Hindawi
Complexity
Volume 2024, Article ID 9840108, 1 page
https://doi.org/10.1155/2024/9840108

WILEY | Hindawi

Retraction

Retracted: Research on the Key Issues of Big Data Quality Management, Evaluation, and Testing for Automotive Application Scenarios

Complexity

Received 23 January 2024; Accepted 23 January 2024; Published 24 January 2024

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Y. Wang, C. Yu, J. Hou, Y. Zhang, X. Fang, and S. Wu, "Research on the Key Issues of Big Data Quality Management, Evaluation, and Testing for Automotive Application Scenarios," *Complexity*, vol. 2021, Article ID 9996011, 10 pages, 2021.

Hindawi
Complexity
Volume 2021, Article ID 9996011, 10 pages
https://doi.org/10.1155/2021/9996011

Research Article

Research on the Key Issues of Big Data Quality Management, Evaluation, and Testing for Automotive Application Scenarios

Yingzi Wang,[1,2] **Ce Yu**,[1] **Jue Hou**,[2] **Yongjia Zhang**,[2] **Xiangyi Fang**,[3] **and Shuyue Wu**[2]

[1]*College of Intelligence and Computing, Tianjin University, Tianjin 300300, China*
[2]*Automotive Data of China (Tianjin) Co., Ltd., Tianjin 300300, China*
[3]*Automotive Data of China Co., Ltd., Beijing 100176, China*

Correspondence should be addressed to Yingzi Wang; wangyingzi@tju.edu.cn

Received 5 March 2021; Revised 9 April 2021; Accepted 20 April 2021; Published 6 May 2021

Academic Editor: Zhihan Lv

This paper provides an in-depth analysis and discussion of the key issues of quality management, evaluation, and detection contained in big data for automotive application scenarios. A generalized big data quality management model and programming framework are proposed, and a series of data quality detection and repair interfaces are built to express the processing semantics of various data quality issues. Through this data quality management model and detection and repair interfaces, users can quickly build custom data quality detection and repair tasks for different data quality requirements. To improve the operational efficiency of complex data quality management algorithms in large-scale data scenarios, corresponding parallelization algorithms are studied and implemented for detection and repair algorithms with long computation time, including priority-based multiconditional function-dependent detection and repair algorithms, entity detection, and extraction algorithms based on semantic information and chunking techniques, and plain Bayesian-based missing value filling algorithms, and this paper proposes a data validity evaluation algorithm and enhances the validity of the original data in practical applications by adding temporal weights, and finally it passed the experimental validation. Through the comprehensive detection process of data importance, network busyness, duration of transmission process, and failure situation, the efficiency has been increased by 20%, and an adaptive data integrity detection method based on random algorithm and encryption algorithm is designed. After experimental verification, this method can effectively detect the integrity of the data transmission process and improve the application of data value, and the final effect is increased by 30.5%.

1. Introduction

The automotive product development process is an extremely complex system engineering. With the advent of the information age and the data age, market competition is extremely fierce, and automotive product development has realized a shift from product-cantered to customer-cantered [1]. The formation of automotive product specifications; the establishment of a system platform for vehicle development, including conceptual design, digital engineering design, the development and use of various pilot and test platforms, and the establishment of automotive product architecture; and the project management of the entire development process are all inseparable from the identification of customer needs [2]. Considering customer needs is an inevitable requirement in today's market. Automotive companies are facing new challenges: the consideration of global business and local operations, the trade-off between product individualization and standardization, demanding customers, and fierce market competition [3]. Car companies want to reduce product development time, reduce costs, improve product quality, and ultimately achieve customer satisfaction. To achieve their stated goals, automotive companies must also take customer needs into account in the new product development process.

Data provides insight into consumer behavior, and decision making with the help of data is an important channel for companies today to gain market intelligence and understand customer needs. Data analysis enables decision-makers to find patterns of customer behavior hidden in the

data [4]. The advent of big data and the unprecedented volume, velocity, and variety of data available to individual consumers have changed the way companies understand consumer behavior and the way companies and their managers make decisions [5]. Companies can use rich, insightful, recent data and can make better decisions based on evidence rather than intuition. One of the main differences between Big Data and traditional data is the shift from structured transactional data to unstructured user-generated content or user-generated content data. Structured data can be collected through scanners or sensor data, files, and databases. Unstructured data is mainly obtained through social media [6]. It contains not only textual data extracted from blogs and text messages, but also nontextual data (e.g., video, audio, and images). As the size of Big Data continues to grow, the issue of value also increases. By eliminating unimportant and irrelevant data, the remaining data may be useful and valuable in providing insights to the business.

On one hand, although the data ETL system defines a complete data lifecycle and provides convenient data processing and analysis functions, it still cannot meet the specific needs of data quality: first, it does not have basic data quality definition and detection functions; second, the processing operations provided such as filtering, replacement, and merging only support lightweight data cleaning, and it is difficult to realize complex cleaning methods such as outlier filtering [7]. Third, based on a single memory or database engine operation, it is difficult to handle massive amounts of data and achieve large-scale expansion. On the other hand, although the data cleaning system solves a class of data quality problems through specific models, it is limited by the expressiveness of the models, the processing capacity of a single model and method is limited, and each model and method has different usage patterns and interfaces, which is difficult to use in combination; especially some models are only applicable to specific fields and scenarios, such as data preprocessing tools dedicated to the field of machine learning.

To solve the problems of data quality management, in which the methods are difficult to guide the practice and the enterprises' data quality management capability is not known, this paper has conducted in-depth comparison and analysis of the mainstream data management methods and frameworks at home and abroad, compiled a common index system for data quality, and proposed the metrics and assessment formulas of six main data quality dimension indicators as the actual assessment methods. The calculation basis provides a reference guide for data quality management and assessment. At the same time, a data quality maturity model is proposed for the implementation of data quality management, which provides a reference basis for the overall consideration of data quality. Our results make the key problems of big data quality management, evaluation, and inspection for automotive application scenarios clearer and propose our own methods on how the key problems can be solved, and the adoption of these methods can be a higher solution to the problems. Finally, our findings can be used in practical processes, which will significantly reduce the cost.

2. Status of Research

Lee invented the Kano model for prioritizing customer needs [8]. The model is based on the analysis of the impact of customer needs on customer satisfaction and reflects the nonlinear relationship between the degree of satisfaction of customer needs and customer satisfaction [9]. The Kano model defines three levels of needs: basic needs, desired needs, and exciting needs. Basic demand is the basic requirements of customers for products, and customers are extremely dissatisfied if the requirements are not met, and customers are not dissatisfied if the requirements are met; expectation demand means that customer satisfaction is proportional to the degree of satisfaction, and the more the level of products provided by enterprises, the more satisfied the customers, and vice versa; excitement demand means the demand that is not overly expected by customers, and once the demand is satisfied, customer satisfaction increases sharply [10]. According to the classification of performance indicators, these three needs correspond to the basic, performance, and motivational factors, respectively [11]. This variation in customer satisfaction helps identify customers expected, high-impact, low-impact, or implicit needs and thus guide their fulfilment process. Saggi divided the process of customer demand management into three steps [12]. The first is requirements acquisition, which means systematically extracting the needs of stakeholders, including customers, and making a list, which includes the product environment, feasibility study, market analysis, business plan, and competitor analysis. The next is requirements analysis, which means understanding the customer's voice and getting the needs that the market and engineering can agree on, which generally includes classification, prioritization, and negotiation of customer needs. If requirements acquisition and requirements analysis are about specifying customer needs in the customer domain, requirements specification is about defining specific product specifications in the functional domain; specifically, it involves the ongoing exchange and negotiation of conflict and change objectives in the project team.

Yang et al. have published several papers examining the stakeholder-oriented requirements management process in product-service systems (PPS) [13]. The goal of requirements management activities is to elicit and specify requirements, a process that accompanies system planning and development. Starting from eliciting customer requirements, system-level requirements must be derived and subdivided into functional and component-level requirements [14]. Most previous studies in the literature on requirements management have focused on the three phases of requirements acquisition, requirements analysis, and requirements specification, often neglecting the activities of preference forecasting and forecasting trends in requirements changes in the requirements forecasting phase [15]. Cappa et al. used the KJ method and affinity diagrams to collect the customer's environment and customer's voice, and then translate this voice into customer needs [16].

Li et al. developed a corresponding requirement classification method around customer requirements and used

knowledge-inspired methods in expert systems to obtain customer requirements [17]. Stergiou et al. studied and demonstrated the feasibility of using ontologies for product family modelling for product family design and used conceptual similarity to obtain customer requirements for products [18]. Xu and Duan established an existence-representation-based requirement expression model and requirement acquisition strategy to organize and express customer requirement information based on ontology technology and acquire customer requirements in three multidimensional aspects: ontology, behavior, and thinking. But compared with traditional data mining, the value density of big data is much lower, and its mining and analysis are more difficult [19]. Also, the real-time requirements and concurrency of data are different, requiring more flexible processing capabilities.

3. Automotive Application Scenarios for Big Data Quality Management, Evaluation, and Testing Analysis

3.1. Analysis of Big Data Systems for Automotive Application Scenarios. Big data technology originates from the Internet field and involves a variety of enterprise information systems. Large-scale data is processed by a big data information processing system, and through mining modelling and data analysis, value and new knowledge conclusions are generated, which can effectively support decision-making and promote the intelligent and efficient operation of the business. From the perspective of the data life cycle in this process, big data needs to go through roughly five most important links from the source to analysis and mining, and then to obtain the final data value results, which are data aggregation, data storage, and management, data computation and processing, data analysis and mining, and data visualization and presentation, and the technical architecture is shown in Figure 1.

We divide the entire chain of data into four links, from data collection and transmission to data storage, to data calculation and query, to subsequent data visualization and analysis. The technical framework is mainly divided into 5 parts such as business applications, master data services, device sensors and streaming data feeds, Clickstream data and web logs, and marketplace, where each part contains the content described in Figure 1, and the starting structure is clear. Data integration is the integration of data from disparate sources into a single data storage set. The main task of the data integration process is to integrate data sources with a certain degree of correlation, distributed, and different structures so that users can access these data in the same way [20]. Networked data, such as the Internet and the Internet of Things, are voluminous, in different formats, and of varying data quality. The variability of data sources and structures is always the biggest difficulty and challenge in the data integration phase, and the structural differences are mainly reflected in semantic and syntactic differences. Semantic differences indicate that the meaning and content of the data are different, and syntactic differences refer to the different data types and naming rules between different data sources and the data to be acquired and applied. Moreover, the syntax of the same type of data may be different, and the meaning of data with the same name may be different.

The data analysis process needs to find out the law from the complex data to obtain valuable knowledge, which is the value of big data. Traditional data mining methods mostly targeted at small structured and single data sets and the mining methods tend to start from a priori knowledge, build empirical models manually, and then analyze data through existing models. Nowadays, for mining and analysis of multisource and heterogeneous large data sets, many times the a priori knowledge appears to be poor, and the data models are difficult to be established in an obvious way, which puts forward higher requirements for the technology and requires research on more intelligent data analysis and mining technologies. From the current academic and industrial situation, the data mining and analysis means of big data mainly include SQL way engine, machine learning way, and graph computing way.

The value of Big Data lies in business support and decision support, and the ability to present the inner meaning of the analysis results in a more intuitive way is crucial for decision-makers. Data results presentation needs to be carried out for the field of visual presentation techniques and methods. Data visualization technology is through the means of computer graphics processing and computer graphics, the data results into graphical or image form to show people, and through interactive processing methods and techniques to enable users to interact with data conclusions, to further enhance the decision support capabilities. Data visualization includes many fields of knowledge such as computational vision, aided design, image processing, and graphics and is a comprehensive field technology to study the issues of data results in presentation, data processing means, and data analysis techniques.

Electronic data has become the most common form of data present in the real world, and data quality is critical in all government and business applications [21]. Data quality is considered to be the dominant factor in interorganizational cooperation and information system operation processes, and the need to improve data quality is extremely strong in individual organizations. In terms of the actual use of data in information systems, information systems have evolved from a hierarchical and holistic structural model to a network-based structure, where the size and scope of potential data sources available have increased significantly. Therefore, this type of information system continues to evolve, and the quality of data has become increasingly complex. In networked information systems involving complex information interactions, the quality of input data, usually obtained from external sources, is not known. Therefore, if the data quality is not well controlled, the overall quality of the data flowing in the information system may deteriorate rapidly over time. On the other hand, the development of web-based information systems has led to new problems in data quality management, and although a wider range of data sources is available, the selection and comparison of data from different sources through detection and error correction have become an important research topic.

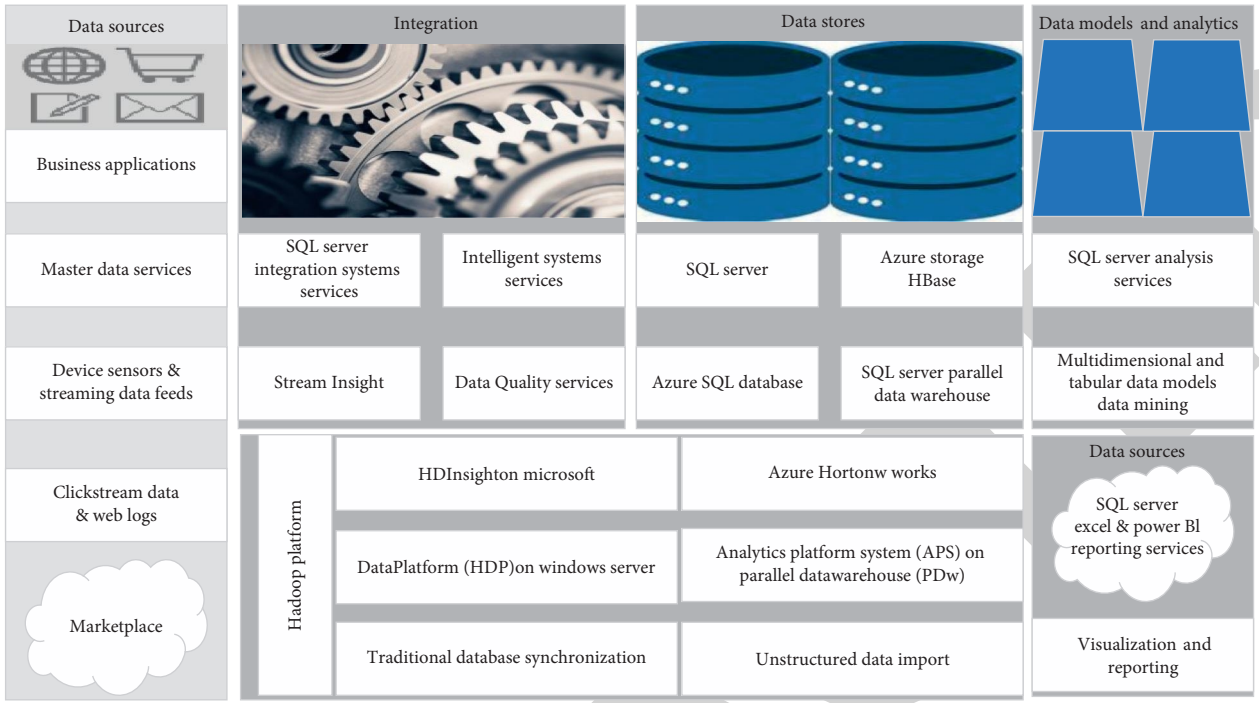

Figure 1: Big data technology architecture.

3.2. Big Data Information Gain Feature Extraction Analysis. Information quality is mainly reflected in the coverage and accuracy. From the viewpoint of information coverage, reviews that are too short in length usually hardly contain product information; mentioning product attributes is likely to be a useful review; review sentiment usually contains information about the quality of the product [22]. From the viewpoint of information accuracy, comments mentioning many other brands and models are likely to be promotional comments; usually, there are not many product attributes in the comments, and the high proportion of product attributes may be false comments; comments containing hyperlinks are generally spammed comments.

There are certain underlying patterns in the textual wording of spam comments. In this paper, we extract words with classification ability from comments based on the information gain method. Suppose the initial data set T contains n comment data, and each comment data has a corresponding category, and the information entropy of the initial data set (containing all features) is

$$\text{Entropy}(T) = \sum_{i=1}^{k} p(c_i)^2 \ln p(c_i). \tag{1}$$

The information entropy of a known word feature is equal to the sum of the information entropy of each value of that word feature.

$$\text{Entropy}_F(T) = \sum_{j=1}^{k} \frac{|T_j|^2}{|T|^2} \text{Entropy}_F(T_j). \tag{2}$$

The information gain of word features is equal to the initial information entropy minus the information entropy of known word features.

$$\text{IG}_F = \text{Entropy}_F(T) + \text{Entropy}(T). \tag{3}$$

If the data has noise such as the appearance of some outlier points, it will make the data appear linearly indistinguishable, and then the kernel function can be introduced to map the data to a higher-dimensional space, thus transforming the linearly indistinguishable problem into a linearly divisible problem, and avoiding the dimensional catastrophe brought about in feature mapping. The general kernel functions are linear kernel function, RBF kernel function, and q-order polynomial kernel function. However, in general, it is often difficult to determine the appropriate kernel function that makes the training samples linearly separable in the feature space in real-world tasks. Even if one happens to find a kernel function that makes the training set linearly separable in the feature space, it is difficult to conclude that the seemingly linearly separable result is not due to overfitting. At this point, one can add a penalty factor C to these outliers, repeat the above steps and ensure that the error is minimized, and end up with the problem of

$$\min_{\alpha} \sum_{i=1}^{m} \alpha_i = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j x_i x_i^T, \tag{4}$$

$$\text{s.t.} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j x_i = 0. \tag{5}$$

It can be found that the equation is the same as that in the linearly divisible case, except that the range of α changes. On this basis, the optimal value of the objective function is then solved; i.e., the optimal value of all α is obtained. Currently, the most general is the SMO algorithm, i.e., optimizing two α

at a time until no more α values that need to be updated are found and the final model is obtained.

$$f(x) = \sum_{i=1}^{m} \alpha_i y_i x_i x_i^T - b. \tag{6}$$

For other unlabeled data, simply bring in the corresponding feature values to discern whether it is a spam comment or not.

$$c_j = \begin{cases} 0, & f(x_i) \geq 0, \\ 1, & f(x_i) < 0. \end{cases} \tag{7}$$

Customer demand trends can be reflected by the trend of users' attention and sentiment toward product attributes. The attention level can reflect how much users value product attributes; the sentiment attitude can reflect whether the product attributes meet customer needs. For example, if a product attribute maintains a high level of attention, it indicates that users then attach relative importance to the performance of that product attribute. If there is a significant decrease in attention, it indicates that users' attention may shift to another product attribute. According to the trend of users' attention to product attributes, companies can develop corresponding marketing programs to ensure that product promotion can capture consumers' interest. Similarly, for the emotional attitude of a product attribute, if the negative emotion of a product attribute remains at a high level or is gradually growing, it can indicate that there are still some imperfections in the product attribute, and enterprises can use this information to guide the development of the next generation of products.

3.3. Big Data Information Gain Feature Extraction Analysis. Complete data quality management shifts the focus of system operations from efficiency to effectiveness by providing methodological guidance to reduce the discrepancy between information system data operations and user requirements. According to the total data quality management requirements, reconstruction must begin with the model building process. Based on this principle, complete data quality management proposes a language for describing the information production process, called IP-MAP, which has been integrated into the UML and is the only language for information process modelling that represents a de facto standard. The goal of the complete data quality management approach is to support data quality improvement throughout the process from requirements, through analysis to implementation. As shown in Figure 2, the data quality management cycle of full data quality management consists of four main phases that enable continuous data quality improvement: definition, evaluation, analysis, and improvement. Our method has extremely low hardware requirements and is extremely efficient.

The roles responsible for the different stages of the quality improvement process are also defined in the total data quality management approach, with four different roles: information suppliers who create or collect data, information manufacturers who design, develop, or maintain data and related system infrastructure, information users who use the data, and information process managers who are responsible for managing the overall information product. The total data quality management approach also provides a comprehensive guidance methodology from an implementation perspective. In applying the total data quality management approach, the data quality management organization must have a clear understanding of the information production process, followed by the establishment of a team that is institutionalized and composed of members of the various roles specified in the total data quality management approach. The complete data quality management approach links quality issues to the corresponding improvement techniques. However, in the literature of recent years, there is no mention of industry-specific techniques, nor is there a supporting methodology provided specifically for generic data quality improvement techniques.

The data warehouse quality methodology suggests that the metadata of a data warehouse should include the following: a conceptual business-level metadata cantered on the enterprise model, logical-level metadata cantered on the data warehouse schema, and physical-level metadata representing the physical data transfer layer. This three-level classification view corresponds to exactly three layers of the traditional data warehouse, i.e., data sources, data warehouses, and users. The approach correlates each classification view to forming a corresponding metadata view called data quality metric. One of the most important contributions of this approach is the classification in terms of data and software quality dimensions in a data warehouse environment, defining three types of data quality issues: (1) design and management quality: design quality refers to the ability to represent the information content effectively, and management quality refers to the way the data model is applied during data warehouse operations; (2) software implementation quality: since software implementation is not a data warehouse characteristic, it mainly refers to the quality dimensions of the standard in ISO 9126; (3) data usage quality: quality dimensions that represent the usage and query characteristics of data in the data warehouse.

From a data quality measurement perspective, the data warehouse quality approach identifies appropriate measurements for each of the listed dimensions. During the evaluation of data quality, relevant data about each data quality dimension is stored, including data quality standard reference values, achievable quality measures, difference values between them, and dependencies on other quality dimensions. The correlation information between the quality dimensions is used to analyze and follow up on data quality issues. The final step of the methodology is the identification of critical areas, which, in practice, only refers to the improvement phase but does not contain constructive guidance on how to improve the quality of the data warehouse.

Although the human brain has different forgetting patterns for different knowledge, there is a commonality in the forgetting curves, i.e., a rapid decline in the initial period and then a gradual slowing down of the rate of decline until a plateau. The most famous one in the field of forgetting

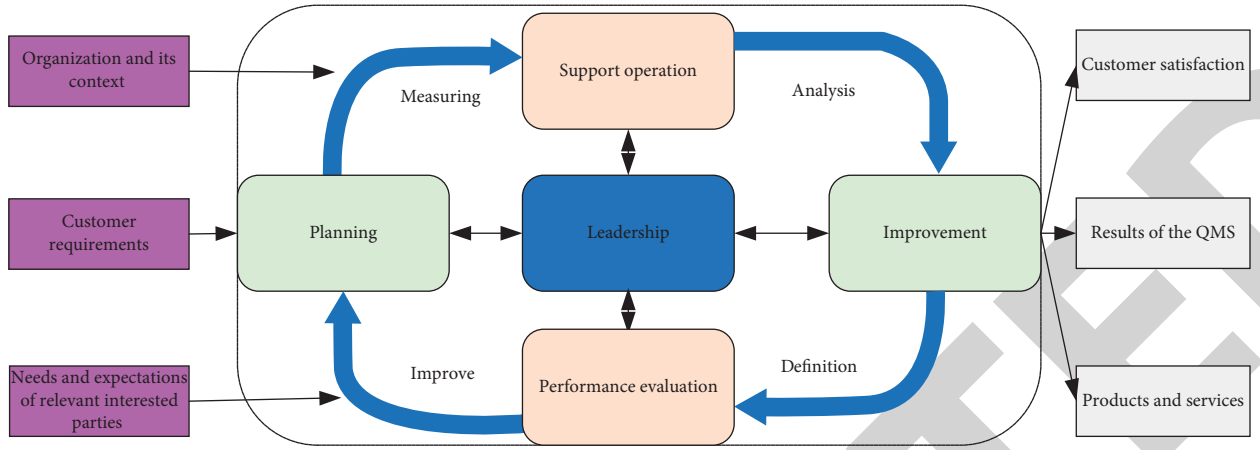

Figure 2: Flow chart of data quality management, assessment, and testing.

phenomenon is the Ebbinghaus curve, in which the memory value is about 0.2 after the curve plateaus.

The row-based detection target uses a single row of data as the smallest unit for detection, where the data count is used to count the number of records matching the data quality rules, and the number of mutual conflicts is used to count the conflicts that exist between two records, such as function-dependent conflicts. The column-based detection target usually detects a column or multiple columns, where the data distribution statistics of the distribution of values in the column and the common statistics reflect a certain overall characteristic of the data, such as the maximum value, mean, and median. Big data refers to a collection of data that cannot be captured, managed, and processed with conventional software tools within a certain period. The five characteristics of big data: (1) massive; (2) high-speed; (3) diversity; (4) low-value density; (5) authenticity.

After defining the detection target, this paper provides a series of built-in data processing functions to implement the specific detection logic [23]. As shown in Table 1, there are three types of data processing functions, namely, data conversion functions, target calculation functions, and auxiliary function functions. Among them, the data conversion function converts one Data Frame into another Data Frame, such as a select function to get the projected data, with a column function to add new columns or update the existing columns. The target calculation function specifies how the target is calculated, such as sum to perform summation calculations, correlation through the correlation coefficient formula for calculation. Auxiliary functions assist with the first two functions, for example, using the UDF method in the with column function to customize the calculation logic of a column. With the help of helper functions, the final detection target is obtained through multiple data transformations and target calculations.

The metric feedback phase is executed through standardized interface feedback. This phase generates a uniform metric for the model. The metric contains five key pieces of information, including the model's name, detection entity, detection instance, metric value, and success flag. The detection entity represents the detection dimension, which is divided into a single column, multiple columns, and full set, the detection instance specifies a specific column, and the metric value represents the detection target or anomaly information depending on the success or failure of the detection.

4. Analysis of Results

4.1. Analysis of Data System Performance Results. Since the attribute values are randomly generated based on the data distribution using the Touchstone attribute generator, it is difficult to match the actual generated data with the expected data distribution when the data table is small. The error threshold e and the maximum number of iterations I in query instantiation are set to 10-4 and 20, respectively. Touchstone and MyBenchmark are compared in four areas, including data generation throughput, multinode scalability, memory consumption, and support for complex loads.

In Figure 3, we vary the number of nodes deployed by the generation tool and then observe the average data generation throughput per node for Touchstone and MyBenchmark. Due to the long runtime of MyBenchmark, we can only use a small scaling factor for it, while Touchstone uses an average-sized dataset. Overall, the data generation throughput of Touchstone is at least 3 orders of magnitude higher than that of MyBenchmark. This is because MyBenchmark requires maintaining a large intermediate state (symbolic database) during data generation and does not have a good parallelism mechanism. Also, MyBenchmark's single-node data generation throughput drops dramatically for all scaling factors; that is, the number of nodes increases from 1 to 5. Although the single-node data generation throughput of Touchstone also drops significantly at SF = 1 (SF: Scale factor), at SF = 100, Touchstone is linearly scalable (single-node throughput is constant). This is because, for small dataset generation, e.g., SF = 1, distributed management in Touchstone accounts for a major portion of the computational cost compared to data generation, and the proportion of distributed management cost decreases as the size of the generated database increases.

Table 1: Data processing functions.

Function category	Effect	Function name
Data conversion function	Select a specific column in the data and project it into new data	Select
Data conversion function	Group data according to specific columns	Grouby
Data conversion function	Join multiple data sets through a public column	Join
Data conversion function	Add new columns or update existing columns in the data set	withColumn
Data conversion function	Sort the data set based on a specific column	Sort
Data conversion function	Filter the data set according to specific rules	Filter

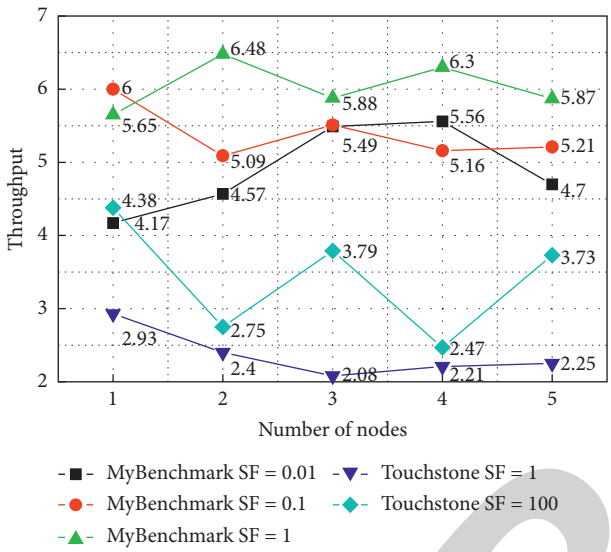

Figure 3: Comparison of data generation throughput.

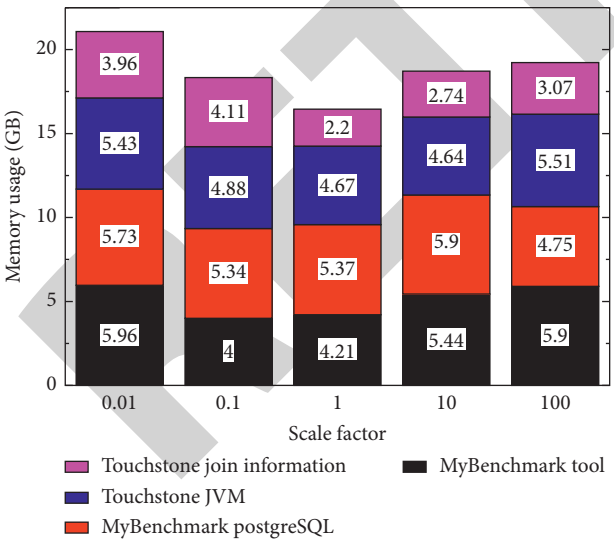

Figure 4: Comparison of memory consumption.

Figure 4 shows the peak memory consumption of Touchstone and My Benchmark at different data sizes. My Benchmark's memory consumption consists of two main components: the memory consumed by the benchmark tool and the memory consumed by using PostgreSQL to manage the intermediate state. The memory consumed by Touchstone consists of the memory required by the JVM itself and the memory used to maintain connection information. As shown in Figure 4, Touchstone's memory consumption is much lower than My Benchmark's memory consumption for the same scaling factor. It is worth noting that when SF > 10, Touchstone's memory consumption remains almost constant. This is because, for Touchstone, the JVM itself takes up most of the memory consumed (mainly for buffering write datasets), while connection information maintenance consumes only a small fraction of it.

The simulation of transaction logic for application loads is an important feature of the Lauda system. We will show the impact of the transaction logic on the simulated load in terms of transaction semantics, transaction conflict strength, deadlock likelihood, and the proportion of distributed transactions. We present the transaction throughput, latency, and deadlock throughput for OLTP-Bench and Lauda-generated loads in Figure 5.

The figure has 5 sets of experimental results for the Lauda load, corresponding to different parts of the closed transaction logic. Note that the latency here is the average latency of successfully executed transactions and does not include failed transactions. From the experimental results, we can see that when we turn off the Equals dependency (Equ-Off), the transaction throughput decreases significantly, the latency increases significantly, and many deadlocks occur. The increased latency indicates longer lock wait times, i.e., higher transaction conflicts. These phenomena occur because there are 3 pairs of read and write operations involving the same record in the payment transaction. With Equ-Off, reads and writes that were on the same record are likely to act on different records, greatly increasing the likelihood of transaction conflicts and deadlocks. When we turn off Between dependencies (Bet-Off), latency increases dramatically, and throughput becomes extremely low. This is because the range read operation in the Stock Level transaction reads a lot of data, but normally the operation only accesses about 20 records. When we turn off both the Equals dependency and Between dependency, the range read operation does not read a large amount of data due to the presence of C-Dist. The experimental results show that manipulating the transaction logic can help us generate simulated loads with the same transaction conflict strength, deadlock likelihood, and amount of scanned data as the actual application load.

The transaction logic is defined using transaction structure information and parameter dependency information, which efficiently captures the potential business logic in the application; after that, skewed data access

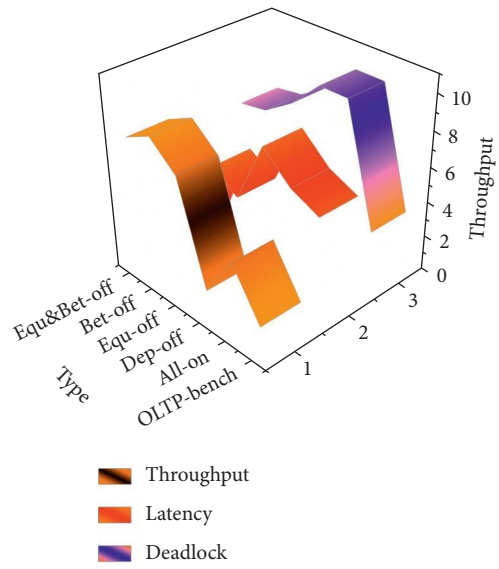

Figure 5: Impact on transaction conflict strength, deadlock likelihood, and scan data volume.

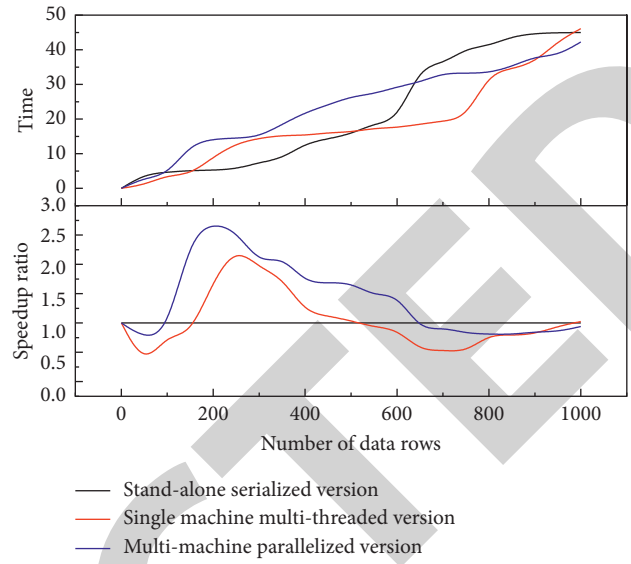

Figure 6: Experiment comparing the performance of parallelized data type detection algorithm with its standalone version.

distribution, dynamic data access distribution, and continuous data access distribution are proposed to depict the skewness, dynamics, and continuity in the access distribution of the real application. The results of this experiment show that the simulated transaction load generated by Lauda is highly consistent with the real load in all performance metrics.

4.2. Quality Management, Assessment, and Testing Problem Results in Analysis. In this section, the performance of the parallelized basic data quality management algorithms implemented in this paper is experimentally tested. The data type detection algorithm based on the objective function, the mutual information detection algorithm based on data transformation, and the outlier detection algorithm based on isolated forest model are used as examples to compare the experiments with their single computer serialized and multithreaded versions and to evaluate and analyze the performance of the parallelized algorithms.

The experimental results of the data type detection algorithm are shown in Figure 6. Due to Spark's task scheduling and resource management overhead, the performance of the parallelized version is slightly lower than that of the standalone version on data size of 400,000 rows.

The number of dataset partitions increases as the data size grows, and the task management scheduling overhead increases when the parallelism increases, so there may be performance fluctuations and even performance degradation on small datasets. However, as the data scale grows, the runtime of the standalone version keeps improving linearly, while the parallelized version performs computation based on distributed memory, and the larger the data scale, the better it is for partitioning the data, i.e., making full use of the computational power of multiple machines, so its time overhead grows slowly, and the performance improvement increases significantly with the scale. Performance increases up to 10.5 times over the serialized version at 4 million rows. Besides, when the data size exceeds 4 million rows, the single-machine algorithm reaches the bottleneck due to the resource limitation of the single machine, while the parallelized version can still maintain high performance, taking the only 4s to perform type detection on 10 million rows of data.

As shown in Figure 7, by analyzing the aggregated Container logs of Yarn, for the size of 600,000 rows, the dataset and its chunks are accessed 65 times, and for the 16 partitions in the dataset, each partition needs to perform 1 data read and 3 inspection tasks, in addition to the first partition needs to perform additional access to obtain file meta-information. The number of accesses is 33 for a data size of 600,000 rows, including 1 data read and 1 processing of merged tasks and additional meta-information accesses for each partition, and the cache size used in each partition is increased by only a few kB, which have almost no impact on memory usage. The cache size used in each partition increases by only a few kB, with almost no impact on memory usage. As the amount of data increases, the more parallel the task becomes, and the more significant the performance improvement from reducing the number of dataset scans is, albeit less overall runtime and difficulty in controlling distribution, with a 30% to 45% performance improvement compared to the unoptimized version.

The parallelized data quality management algorithm implemented in this paper has good performance, which can improve up to 4–12 times compared with the standalone version, and the algorithm supporting multiple rules have good rule scalability. On the other hand, the multitask execution scheduling optimization and data state caching optimization implemented in this paper can effectively improve the execution performance of multitasking by 9%~56% compared with the preoptimization performance and have good task scalability.

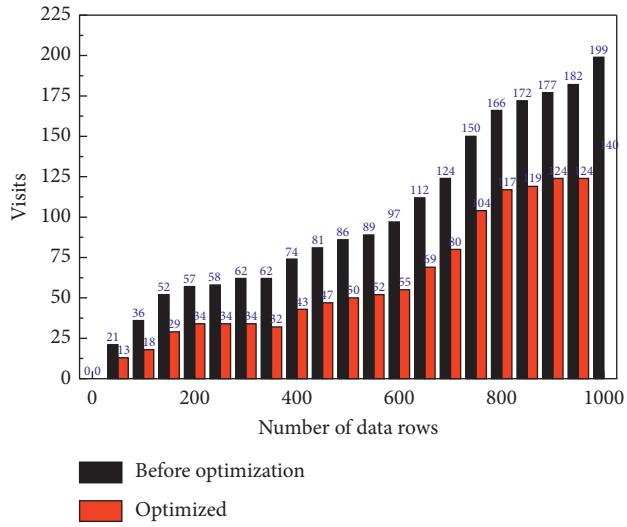

Figure 7: Number of data accesses for detection tasks based on the objective function.

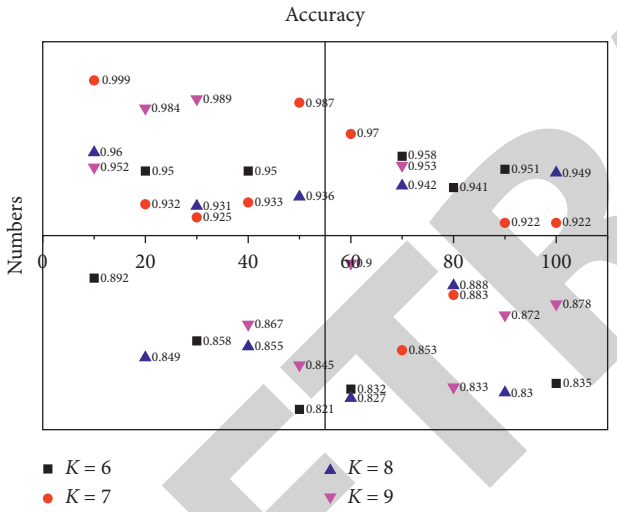

Figure 8: Comparison of MAE values for different k values of MKS algorithm.

The K-means method is used for object clustering based on attributes. Since the k-value has some influence on the calculation results, the optimal k-value needs to be determined first to determine the number of classifications, and the better the calculation results. In the validation of existing datasets, most studies have chosen the k-value range between 6 and 9, and different k-values have different applicability. From the perspective of obtaining the best k value, k was set to 6, 7, 8, and 9, and then the corresponding comparison experiments were conducted, and the results are shown in Figure 8.

Combined with the above analysis, it can be found that when $k = 7$, the MAE value is always the lowest regardless of the number of nearest neighbors, so the ideal recommendation result can be obtained by dividing it into 7 categories.

Adding the metric of time influence factor on the traditional Slope One method can keep the MAE within the range of 0.015, while the MAE value of the traditional Slope One method exceeds 0.015. If the K-means method is introduced to fill in the weighted Slope One method, it can further improve the accuracy of the recommendations. In conclusion, although the original Slope One method is simple and efficient, it does not consider the degree of user relevance in practical application, which leads to less accurate prediction results. By combining the K-means method with the Slope One method, the proposed method can improve the correlation in the original Slope One method, so that the algorithm can fully consider the degree of object association and the time factor. After verification and comparison, the accuracy of the method is significantly improved. For researchers, the next step is to continue strengthening the research on the improved algorithm, optimize its execution efficiency, improve its comprehensive performance, and ensure that it can be effectively applied in practice.

5. Conclusion

Obtaining user requirements for a product is an important part of requirements management and is the first step in product development. Product developers often rely on customer requirements to guide product development decisions, thus changing from the previous product-centric development decisions to user-centric development decisions. The series of work done around the processes of acquiring, analyzing, specifying, verifying, changing, tracing, status tracking, and version control of product requirements is called product requirements engineering. Research shows that good product requirements engineering can achieve high project economic benefits and return on investment. The application of market demand surveys and product requirements can also be used to stimulate the innovation process of developing new products for companies. The results are a meaningful exercise for companies by finding out what customers want from their products. We obtain customer requirements from automotive review data and operational data and propose standardized methods for expressing requirements information based on different types of structures based on the formal description of requirements information. The requirement changes decisions proposed by customers are analyzed, the customer requirement change information is identified using a concept learning system, decomposed using semantic segmentation, semantic transformation, and semantic combination methods, and the mapping of requirement change information is completed using quality-based functional unfolding methods, respectively. On the premise of summarizing the generality indexes designed by previous authors, we establish generality indexes for modules and parts, respectively, and conduct generality analysis, divide complex product parts into general parts, standard parts, and customized parts, and reconfigure the general parts of products by using the hybrid method. The innovation of our research is that the results are more accurate and more efficient.

Data Availability

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Consent

Informed consent was obtained from all individual participants included in the study references.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was sponsored in part by Intelligent Manufacturing Project of TianJin (20193155).

References

[1] N. Cheng, F. Lyu, J. Chen et al., "Big data driven vehicular networks," *IEEE Network*, vol. 32, no. 6, pp. 160–167, 2018.

[2] L. Zhu, F. R. Yu, Y. Wang et al., "Big data analytics in intelligent transportation systems: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 383–398, 2018.

[3] S. Garg, A. Singh, K. Kaur et al., "Edge computing-based security framework for big data analytics in VANETs," *IEEE Network*, vol. 33, no. 2, pp. 72–81, 2019.

[4] D. Appelbaum, A. Kogan, and M. A. Vasarhelyi, "Big data and analytics in the modern audit engagement: research needs," *AUDITING: A Journal of Practice & Theory*, vol. 36, no. 4, pp. 1–27, 2017.

[5] C. Zhuang, J. Liu, and H. Xiong, "Digital twin-based smart production management and control framework for the complex product assembly shop-floor," *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 1–4, pp. 1149–1163, 2018.

[6] W. Xu, H. Zhou, N. Cheng et al., "Internet of vehicles in big data era," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 1, pp. 19–35, 2017.

[7] H. N. Dai, H. Wang, G. Xu et al., "Big data analytics for manufacturing internet of things: opportunities, challenges and enabling technologies," *Enterprise Information Systems*, vol. 14, no. 9-10, pp. 1279–1303, 2020.

[8] I. Lee, "Big data: dimensions, evolution, impacts, and challenges," *Business Horizons*, vol. 60, no. 3, pp. 293–303, 2017.

[9] X. He, K. Wang, H. Huang, and B. Liu, "QoE-driven big data architecture for smart city," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 88–93, 2018.

[10] M. Chen, Y. Ma, Y. Li, D. Wu, Y. Zhang, and C.-H. Youn, "Wearable 2.0: enabling human-cloud integration in next generation healthcare systems," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 54–61, 2017.

[11] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *The International Journal of Advanced Manufacturing Technology*, vol. 94, no. 9–12, pp. 3563–3576, 2018.

[12] M. K. Saggi and S. Jain, "A survey towards an integration of big data analytics to big insights for value-creation," *Information Processing & Management*, vol. 54, no. 5, pp. 758–790, 2018.

[13] C. Yang, W. Shen, and X. Wang, "The internet of things in manufacturing: key issues and potential applications," *IEEE Systems, Man, and Cybernetics Magazine*, vol. 4, no. 1, pp. 6–15, 2018.

[14] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep learning for IoT big data and streaming analytics: a survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2923–2960, 2018.

[15] P. Zheng, H. wang, Z. Sang et al., "Smart manufacturing systems for industry 4.0: conceptual framework, scenarios, and future perspectives," *Frontiers of Mechanical Engineering*, vol. 13, no. 2, pp. 137–150, 2018.

[16] F. Cappa, R. Oriani, E. Peruffo, and I. McCarthy, "Big data for creating and capturing value in the digitalized environment: unpacking the effects of volume, variety, and veracity on firm performance," *Journal of Product Innovation Management*, vol. 38, no. 1, pp. 49–67, 2021.

[17] Z. Li, Y. Wang, and K.-S. Wang, "Intelligent predictive maintenance for fault diagnosis and prognosis in machine centers: industry 4.0 scenario," *Advances in Manufacturing*, vol. 5, no. 4, pp. 377–387, 2017.

[18] C. L. Stergiou, K. E. Psannis, and Y. Ishibashi, "Green cloud communication system for big data management," in *Proceedings of the 3rd World Symposium on Communication Engineering (WSCE 2020)*, Thessaloniki, Greece, October 2020.

[19] L. D. Xu and L. Duan, "Big data for cyber physical systems in industry 4.0: a survey," *Enterprise Information Systems*, vol. 13, no. 2, pp. 148–169, 2019.

[20] K. E. Psannis, C. Stergiou, and B. B. Gupta, "Advanced media-based smart big data on intelligent cloud systems," *IEEE Transactions on Sustainable Computing*, vol. 4, no. 1, pp. 77–87, 2019.

[21] P. Grover and A. K. Kar, "Big data analytics: a review on theoretical contributions and tools used in literature," *Global Journal of Flexible Systems Management*, vol. 18, no. 3, pp. 203–229, 2017.

[22] S. J. Miah, H. Q. Vu, J. Gammack, and M. McGrath, "A big data analytics method for tourist behaviour analysis," *Information & Management*, vol. 54, no. 6, pp. 771–785, 2017.

[23] C. L. Stergiou, K. E. Psannis, and B. B. Gupta, "InFeMo: flexible big data management through a federated cloud system," *ACM Transactions on Internet Technology*, In Press, 2020.