcome this condition, the business needs to make the right decisions required to deal with these rapid changes by analysing the vast data sources that have been generated.

ISSN: 2734-9373
https://doi.org/10.51316/jst.166.ssad.2023.33.2.2
Received: March 14, 2023; accepted: March 25, 2023

The development of data warehouse is a way to extract the important information from the scattered data in some information systems with a centralized integrated storage and support the need for data history. These integrated data can be used for information delivery activities that can be reviewed from various dimensions and set to a level of detail [4].

The further utilisation of the information contained in the data warehouse is the activity of data analysis using certain techniques and methods. There are several algorithms for knowledge data discovery, like classifying, clustering, and mining [5]. The data contained in the data warehouse can be used as input to the application system, for example, as a dashboard [4]. With this dashboard, it is expected to be a solution for the business process to monitor the financial condition. Given the limitations of traditional techniques used so far and the new data requirements, enterprises face several challenges in managing large volumes of data. The concepts of Data Warehouse and Big Data tend to blend and it is not easy to find a divide between them [6]. While Data Warehouse is a mature management paradigm supported by widespread and well- established methodologies [7-9], Big Data is still a field under development, which seeks to address individual aspects of the problem but still lacks an integral solution. To solve this problem, this paper aims to design and implement a modern data mining system for business to support data analysis process. The designed system accommodates the Hadoop platform to support parallel and distributed processing of large volumes of data, and most solutions involve Hadoop technology. In addition to that, the system is also designed with Oracle Server Database which is friendly and close to most people because it uses the SQL query language.

As a result, the business will have a dashboard to monitor the current condition of the financial atmosphere of the company. The dashboards will show us what is happening with the products in our company, consumer product numbers, consumer trends, and other in-depth reports.

The main contributions of this paper are given below.

- The designed system enables processing the data whether it is of any type of data.
- The system provides a solution to exploit mining statistics.
- The proposed system could be implemented for other companies that need a support system for analysis.

The remainder of this paper is organized as follows. Section 2 presents the related works. Section 3 presents the design of the system Section 4 present the deployment of the proposed system. Section 5 shows the experimental results and analysis. Section 6 presents a comparison with another model. Finally, the conclusions are drawn in Section 7.

2. Related works

The data warehouse is the combination of concepts and technologies that help organisations manage and maintain historical data obtained from operational and transactional applications [10]. It helps knowledge workers (executives, managers, analysts) to make quicker and more informed decisions [4]. Data warehouse is a new paradigm in strategic decision making environment. The data warehouse is not a product, but an environment in which users can find strategic information [11]. Data warehouses serve as a central repository for storing and analyzing information to make better informed decisions [12]. The data warehouse contains a collection of logical data separate from the operational database and is a summary. Data warehouse allows the integration of various types of data from a variety of applications or systems [6]. This ensures a one-door access mechanism for management to obtain information and analyse it for decision-making. The data warehouse has several characteristics [12, 13]: subject-orientated, integrated data, non-volatile, and variable in time.

Dimensional modeling is a construction model used for data warehouses. It is a call-based model that supports high-level query access. Star Schema is a form of dimensional modeling scheme that contains a fact table at its center and dimensional tables. The fact table contains a descriptive attribute that is used for the query and the foreign key process to connect to the dimension table. Decision analysis attributes consist of performance measures, operational metrics, aggregate sizes, and all other metrics needed to analyze organizational performance. The fact table shows what is supported by the data warehouse for decision analysis. The dimension table contains attributes that describe the entered data in the fact table [4, 13].

Extract, Transform, and Load (ETL) is a data integration process that extracts data from outside sources, transforms the data according to business needs, and stores them in the data warehouse [5]. Data used in the ETL process can come from a variety of sources, including enterprise resource planning (ERP) applications, flat files, and spreadsheets. However, since data warehouses are very large and take time to create, ‘Data Marts’ can be created. ‘Data Marts’ are smaller than data warehouses and are intended to store data from a part of the organization (i.e., a department in the enterprise). The data warehouse will store data for the entire company. These data marts can be built separately. Or, a part of the data warehouse intended for a specific function or department can be extracted to create a data mart [12]. With the arrival of big data, the traditional data warehouse cannot handle a large amount of data [15]. So, the concept of Data Lake was

born to solve the problems that Data Warehouse was unable to solve.

The basic idea of Data Lake is simple; all data emitted by the organisation will be stored in a single data structure called Data Lake. Data will be stored in the lake in their original format. Complex pre-processing and data loading transformation will be eliminated. The upfront cost of data ingestion can also be reduced [6]. Once the data is placed in a lake, the organization can tap into that raw data source for a variety of business purposes [14]. The authors in [15] suggested more specifications for Data Lake, especially from the point of view of the business domain rather than the research community. The authors in [15] suggested more specifications for Data Lake, especially from the point of view of the business domain rather than the research community. All data are loaded from the source systems, no data are turned away, and data are stored at the leaf level in an untransformed or nearly untransformed state.

The distinct characteristic of Data Lake is that it attracts more attention from business fields than from academic research fields. Data Lake is a relatively new concept even for the big data domain [16]. A Data Lake brings a variety of capabilities to the enterprise by centralizing the data. With data being centralized, the enterprise can tap into capabilities that have not yet been explored. These data can help enterprises with much more meaningful business insights compared to any single system in the enterprise [17].

The integration between big data technology such as Hadoop and data warehouse is essential [4]. To support parallel and distributed processing of large volumes of data, most solutions involve Hadoop technology [18]. The Hadoop framework is used by many big companies such as Google, Yahoo, IBM, for applications such as search engine, advertising, and information gathering and processing [3]. It is capable to perform analysis of large heterogeneous datasets at unprecedented speeds [3]. When we have data storage and processing system, we will need data visualization. In recent years, data visualization has been a staple topic of discussion in libraries and in the broader world of business and journalism. PowerBI is a typical tool in this field [17].

Once data have been shaped and configured by PowerQuery, the tables can be loaded to the Power BI visualization layer. Multiple tables can be used in a given set of visualizations and can be linked together in a “data model” similar to those found in traditional relational database systems. It is not necessary to put all data into a single table before beginning exploratory analysis and visualization. A Power BI report contains one or more pages on which one or more visuals can be grouped. Creating a visualisation is a drag-and-drop process; the tables and fields of available data are listed in a side panel, and a second pane displays a selector for the form of the visualisation and blanks for the data. Data fields can be easily moved between axis labels, legends, and area values without redefining the underlying table.

Currently, we have collected data from two model systems. The first model is presented in [4]. The model in this paper presents an overview of a common data mining model and big data to get reports, the model has the function of aggregating all kinds of data into a common place. Integrated construction between the conventional database management system and Hadoop’s HDFS distributed file storage technology. The second model in [12] presents a traditional data mining model; This model can only handle structured data, and the storage capacity and scalability of this model are quite limited. In turn, with traditional storage capabilities, this model allows compatibility, stability, and ease of use for all audiences.

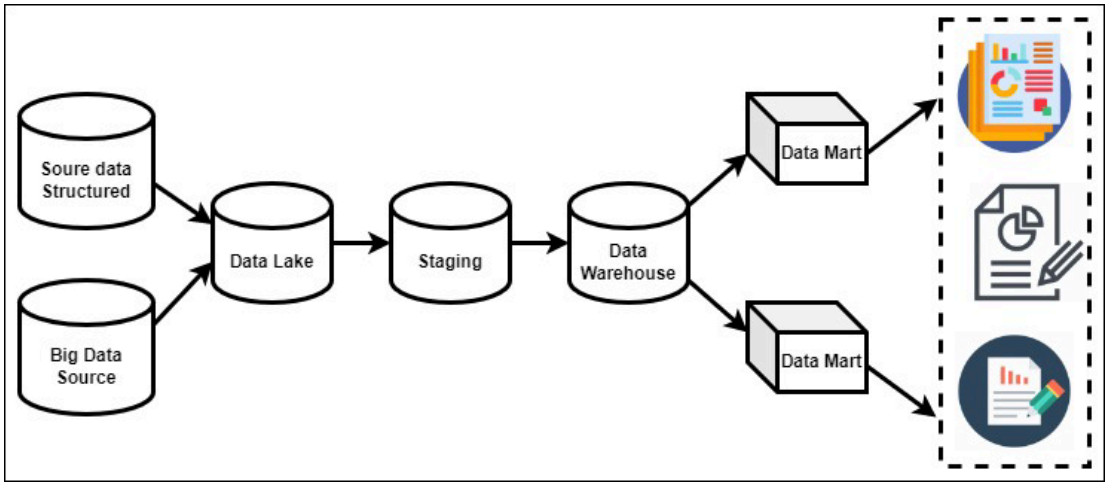


Fig. 2. The proposed system architecture.

3. Proposed Solution

3.1. System Model Design

The proposed model is shown in Fig. 2. The source data goes through the Data Lake, Staging, Data Warehouse, and Data Mart blocks to get the data report that must follow different processes and data flows, after running through those data streams, the parameters and visual reports will be presented through a dedicated reporting tool, which allows users to simply do the reporting without worrying about how the data flow runs.

The data stored as files will be centralized in the Data Lake, which is a distributed file storage place. Here, we will proceed to restructure all the data in a structured form. The process of data being transported into the Data Lake that shown by Flowcharts in Fig. 3. We will proceed to put the data into staging to store temporary data before being processed and put into the Data Warehouse. The process of data being transported into the staging. The flow chart that puts the data into Staging is shown in Fig. 4.

The processing of putting data from Staging to Data Warehouse follows in Algorithm 1.

Algorithm 1. The data flow into the Data Warehouse.

Require: data are fully in the Staging region
Ensure: data is ETL in accordance with the business
1: *current_server* ← *currentServer*(Staging)
2: *count_table* ← *countTable*(Staging)
3: **for** *tableNumber* = 1, 2, . . . , *count_table* **do**
4: *currentTable* ← *nameTableStaging*(*tableNumber*)
5: **if** *rowsCount*(*currentTable*) == 0 **then**
etlDataLakeToStaging(*currentTable*)
6: **end if**
7: **end for**
8: *current_server* ← *currentServer*(*DataWarehouse*)
9: *count_table* ← *countTable*(*DataWarehouse*)
10: **for** *tableNumber* = 1, 2, . . . , *count_table* **do**
11: *currentTable* ← *nameTableWarehouse*(*tableNumber*)
12: *sizeTable* ← *sizeTable*(*currentTable*)
13: **if** *limitCapacityWarehouse*(*sizeTable*) **then**
etlWarehouseToBackup(*currentTable*)
truncate(*currentTable*)
14: **end if**
15: **end for**
16: **for** *tableNumber* = 1, 2, . . . , *count_table* **do**
17: *currentTable* ← *nameTableWarehouse*(*tableNumber*)
18: *businessTable* ← *businessTable*(*currentTable*)
19: *selectTable*(*businessTable*)
20: *selectTable* ← *selectTable*(*businessTable*)
21: *etlStagingToWarehouse*(*selectTable*, *businessTable*)
22: **end for**

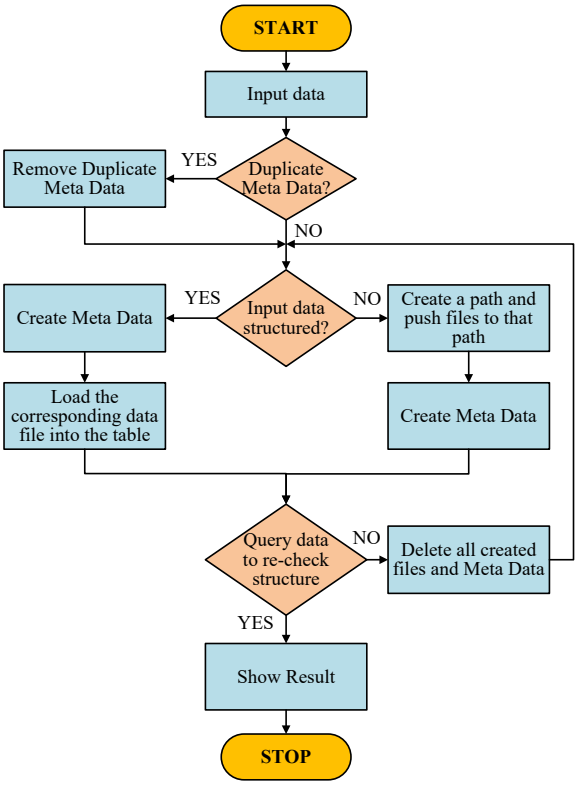


Fig. 3. Flowchart of data flow into Data Lake.

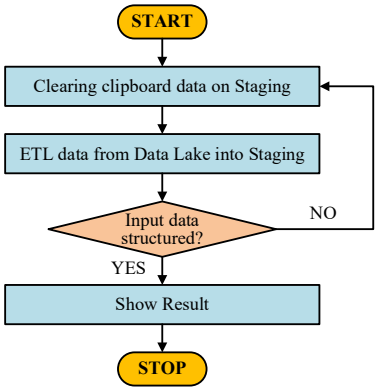


Fig. 4. Data flow into Staging.

3.2. Storage Model Design

The Data Lake model is organized and stored by a cluster of three servers, in which one server plays the role of master and the other two servers act as slaves. To deploy Data Lake, we use the distributed file storage system (HDFS) and design the Data Lake storage model as in Fig. 5.

To take advantage of in-depth data according to each departmental operation, the data of Data Mart according to the flow chart shown in Algorithm 2.

After finishing the data pipeline, we use the reporting tool to extract the statistical report. Staging is a container for data that is transported in the Data Lake and has a transformation in value. From the

information of the source tables, we designed tables to store data in the staging area as in Fig. 6.

Algorithm 2. The data flow into the Data Mart

Require: data are fully in the Data Warehouse
Ensure: data is ETL in accordance with the business
1: *current_server* ← *currentServer*(DataMart)
2: *count_table* ← *countTable*(DataMart)
3: **for** *tableNumber* = 1, 2, . . . , *count_table* **do**
4: *currentTable* ← *nameTableDataMarttableNumber*)
5: *sizeTable* ← *sizeTable*(*currentTable*)
6: **if** *limitCapacityDataMart*(*sizeTable*) **then**
etlDataMartToBackup(*currentTable*)
truncate(*currentTable*)
7: **end if**
8: **end for**
9: **for** *tableNumber* = 1, 2, . . . , *count_table* **do**
10: currentTable ← nameTableDataMart(tableNumber)
11: *businessTable* ← *businessTable*(*currentTable*)
12: *selectTable* ← *selectTable*(*businessTable*)
13: etlWarehouseToDataMart(selectTable, businessTable)
14: **end for**

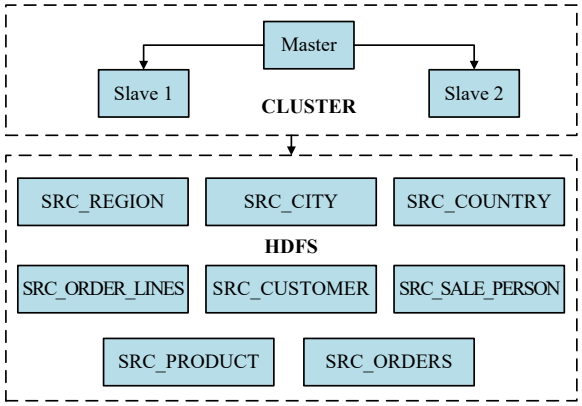


Fig. 5. Model that stores data in Data Lake.

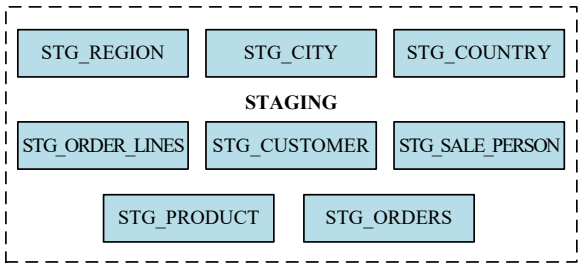


Fig. 6. Data storage tables in the staging area.

When we design the model storage in the Data Warehouse, we need to discuss the Data Warehouse Schema which expresses the logical content of the materialized views constituting the Data Warehouse, is provided in terms of a set of relations. Similarly to the case of the sources, each relation of the data warehouse schema is described in terms of a query over the conceptual model.

From a technical point of view, such queries are unions of conjunctive queries. More precisely, a query q over the Conceptual Model has the form:

$$T(\vec{x}) \leftarrow q(\vec{x}, \vec{y}_m) \quad (1)$$

where the head $T(\vec{x})$ defines the schema of the relation in terms of a name T, and its arity, i.e. the number ofcolumns (number of components of $\vec{x}$, and the body $q(\vec{x}, \vec{y})$ describes the content of the relation in terms of the Conceptual Model. The body has the form

$$\text{con}_1\ (\vec{x}, \vec{y}_1)\ \text{OR ... OR con}\ j_m(\vec{x}, \vec{y}_m) \quad (2)$$

where each $\text{con}j_i(\vec{x}, \vec{y}_i)$ is a conjunction of atoms, and $\vec{x}, \vec{y}_C$ are all the variables appearing in the conjunct (we use $\vec{X}$ to denote a tuple of variables $x_1, \ldots, x_n$, for some n). Each atom is of the form $E(t), R(\vec{t})$, or $A(t, t')$, where $\vec{t}, t$, and t are variables in $\vec{x}, \vec{y}$, or constants, and E, R, and A are respectively entities, relationships, and attributes appearing in the Conceptual Model [12].

After that, we will also consider queries whose body may contain special predicates that do not appear in the conceptual model. The semantics of the queries are as follows. Given a database that satisfies the conceptual model, a query will be generated by

$$T(\vec{x}) \leftarrow \text{con}_1\ (\vec{x}, \vec{y}_1) OR \cdots OR\ \text{con}\ j_m(\vec{x}, \vec{y}_m) \quad (3)$$

of arity n is interpreted as the set of n-tuples $(d_1, \ldots, d_n)$, with each d_i an object of the database, such that, when substituting each d_i for x_i, the formula

$$\exists \vec{y}_1 \cdot \text{con}_1\ (\vec{x}, \vec{y}_1)\ \text{OR} \cdots \text{OR}\ \exists \vec{y}_m \cdot \text{con}\ j_m(\vec{x}, \vec{y}_m) \quad (4)$$

evaluated with true [12]. Suitable inference techniques allow for carrying out the following reasoning services on queries by taking into account the Conceptual Mode [19]:

- *Query containment*. Given two relational queries q_1 and q_2 (of the same arity n) in the conceptual model, we say that q_1 is contained in q_2, if the set of tuples denoted by q_1 is contained in the set of tuples denoted by q_2 in every database satisfying the conceptual model;
- *Query consistency*. A relational query q on the conceptual model is consistent if there exists a database that satisfies the conceptual model in which the set of tuples denoted by q is not empty;

- *Query disjointness*. Two relational queries q_1 and q_2 (of the same order) in the conceptual model are disjoint if the intersection of the set of tuples denoted by q_1 and the set of tuples denoted by q_2 is empty, in every database satisfying the conceptual model.

Based on the theory of building a data warehouse, Eq. (1), Eq. (2), Eq. (3), Eq. (4), and comparing it with the source data set, we chose the Star Schema model to build the Data Warehouse so that it is suitable for

exploiting data, which is also the fundamental data warehouse, it can be able to exploit more in-depth data. The storage model of Data Warehouse is shown in Fig. 7.

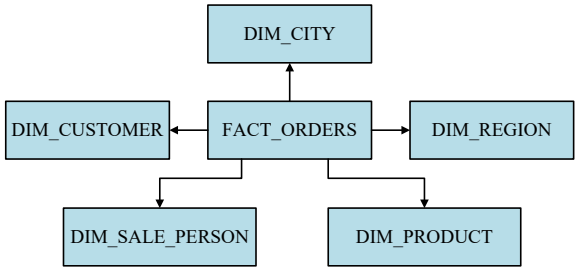


Fig. 7. Data storage tables on the Staging area.

4. Implementation

First, the local data will be put into the Data Lake to restructure the data. After loading the data files and normalizing, the Data Lake will contain the source data files as shown in Fig. 8. After restructuring, semi-structured can be queried on the Data Lake as shown in Fig. 9.

If the data has been stored in Data Lake, we conduct ETL for these data to store in the temporary area (Staging) on the Server Database before calculating the logic to put the data into the data warehouse. Before ETL data about Staging, we will create Schema Staging on Oracle Database, then create tables corresponding to the source table to push the data into Staging. The tables are in the list of the proposeding model Similar to creating tables on staging, we create tables on Data Warehouse based on the previous design. Then we will use the Oracle Data Integrator tool to ETL data from the tables for staging to the Data Warehouse. Finally, ETL data from the data warehouse into the Data Mart. This is the place to store data for in-depth data mining for each business segment., for example, the ETL process of a table in Fig. 10. When we do the data flow from data lake to data store, we will use the PowerBI reporting tool to extract data analysis reports from Data Mart. The data Mart tables can be viewed in PowerBI as shown in Fig. 11.

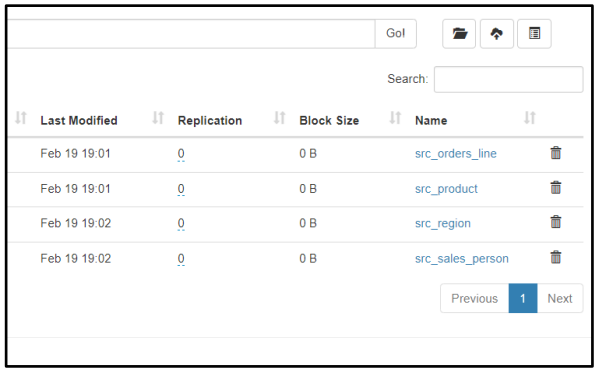


Fig. 8. Directory containing source data files.

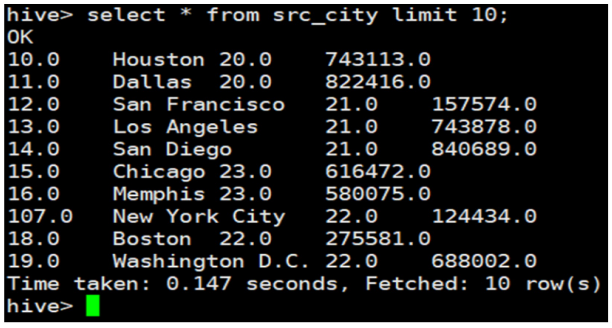
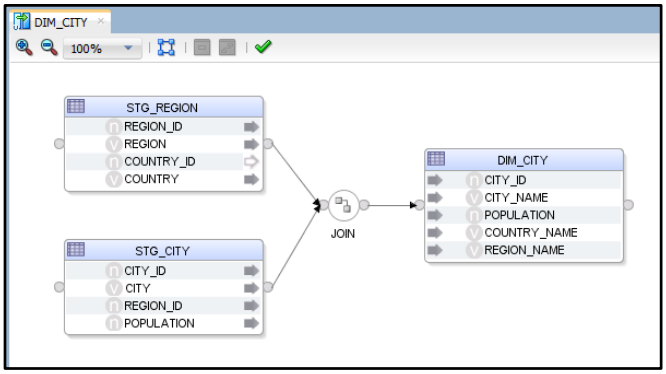
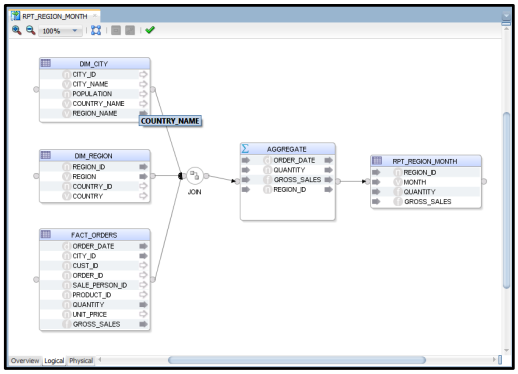


Fig. 9. Query semi-structured data on HDFS.



(a) Put data in table DIM CITY.



(b) Put data in table RPT REGION MONTH.

Fig. 10. Place the data in a table.

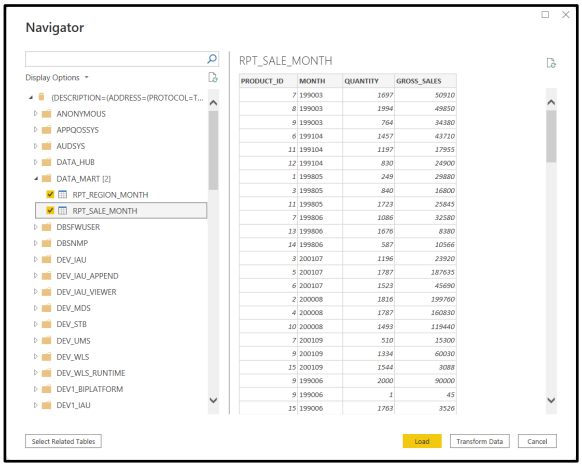


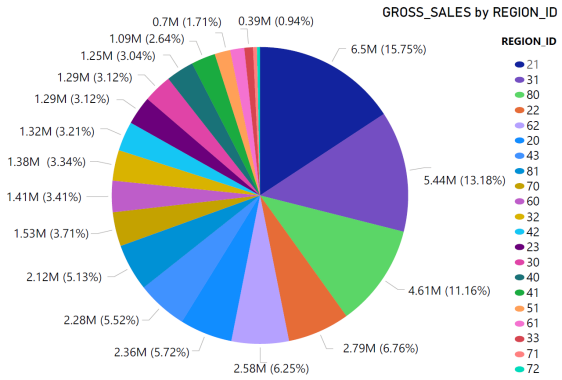
Fig. 11. Data Mart data table displayed in PowerBI.

5. Experimental Results and Analysis

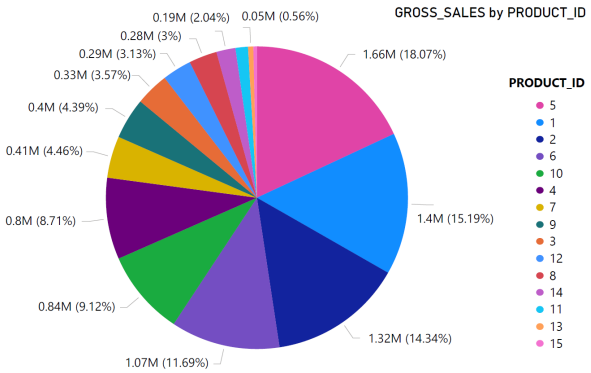
After the data flow has been run to the Data Mart, we proceed to get the reports by business using the Power BI tool to take the report whichgives this trading company a first-hand look at the number of products sold in each region's markets. The company can evaluate potential markets and business areas to develop appropriate product business strategies for those regions or markets. It is shown in Fig. 12

Based on the report, which is in Fig. 12, the company can see the key products for its business. From there, the company can come up with strategies to improve products that do not bring in a lot of revenue, or create more programmes for products that bring in a lot of revenue. Beside that, it shows potential business areas, which generates large revenue. From this, it is possible to make strategies or business decisions that are suitable for each region and market.

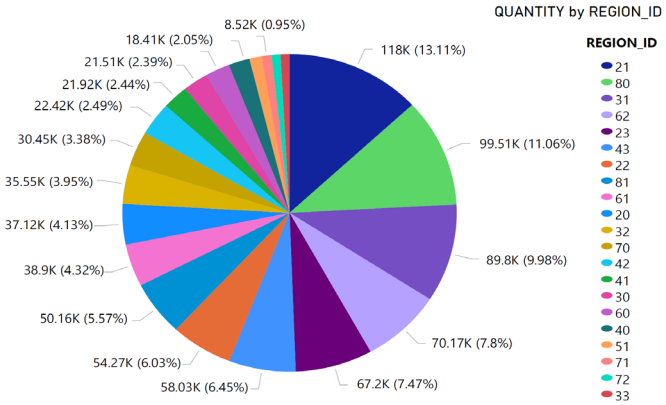
Similarly to Fig. 12, the report of Fig. 13 shows businesses the number of products sold in business months, helping businesses assess at what time of year the number of products sold will be high. Besides that, the company can rely on the columns in the report to assess the period of time in which a month of the year sales will increase, thereby making appropriate programs and strategies for the business months. key business. This is an indepth analysis report that helps this business company to see in which months the product consumption of each region will increase or decrease. This will greatly help the company to be able to distribute the source of the product in the markets or regions at the right time.



a) Revenue report by region

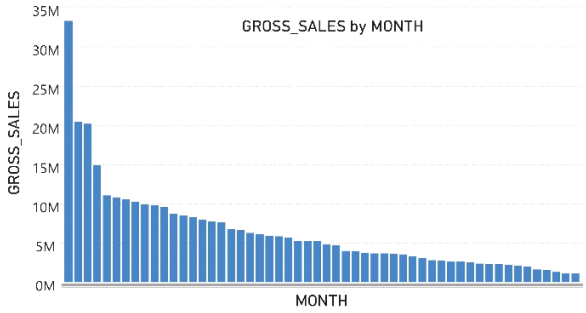


b) Sales report by product

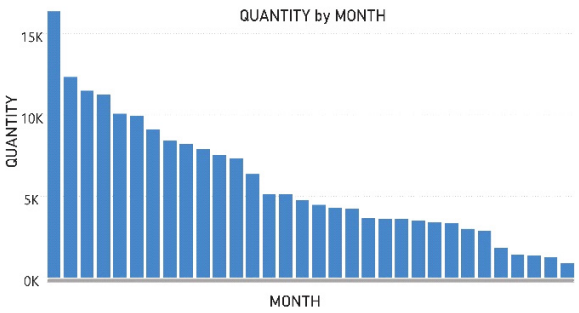


c) Report the numer of products sold by region

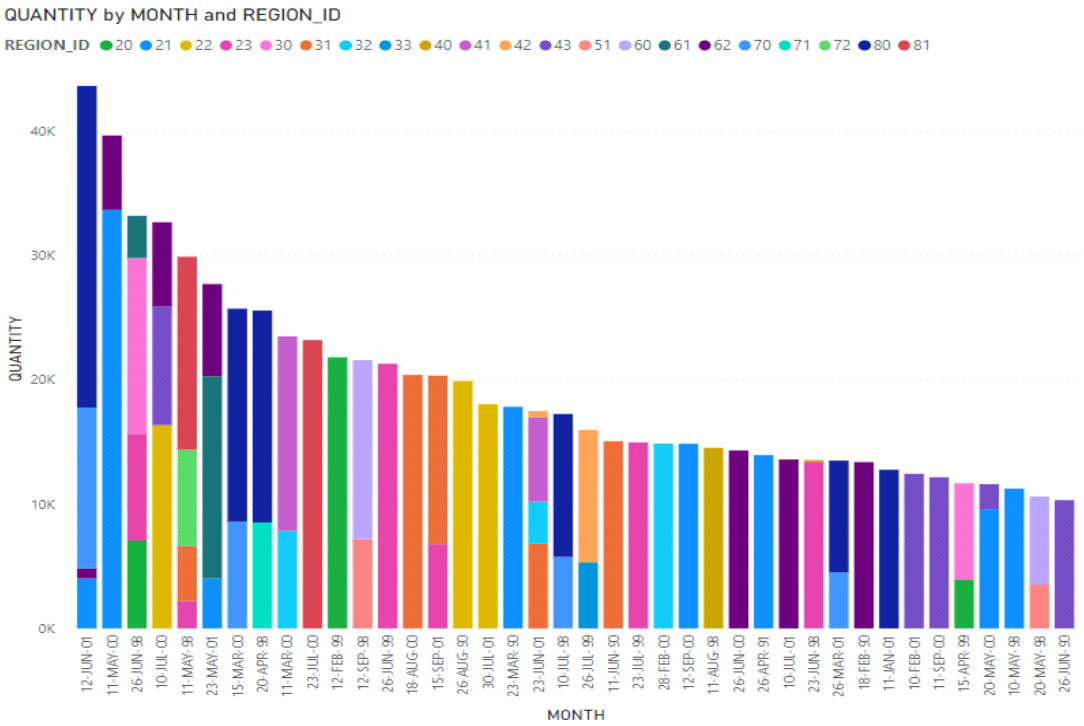
Fig. 12. In-depth report in shape form



(a) Report total revenue by month



(b) Report total revenue by month



(c) Number of accessories consumption by region in each month

Fig. 13. In-depth report in column form

Table 1. Comparison with another model.

Comparative characteristics	**Our Model**	**Model in ref [12]**	**Model in ref [4]**
Variety of Data	all	just data structured	all
Storage capacity	expanded	not expanded	expanded
Difficulty to use	normal	easy	normal
Maintenance	difficult	easy	normal
Fault tolerance	high	low	medium
Customizability	high	low	medium
Mining level	high	medium	high
Processing speed	high	medium	high
Availability	high	medium	high
Resource cost	custom	high	custom
Totalscore	8	6	7.5

6. Comparison with Other Model Data Mining

To evaluate our proposed model more realistically, we will compare our model with other data mining models. Ratings will be given based on the most characteristic technical requirements of a data mining system with the scoring scale is a 10-point scale. The comparison table is shown in Table 1. From Table 1, the model we have proposed will bring many advantages, of course to achieve it we still have to spend money. when there is some disadvantage in the maintenance of the system and in the combination of components.

7. Conclusion

In this paper, we have collected a business data set of an enterprise to run on the built model which we have proposed. From the built model, we can get statistical reports easily, intuitively and accurately in a short time. The results when the model is applied show that the source data sets are organised, stored, and analyzed with many different criteria, and are displayed on the charts in an obvious and detailed way. Furthermore, we also compare the proposed model with some existing models. The results show that the proposed model is easy to use for end-users. It has high scalability and fault tolerance, and a faster processing speed compared to traditional models.

References

[1] M. M. Al-Debei, Data warehouse as a backbone for business intelligence: Issues and challenges, European Journal of Economics, Finance and Administrative Sciences, vol. 33, no. 1, pp. 153-166, 2011.

[2] H. J. Watson and B. H. Wixom, The current state of business intelligence, Computer, vol. 40, no. 9, pp. 96-99, 2007, https://doi.org/10.1109/MC.2007.331

[3] J. Nandimath, E. Banerjee, A. Patil, P. Kakade, S. Vaidya, and D. Chaturvedi, Big data analysis using apache hadoop, in 2013 IEEE 14th International Conference on Information Reuse & Integration (IRI). IEEE, 2013, pp. 700-703, https://doi.org/10.1109/IRI.2013.6642536

[4] L. W. Santoso et al., Data warehouse with big data technology for higher education, Procedia Computer Science, vol. 124, pp. 93-99, 2017, https://doi.org/10.1016/j.procs.2017.12.134

[5] L. W. Santoso, Classifier combination for telegraphese restoration, in 2011 International Conference on Uncertainty Reasoning and Knowledge Engineering, vol. 1. IEEE, 2011, pp. 79-82, https://doi.org/10.1109/URKE.2011.6007844

[6] J. Joe, Data warehouse and big data integration, Int. Journal of Comp. Sci. and Inf. Tech., Vol. 9(2), p.1-17, 2022.

[7] R. Kimball, M. Ross, W. Thorthwaite, B. Becker, and J. Mundy, The data warehouse lifecycle toolkit. John Wiley & Sons, 2008.

[8] C. Todman, Designing a Data Warehouse: Supporting Customer Relationship Management. Prentice Hall PTR, 2000.

[9] W. H. Inmon, Building the Data Warehouse. John wiley & sons, 2005.

[10] M. Golfarelli and S. Rizzi, A survey on temporal data warehousing, IJDWM, vol. 5, pp. 1-17, 01 2009, https://doi.org/10.4018/jdwm.2009010101

[11] P. Ponniah, Data Warehousing: A Comprehensive Guide for It Professional, New York: The McGraw-Hill Companies, 2010.

[12] G. S. Reddy, R. Srinivasu, M. P. C. Rao, and S. R. Rikkula, Data warehousing, data mining, olap and oltp technologies are essential elements to support decision-making process in industries, International Journal on Computer Science and Engineering, vol. 2, no. 9, pp. 2865-2873, 2010.

[13] V. Gour, S. Sarangdevot, G. S. Tanwar, and A. Sharma, Improve performance of extract, transform and load (ETL) in data warehouse, International Journal on Computer Science and Engineering, vol. 2, no. 3, pp. 786-789, 2010, https://doi.org/10.5120/623-887

[14] Daniel Gutierrez, Gartner says beware of the data lake fallacy - insideBIGDATA, insidebigdata.com, https://insidebigdata.com (accessed on 14 March 2019).

[15] C. Campbell, Top five differences between data lakes and data warehouses, White paper: BLUE GRANITE, 2015.

[16] P. P. Khine and Z. S. Wang, Data lake: a new ideology in big data era, in ITM web of conferences, vol. 17. EDP Sciences, 2018, p. 03025, https://doi.org/10.1051/itmconf/20181703025

[17] T. John and P. Misra, Data Lake for Enterprises. Packt Publishing Ltd, 2017.

[18] S. G. Manikandan and S. Ravi, Big data analysis using apache hadoop, in 2014 International Conference on IT Convergence and Security (ICITCS). IEEE, 2014, pp. 1-4, https://doi.org/10.1109/ICITCS.2014.7021746

[19] D. Calvanese, G. De Giacomo, and M. Lenzerini, On the decidability of query containment under constraints, in Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on principles of database systems, 1998, pp. 149-158, https://doi.org/10.1145/275487.275504