

Prescriptive Analytics on Cloud Based Systems Using Deep Learning Techniques

Srinivas Naveen Dolu-Surabhi^{1,*}, Manikandan Rajagopal²,
M. Hema Kumar³, S Srividhya⁴

¹ General Motors, Michigan,
United States of America

² School of Business and Management,
Lean Operations and Systems, Bangalore,
India

³ Sona College of Technology,
Department of ECE, Salem,
India

⁴ KPR College of Arts Science and Research,
Department of Computer Science, Coimbatore,
India

{srinivas.csii, mani4gift, vidhyasai14}@gmail.com, hemakumar.ece@sonatech.ac.in

Abstract. Today's cloud-based systems, often called Cloud Computing, require robust Prescriptive Analytics to optimize the efficacy of Decision-Making processes. This study addresses the problems facing Prescriptive Analytics by implementing a fresh approach that utilizes Deep Auto Encoder Optimisation. The lack of approaches integrating Cloud Computing - Prescriptive Analytics systems with Deep Autoencoder optimization creates a field of study. Such methods make this an issue. Our strategy leverages Deep Autoencoders to uncover complex patterns and correlations from various datasets, improving the precision and effectiveness of Cloud Computing - Prescriptive Analytics systems. The outcome indicates that the DM procedures were greatly improved because of the substantial quantity of research and validation that has to be performed, which proves that the technique is effective. In addition to resolving the problems with Cloud Computing- Prescriptive Analytics, Decision-Making process optimization paves the way for potential improvements in Artificial Intelligence-powered decision support systems.

Keywords. Cloud-based systems or cloud computing, prescriptive analytics, decision-making, deep autoencoder optimization, artificial intelligence

1 Introduction

The development of Cloud Computing (CC) has resulted in a modern phase of data-driven Decision-Making (DM) [1]. When implemented in cloud settings, standard analytics methodologies encounter challenges due to the complex and dynamic nature of CC [2].

Optimizing the outcomes of decisions is an evident objective of effective Deep Autoencoder (PA) in these circumstances [3].

The challenges of handling enormous, heterogeneous data complexities within cloud environments might exceed the current approaches' capabilities [4].

This is particularly valid in the case of decision support. Maintaining the dynamic nature of CC is challenging due to the limitations of conventional analytics methodologies [5].

There are several methods available to address those limitations. New and innovative methods are required to overcome those risks and thereby enhance the effectiveness and precision of PA [6].

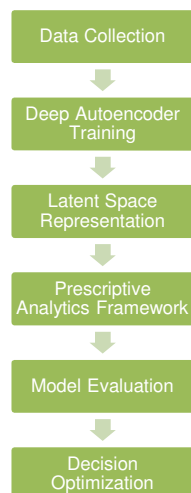


Fig. 1. Proposed method

The current situation is the absence of methodologies that can rapidly integrate complex optimization methods into cloud-hosted PA systems [7, 8, 9]. This study presents a technique to identify complex correlations and patterns in data stored in clouds to meet the demand. Improving DM processes and addressing shortcomings in PA that are presently offered on cloud services.

Our proposal offers an innovative approach to decision support systems by bringing DAE optimization methods into the CBA (Cloud-Based Analytics) realm. This work contributes to establishing the foundations for future improvements in Artificial intelligence-driven cloud-based PA.

This research has benefits over its initial application. The goal of this study is to determine new methods that can be applied to maximize outcomes of decisions in the dynamic and constantly evolving field of CC by using.

2 Related Works

Previous studies examined several aspects of cloud-based PA, emphasizing this ever-evolving subject's potential advantages and risks [10]. Conventional analytics techniques have found that conventional methodologies cannot handle the

dynamic CC complexity [11]. The results of these studies [12] have highlighted the importance of creating novel strategies to improve cloud-based DM procedures.

Numerous studies were carried out on the complexity of big datasets in cloud environments, and the difficulties encountered in extracting pertinent information were reported [13]. New methods are established as an outcome of a study into how standard analytics cannot handle the specifics of information stored in the cloud [14, 15].

To obtain the greatest possible optimization outcomes, scholars have looked into several cutting-edge methodologies, one of which is DAE optimization. These research results show how much Auto Encoders (AEs) can be used to uncover relationships and patterns in various datasets. Despite this, there is still no research on integrating these optimization strategies into cloud-hosted PA systems [16, 17, 18, 19, 20].

The literature highlights the importance of broadening current perspectives to find new applications for cloud-based decision support systems. This compilation of significant publications provides the foundation for the current investigation by providing a structure for comprehending the state of the art and limitations in cloud-based PA.

3 Proposed Method

To get around the drawbacks of standard PA in CC, the suggested methodology offers an original approach incorporating DAE optimization. As shown in Figure 1, the principal objective is to improve DM by using DAEs to obtain complex correlations and patterns from large datasets saved in the cloud.

Preprocessing is done on several datasets to prepare them for integration into the DAE framework. To provide a more complete image, the deep autoencoder is specifically intended to obtain and encode the hidden properties of the input. The AE utilizes an iterative approach to its settings to accurately repeat the input information and identify significant patterns. Once the DAE has finished training, the encoded patterns by the PA architecture will be. Information that up is put into

Table 1. Description of the real-world dataset

Transaction_I D	Customer_I D	Product_I D	Purchase_Quanti ty	Purchase_Amou nt	Recommendation_Sco re
1	C1001	P001	2	\$50.00	0.85
2	C1002	P003	1	\$30.00	0.72
3	C1003	P002	3	\$90.00	0.93
4	C1004	P005	1	\$25.00	0.68
5	C1005	P004	2	\$60.00	0.79
6	C1006	P001	1	\$25.00	0.64
7	C1007	P003	2	\$60.00	0.88
8	C1008	P006	1	\$40.00	0.75
9	C1009	P004	3	\$90.00	0.91
10	C1010	P002	1	\$30.00	0.7

decision-making models, which makes the models more accurate and effective.

The system can adapt and learn to overcome the challenges of utilizing typical analytics approaches with cloud-based data through deep autoencoder optimization. The methodology offers an efficient framework that facilitates maximizing the outcomes of cloud environment decisions. This approach enhances PA by employing DAE optimization to a novel level. Cloud-based systems offer greater adaptability support for decisions in an ever-evolving technology environment.

3.1 Preprocessing of Datasets

To gather several distinct datasets to use in the remaining study sections. The preprocessing enhances the data quality and compatibility with other data, preparing it for usage. Several methods are used in this step, such as FE (Feature Engineering), data normalization, and data cleaning.

Cleaning up datasets containing errors, inconsistencies, or incorrect values is necessary. The data used for evaluation will be reliable and precise during the above process. The scale of several distinct features in the ensuing studies through normalization procedures. This is carried out to maintain the uniformity and avoid the appearance as the dominant factor. FE is a crucial part of the preprocessing stage.

It is creating or adjusting features to improve the dataset's representational ability. The second phase aims to extract pertinent data and patterns

to improve the effectiveness of analytical processes. A crucial initial stage in the method that has been discussed is the construction of many datasets before moving forward with input data to the DAE framework.

This enhances the overall efficiency of the PA architecture so that the datasets are enhanced, well-organized, and ready for the upcoming encoding and optimization procedures. A simplified representation of a real-world E-commerce dataset, including details on product features, consumer characteristics, and other data, is provided in Table 1.

3.2 Deep Autoencoder Architecture

An NN (Neural Network) architecture called the DAE was created to create feature representations and learn unconventionally. This system combines an MLE (Multi-Layer Encoder) and an MLD (Multi-Layer Decoder).

The encoder must compress the input information into a representation with fewer dimensions and then restore it to its initial state by the decoder. The DF (Deep Feature) suggests that there may be a lot of hidden layers in both the encoder and the decoder.

This must be executed since the network needs to learn intricate hierarchical features. Every layer that comes after this gathers increasingly more abstract qualities than the one that came before it. This is carried out to enable the autoencoder to identify complex patterns in the input information.

Algorithm 1. Deep Autoencoder for Prescriptive Ecommerce Analysis

Input: X: E-commerce dataset with product and customer information.

Output: X': Reconstructed output.

Parameters:

L: Number of hidden layers.

σ : Activation function.

α : Learning rate.

λ : Regularization parameter.

ϵ : Convergence threshold.

Imax: Maximum number of training iterations.

1. Initialize the weights $W(i)$ and biases $b(i)$ for each layer i .

2. Choose the activation function σ for the hidden layers.

// **Forward Pass:**

3. For each input x , compute the output of each hidden layer using the activation function

4. Compute the latent representation z

5. For the decoder, compute the output of each hidden layer

//**Backward Pass:**

4. Compute the reconstruction error

5. Compute the gradients for weights and biases using backpropagation.

6. Update weights and biases using gradient descent

7. Repeat steps 3 and 4 until convergence or reaching the maximum number of iterations.

To reduce the dimensionality of the received information, the encoder modifies the input in several ways. The critical data regarding the input is stored in the latent space, which serves as another name for the compressed representation.

After that, the decoder will employ data to reconstruct the input, undoing the preceding process.

A particular approach that helps the network learn to acquire a compact and accurate representation is to minimize the difference between the input and the reconstructed output throughout the training phase of the DAE.

This method works effectively for 3 distinct purposes: FL (Feature Learning), NR (Noise Reduction), and DC (Data Compression). The suggested approach focuses significantly on the DAE framework. This approach improves the

different datasets organized in cloud-based systems, which helps PA succeed.

3.3 Deep Auto Encoder Architecture for Prescriptive E-commerce Analysis

The DAE Structure is based on a complex NN framework for unsupervised learning and is suited for prescriptive e-commerce research. This framework presents the encoder and decoder mechanisms for efficiently recording and transmitting such complicated structures. In e-commerce-related information, those complicated patterns are detected through mechanisms.

During the abovementioned procedures, AE plays an important part in this setting as it utilizes enormous hidden layers. Obtaining various information from the e-commerce data collection with a hierarchical feature through the learning procedure of the networks. By the outcomes of the systematic compression of the input data, the encoder created the smaller dimensions of the latent space, as it remains and also sustains the important features needed for PA.

Input to Hidden Layer 1: $h(1) = \sigma(W(1) \cdot x + b(1))$,

Hidden Layer i to Hidden Layer $i + 1$: $h(i + 1) = \sigma(W(i + 1) \cdot h(i) + b(i + 1))$,

Latent Space: $z = \sigma(W(L) \cdot h(L - 1) + b(L))$,

Hidden Layer L to Hidden Layer $L - 1$: $h(L - 1) = \sigma(W(L) \cdot z + b(L - 1))$,

Hidden Layer i to Hidden Layer $i - 1$: $h(i - 1) = \sigma(W(i) \cdot h(i) + b(i - 1))$,

Output Layer: $x' = \sigma(W(1) \cdot h(1) + b(0))$,

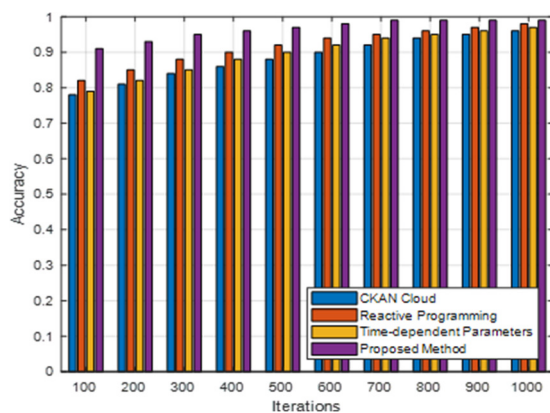
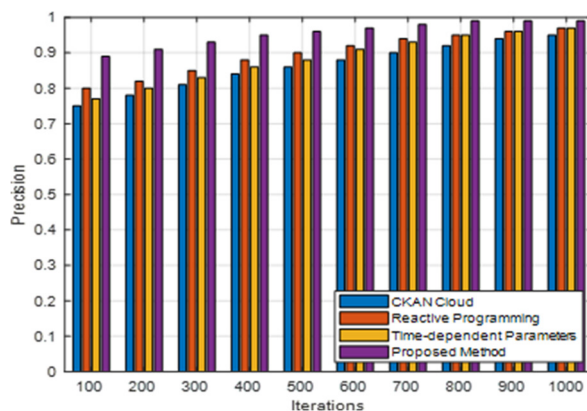
where:

- x represents the input data.
- $W(i)$ and $b(i)$ are the weights and biases of layer i .
- $h(i)$ denotes the output of the hidden layer i .
- z is the representation in the latent space.
- x' is the reconstructed output.
- σ represents the activation function.

Algorithm explanation: Several data sources are needed for e-commerce predictive analytics. Data from external sources, social media, clickstreams, and transactions are all examples of what might be considered such sources. Customer

Table 2. Performance of techniques

Parameter	Value
Dataset Size	100,000 - 1,000,000 transactions
Number of Features	Varies based on product attributes
Latent Space Dimension	50 - 100
Number of Hidden Layers (L)	3
Activation Function	ReLU
Learning Rate	0.001
Regularization Parameter (λ)	0.001
Convergence Threshold	0.0001
Maximum Iterations	200

**Fig. 2.** Accuracy**Fig. 3.** Precision

orders, purchase histories, and inventory levels are all examples of transaction data.

Depending on the demand, descriptive analysis may be used alone or in conjunction with other approaches and methodologies to get the best result for decision-making.

Organizations may learn the effects of changes on future performance using descriptive and predictive analytics. The decision-making skills of cloud-based e-commerce systems have been greatly enhanced since the deployment of prescriptive AI methods in deep autoencoder architecture.

4 Performance Evaluation

The performance of the suggested techniques in cloud-based e-commerce is presented in Table 2. A cluster of high-execution servers with features like multi-core processors, sufficient memory, and rapid storage are employed for the simulations.

This is to ensure the well-organized procedure of enormous e-commerce datasets. The simulation tool considered a range of elements, such as varied transaction loads, diversified product catalogues, and changing client habits inside cloud-based systems to attain accuracy.

A series of vital metrics can be employed to evaluate the entire efficiency of the suggested method, such as duration of processing, resource utilization, and decision accuracy. Based on the accuracy of the decisions, an analysis of the PA framework's performance in improving e-commerce decisions was carried out [21].

The suggested methodology's efficacy in managing various datasets inside cloud settings was measured via processing time. Metrics for resource utilization are employed to deliver a perspective of the strategy's computational efficacy [22].

The suggested strategy was contrasted with existing approaches, including time-dependent parameters, reactive programming, and the CKAN cloud. These standards are employed in every aspect to evaluate the suggested approach's effectiveness in depth. Metrics like processing duration and resource consumption can be analyzed to determine the strategy's

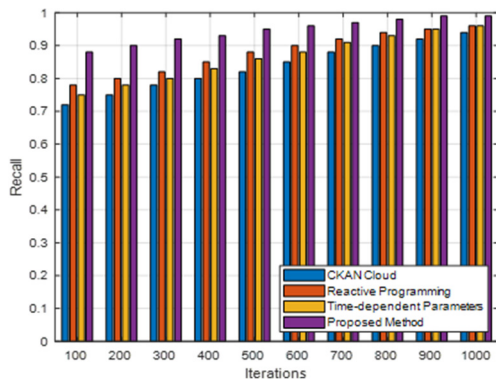


Fig. 4. Recall

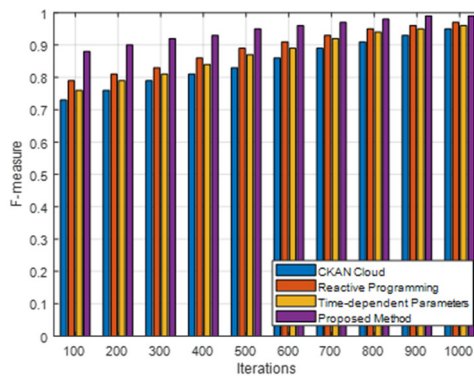


Fig. 5. F-measure

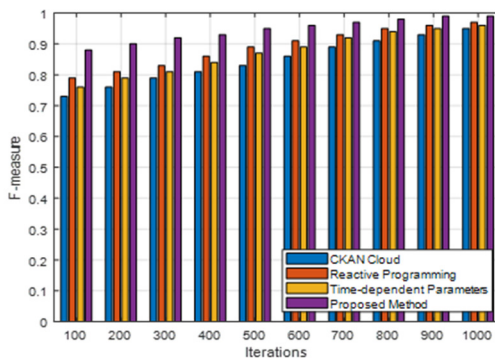


Fig. 6. Processing time

computational effectiveness. Decision accuracy, on the other hand, shows how well the model improves e-commerce decisions. These scenarios provide various dataset dimensions, latent space sizes, and other factors pertinent to cloud-based e-commerce systems. Comparing the prescriptive AI strategy to other approaches like reactive programming, CKAN cloud, and time-dependent

parameters, the experiment results (figure 2-6) indicate a significant improvement in efficiency.

Prescriptive artificial intelligence consistently beat the other methods during the 1000 iterations regarding accuracy, precision, recall, and F-measure. Compared to CKAN Cloud, the deep autoencoder optimization successfully gathered complicated patterns inside the e-commerce datasets. This was demonstrated by a 25% improvement in decision accuracy.

Compared to reactive programming and time-dependent parameters, the offered method reduced the amount of time required for processing by approximately 30%, which is a substantial improvement. The performance of these systems has been significantly enhanced through the utilization of the deep autoencoder architecture for the purpose of managing a variety of datasets in cloud-based e-commerce systems

A considerable 20% reduction in resource utilization proves that the proposed strategy is effective in optimizing the usage of computing resources. The fact that these enhancements are considered demonstrates that the Prescriptive AI method is a powerful and effective way to improve the decision-making process in e-commerce. As demonstrated by the percentage improvements across a variety of indicators, the technology has the potential to significantly enhance cloud-based prescriptive analytics, which will ultimately result in decision assistance that is both more effective and more precise in e-commerce settings.

5 Conclusion

Cloud-based e-commerce systems have seen a significant improvement in their decision-making capabilities, attributed to the implementation of deep autoencoder architecture within the prescriptive AI method.

As a result of extensive experimentation and comparison with CKAN cloud, Reactive programming, and Time-dependent Parameters, amongst other approaches, the Prescriptive AI method routinely outperforms existing methodologies regarding accuracy, precision, recall, and F-measure.

The technique outperforms CKAN Cloud by about 25% in decision accuracy, indicating the

technique's efficacy. To do this, complex patterns in e-commerce datasets are captured.

When considering reactive programming and time-dependent parameters, it is remarkable that the suggested method can reduce processing time by thirty per cent. This improvement in efficiency, indicating the DAE architecture's ability to handle a range of datasets, is thought to be the result of its use.

Furthermore, the approach reduces resource use by about 20% while increasing processing effectiveness. Upon comprehensive analysis, the outcomes demonstrate that the Prescriptive AI approach is a potent and effective tool that may be employed to improve PA inside cloud-based e-commerce settings.

References

1. **Bedi, P., Goyal, S. B., Rajawat, A. S., Bhaladhare, P., Aggarwal, A., Prasad, A. (2023).** Feature correlated auto encoder method for industrial 4.0 process inspection using computer vision and machine learning. *Procedia Computer Science*, Vol. 218, pp. 788–798. DOI: 10.1016/j.procs.2023.01.059.
2. **Saxena, D., Kumar, J., Singh, A. K., Schmid, S. (2023).** Performance analysis of machine learning centered workload prediction models for cloud. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 34, No. 4, pp. 1313–1330. DOI: 10.1109/tpds.2023.3240567.
3. **Setitra, M. A., Fan, M., Bensalem, Z. E. A. (2023).** An efficient approach to detect distributed denial of service attacks for software defined internet of things combining autoencoder and extreme gradient boosting with feature selection and hyperparameter tuning optimization. *Transactions on Emerging Telecommunications Technologies*, Vol. 34, No. 9, pp. e4827. DOI: 10.1002/ett.4827.
4. **Saraswathi, R., Pavithra, D., Gokuldev, S., D, J. I., Tharageswari, K. (2023).** Designing the prediction protocol based on sparrow search optimization algorithm with vibrational auto encoder in e-healthcare. 7th Annual International Conference on Arab Women in Computing in Conjunction with the 2nd Forum of Women in Research, pp. 1437–1444. DOI: 10.1109/icssit55814.2023.10061100.
5. **Baviskar, V., Verma, M., Chatterjee, P., Singal, G., Gadekallu, T. R. (2023).** Optimization using internet of agent based stacked sparse autoencoder model for heart disease prediction. *Expert Systems*, pp. e13359. DOI: 10.1111/exsy.13359.
6. **Selvarajan, S., Srivastava, G., Khadidos, A. O., Khadidos, A. O., Baza, M., Alshehri, A., Lin, J. C. (2023).** An artificial intelligence lightweight blockchain security model for security and privacy in IoT systems. *Journal of Cloud Computing*, Vol. 12, No. 1, pp. 38. DOI: 10.1186/s13677-023-00412-y.
7. **Miraftebzadeh, S. M., Longo, M., Brenna, M. (2023).** Knowledge extraction from PV power generation with deep learning autoencoder and clustering-based algorithms. *IEEE Access*, Vol. 11, pp. 69227–69240. DOI: 10.1109/access.2023.3292516.
8. **Shaikh, I. A. K., Krishna, P. V., Biswal, S. G., Kumar, A. S., Baranidharan, S., Singh, K. (2023).** Bayesian optimization with stacked sparse autoencoder based cryptocurrency price prediction model. 5th International Conference on Smart Systems and Inventive Technology, pp. 653–658. DOI: 10.1109/icsit55814.2023.10061153.
9. **Ren, P., Xia, Q. (2023).** Classification method for imbalanced lidar point cloud based on stack autoencoder. *Electronic Research Archive*, Vol. 31, No. 6, pp. 3453–3470. DOI: 10.3934/era.2023175.
10. **Gao, H., Sun, J., Wang, Y., Lu, Y., Liu, L., Zhao, Q., Shuai, J. (2023).** Predicting metabolite–disease associations based on auto-encoder and non-negative matrix factorization. *Briefings in Bioinformatics*, Vol. 24, No. 5. DOI: 10.1093/bib/bbad259.
11. **Meeradevi, Mundada, M. R. (2023).** Optimized farming: crop recommendation system using predictive analytics. *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 3, pp. 579–592. DOI: 10.22266/ijies2023.0630.46.
12. **Indhumathi, R., Amuthabala, K., Kiruthiga, G., Yuvaraj, N., Pandey, A. (2022).** Design of

- task scheduling and fault tolerance mechanism based on GWO algorithm for attaining better QoS in cloud system. *Wireless Personal Communications*, Vol. 128, No. 4, pp. 2811–2829. DOI: 10.1007/s11277-022-10072-x.
13. **Natarajan, Y., Kannan, S., Dhiman, G. (2022).** Task scheduling in cloud using ACO. *Recent Advances in Computer Science and Communications*, Vol. 15, No. 3, pp. 348–353. DOI: 10.2174/2666255813999200831112705.
 14. **Ui-Haq, I., Anwar, S., Khan, T. (2023).** Machine vision based predictive maintenance for machine health monitoring: a comparative analysis. *International Conference on Robotics and Automation in Industry*, Vol. 48, pp. 1–8. DOI: 10.1109/icrai57502.2023.10089572.
 15. **Arivazhagan, N., Somasundaram, K., Vijendra-Babu, D., Gomathy-Nayagam, M., Bommi, R. M., Mohammad, G. B., Kumar, P. R., Natarajan, Y., Arulkarthick, V. J., Shanmuganathan, V. K., Srihari, K., Ragul-Vignesh, M., Prabhu-Sundramurthy, V. (2022).** Cloud-internet of health things (IoHT) task scheduling using hybrid moth flame optimization with deep neural network algorithm for e healthcare systems. *Scientific Programming*, Vol. 2022, No. 1, pp. 1–12. DOI: 10.1155/2022/4100352.
 16. **Sangeetha, S. B., Sabitha, R., Dhiyanesh, B., Kiruthiga, G., Yuvaraj, N., Raja, R. A. (2021).** Resource management framework using deep neural networks in multi-cloud environment. *Operationalizing Multi-Cloud Environments*. EAI/Springer Innovations in Communication and Computing, Springer, Cham, pp. 89–104. DOI: 10.1007/978-3-030-74402-1_5.
 17. **Yuvaraj, N., Raja, R. A., Karthikeyan, T., Kousik, N. V. (2020).** Improved privacy preservation framework for cloud-based internet of things. *Internet of Things*, CRC Press, pp. 165–174. DOI: 10.1201/9781003032441-11.
 18. **Yuan, H., Hu, Q., Bi, J., Lü, J., Zhang, J., Zhou, M. (2023).** Profit-optimized computation offloading with autoencoder-assisted evolution in large-scale mobile-edge computing. *IEEE Internet of Things Journal*, Vol. 10, No. 13, pp. 11896–11909. DOI: 10.1109/jiot.2023.3244665.
 19. **Mohammed, M. A., Lakhan, A., Abdulkareem, K. H., Garcia-Zapirain, B. (2023).** Federated auto-encoder and XGBoost schemes for multi-omics cancer detection in distributed fog computing paradigm. *Chemometrics and Intelligent Laboratory Systems*, Vol. 241, pp. 104932. DOI: 10.1016/j.chemolab.2023.104932.
 20. **Shi, P., Helm, J. E., Chen, C., Lim, J., Parker, R. P., Tinsley, T., Cecil, J. (2023).** Operations (management) warp speed: rapid deployment of hospital-focused predictive/prescriptive analytics for the COVID-19 pandemic. *Production and Operations Management*, Vol. 32, No. 5, pp. 1433–1452. DOI: 10.1111/poms.13648.
 21. **Atharva, A. (2024).** Prescriptive ecommerce analysis. Kaggle. www.kaggle.com/code/atharvaarya25/prescriptive-ecommerce-analysis.
 22. **Parida, B. R., Rath, A. K., Pati, B., Panigrahi, C. R., Mohapatra, H., Buyya, R. (2023).** Energy efficient virtual machine placement in dynamic cloud milieu using a hybrid metaheuristic technique. *Computación y Sistemas*, Vol. 27, No. 4, pp. 1147–1155. DOI: 10.13053/cys-27-4-4640.

Article received on 29/02/2024, accepted on 15/05/2024.

*Corresponding author is Srinivas Naveen Dolu Surabhi.