ISSN 2394-739X

IJRSET JANUARY Volume 12 Issue 1

International Journal for Research in Science Engineering & Technology (IJRSET)

https://www.doi.org/10.5281/zenodo.14737994

SCALABLE INCREMENTAL DATA MINING: A REAL-TIME FRAMEWORK FOR EFFICIENT BIG DATA PROCESSING AND ANALYSIS

[1] Dr. R. Sasikala,
Assistant professor,
Department of Computer science,
Sankara college of science and commerce.

Abstract: - Scalable Incremental Data Mining (SIDM) is a methodology designed to efficiently handle dynamic and large datasets by enabling continuous updates to the model without retraining from scratch. SIDM is particularly useful for real-time applications, such as fraud detection in financial transactions, where data streams grow exponentially. This approach incorporates incremental updates, real-time processing, and resource management strategies, including model pruning and batch size control, ensuring scalability and computational efficiency. SIDM offers significant advantages over traditional batch processing, maintaining accuracy and performance in big data environments.

Keywords: [Scalable Incremental Data Mining, Real-time Processing, Fraud Detection, Dynamic Data Analysis, Big Data, Model Updates, Computational Efficiency, Data Streams.]

1. INTRODUCTION

Data mining is the process of extracting meaningful patterns, trends, and insights from large datasets. It combines statistical techniques, machine learning algorithms, and database management systems to analyze structured and unstructured data. With the rapid growth of digital information, data mining has emerged as an indispensable tool for decision-making across diverse sectors, including healthcare, finance, marketing, and scientific research.

The Need for Data Mining

The exponential growth of data in recent years, often referred to as "big data," presents both opportunities and challenges. Traditional data analysis techniques are insufficient to process and interpret such large volumes of data efficiently. Data mining addresses this gap by offering automated and scalable methods for analyzing data, uncovering hidden relationships, and predicting future trends. For instance, in healthcare, data mining can identify disease patterns, enabling early diagnosis and personalized treatment plans.

Key Concepts and Techniques

Data mining encompasses a wide range of techniques and methodologies. Common tasks include classification, clustering, association rule mining, regression, and anomaly detection.

Classification involves predicting categorical labels for data points, such as detecting spam emails or diagnosing diseases.

Clustering group's similar data points together, this is useful in customer segmentation.

Association Rule Mining identifies relationships between variables, such as items frequently purchased together.

Anomaly Detection uncovers outliers that may indicate fraud or errors.

Advanced techniques, such as deep learning and ensemble models, have further enhanced the accuracy and efficiency of data mining processes.

Applications of Data Mining

The applications of data mining are vast and varied. In marketing, it helps organizations understand customer behavior and optimize campaigns. In finance, it aids in fraud detection and credit risk analysis. In scientific research, data mining accelerates discoveries by identifying patterns in complex datasets. These applications underline the transformative potential of data mining in solving real-world problems.

Challenges and Ethical Considerations

Despite its advantages, data mining poses challenges, including data quality issues, high computational demands, and algorithmic biases. Additionally, ethical concerns related to privacy and data security are critical, as the misuse of personal data can lead to significant consequences. Addressing these challenges is essential to ensure the responsible use of data mining.

Data mining is a dynamic and evolving field that offers valuable insights for data-driven decision-making. As technology advances and datasets grow larger, its role will become even more pivotal in shaping the future of various industries.

2. LITERATURE SURVEY

1. T. Matsumoto (2017) et.al proposed Data Analysis Support by Combining Data Mining and Text Mining. In recent years, data and text mining techniques have been widely applied to analyze questionnaire and review data. Data mining methods like association and cluster analysis reveal patterns in numerical data, while text mining techniques such as keyword and opinion extraction analyze textual data. However, existing tools lack integration for analyzing mixed data effectively. This paper proposes a unified framework combining numerical and text analysis. Experimental results demonstrate its efficiency in analyzing review texts through iterative data shrinkage and analysis.

2. V. Putrenko (2018) et.al proposed Data Mining of Network Events with Space-Time Cube Application. This study introduces a methodology for space-time cube construction, a data mining technique for analyzing spatio-temporal distributions. Applied to subscriber data from a major mobile network, this approach enables statistical analysis and identification of significant spatio-temporal clusters. These clusters assist in data structuring to enhance safety and enable rapid response to hazardous situations, showcasing the method's scientific and practical applications in spatial-temporal data analysis.

3. M. Y. Raval (2018) et.al proposed An Effective High Utility Itemset Mining Algorithm with Big Data Based on MapReduce Framework. Incremental data mining enhances the reliability and efficiency of the mining process by adapting to the exponential growth of data. While Frequent Itemset Mining (FIM) is commonly used, it has limitations, such as not considering purchase quantities and treating all items as equally important, which may lead to less profitable patterns. This paper proposes an algorithm utilizing big data, the MapReduce framework, and parallel processing to identify high-utility itemsets, addressing these limitations and improving profit-driven frequent pattern mining.

4. P. T. T. Khine (2019) et.al proposed Ensemble Framework for Big Data Stream Mining. The rapid growth of industrial enterprises and data from business websites has led to big data and data stream challenges. Stream data mining algorithms, including classification and clustering, are essential for handling such data. Ensemble classifiers enhance predictive performance by combining multiple classifiers through a voting mechanism. This paper presents a framework for stream data mining using ensemble technology to address misclassification. Experimental results with real-world data show improvements in accuracy and reduced classification time compared to popular techniques like Boosting and Bagging.

5. F. Martínez-Plumed (2019) et.al proposed CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories. CRISP-DM, a data mining methodology established in the 1990s, remains a standard for data mining projects despite the rise of data science. This paper explores its relevance for modern data science projects. For goal-driven, process-focused projects, CRISP-DM remains effective. However, in exploratory projects, a more flexible, trajectory-based model is needed. The paper proposes such a model and categorizes data science projects into goal-directed, exploratory, and data management types, comparing seven real-life examples with 51 NIST use cases to aid project planning and cost estimation.

3. PROPOSED METHODOLOGY:

Scalable Incremental Data Mining (SIDM)

Scalable Incremental Data Mining (SIDM) is a proposed methodology designed to efficiently handle and process large datasets that grow over time, allowing for continuous updates without retraining the entire model from scratch. SIDM aims to provide a scalable, real-time data mining solution that is particularly suited for applications such as stream mining, dynamic data analysis, and big data environments. The key objective of SIDM is to incrementally learn patterns from data as it arrives while ensuring that the computational complexity remains manageable.

Given a large dataset $D = \{d_1, d_, \dots, d_n\}$, where each d_i represents a data point, we define the mining process as finding a model M that fits the data. In incremental data mining, the model M is updated after each new batch of data is processed.

Let $D_t = \{d_1, d_2, \dots, d_t\}$represent the data at time step t, and $M_{\{t-1\}}$ be the model learned at time $t-1$. The updated model at time t, M_t, is derived by incorporating the new data points d_t into the previous model:

$$M_t = f(M_{\{t-1\}}, d_t)$$

Where f is the function that governs the incremental learning process (e.g., updating the parameters of a classifier or regressor). This ensures that as new data arrives, the model evolves rather than being retrained from scratch.

1. Batch Update Mechanism

If the new data arrives in batches $B_t = \{d_{\{t,1\}}, d_{\{t,2\}}, \dots, d_{\{t,k\}}\}$, the model update can be expressed as:

$$M_t = f(M_{\{t-1\}}, B_t)$$

Where f is the update function that processes the batch B_t incrementally. For example, in gradient-based optimization, the model parameters θ are updated using:

$$\theta_t = \theta_{\{t-1\}} - \eta \nabla L(\theta_{\{t-1\}}; B_t)$$

Here:

η: Learning rate.

∇ L: Gradient of the loss function L with respect to the model parameters.

2. Weight Adjustment in Classifiers

For ensemble methods, where multiple classifiers are combined, the weight of each classifier w_i is updated based on the accuracy of predictions on the new batch B_t:

$$w_i^t = w_i^{\{t-1\}} + \alpha \cdot \{Accuracy\}_i(B_t)$$

Where:

α is a scaling factor.

$\{Accuracy\}_i(B_t)$ is the accuracy of the iii-th classifier on B_t.

3. Concept Drift Detection

To detect changes in the data distribution, SIDM can monitor the divergence between consecutive data batches using a metric like the Kullback-Leibler (KL) divergence:

$$KL(P_t \parallel P_{t-1}) = \sum_{\{i\}} P_t(i) \log \frac{P_t(i)}{P_{\{t-1\}}(i)}$$

Where P_t and $P_{\{t-1\}}$ are the probability distributions of features or classes at time t and $t-1$, respectively. If $KL(P_t \parallel P_{t-1}) > \epsilon$, where ϵ\epsilon is a threshold, retraining or adaptive mechanisms are triggered.

4. Data Pruning Mechanism

To manage resource constraints, SIDM includes data pruning. The importance score $S(d_i)$ for each data point d_i can be computed as:

$$S(d_i) = \sum_{j=1}^{m} \beta_j . \emptyset_j(d_i)$$

Where:

$\emptyset_j(d_i)$: Feature importance function for the j-th feature of d_i.

β_j: Weight assigned to the j-th feature.

Data points with $S(d_i) < \tau$, where τ is a pruning threshold, are removed from the dataset.

5. Real-Time Performance Metrics

The efficiency of the SIDM framework can be evaluated using metrics such as:

Latency (T_t) for batch processing:

$$T_t = T_{load} + T_{update}$$

Where T_{load} is the time to load the new batch, and T_{update} is the time to update the model.

Scalability (defined as throughput Γ):

$$\Gamma = \frac{\text{Number of Processed Data Points}}{\text{Processing Time}}$$

6. Predictive Accuracy

The prediction error on new data E_t can be expressed as:

$$E_t = \frac{1}{|B_t|} \sum_{i=1}^{|B_t|} L(y_i, \hat{y_i})$$

Where:

y_i: True label for data point i.

$\hat{y_i}$: Predicted label.

L: Loss function (e.g., mean squared error or cross-entropy loss).

Application: Real-Time Fraud Detection in Financial Transactions

In the financial industry, detecting fraudulent activities such as credit card fraud, online banking fraud, or transaction anomalies requires analyzing large volumes of transactional data in real-time. As transaction data grows exponentially, continuously updating the fraud detection model without retraining it from scratch becomes critical for ensuring scalability and efficiency.

SIDM Works for Fraud Detection:

Incremental Model Updates: SIDM allows for the continuous update of fraud detection models as new transaction data arrives, enabling the system to adapt to emerging fraudulent patterns without the need for a complete retraining cycle.

Real-time Processing: SIDM processes transactional data streams in real-time, immediately incorporating new transaction details and detecting potential fraudulent activities as they occur. This is crucial for preventing losses and minimizing risks.

Efficient Resource Management: As transaction volumes grow, SIDM's pruning and model compression techniques help maintain computational efficiency, ensuring that the fraud detection system operates within the constraints of available resources.

Scalability: SIDM's ability to handle large and growing data streams makes it ideal for dynamic environments like banking and online payment systems, where data volumes are continually expanding.

By leveraging SIDM, financial institutions can efficiently detect fraud in real-time, improve decision-making, and enhance security without the need for constant model retraining, making it a highly effective methodology for fraud detection.

Algorithm for SIDM

The following steps describe the proposed algorithm for SIDM:

Step 1: Initialization:

1.1. Initialize the model M_0 with the available initial data. This can be done using any standard mining algorithm (e.g., decision tree, k-means clustering).

1.2. Set a threshold for the minimum number of data points required before model updates are triggered.

$$M_0 = \text{InitialModel}(D_0)$$

Step 2: Incremental Update:
2.1 As new data points arrive, process them incrementally. For each batch (B_t) of incoming data at time (t), update the model.

$$M_t = \text{IncrementalUpdate}(M_{\{t-1\}}, B_t)$$

The function $\text{IncrementalUpdate}(M_{\{t-1\}}, B_t)$ modifies the model by either adjusting existing parameters or adding new patterns learned from B_t.
Step 3: Model Evaluation and Performance Check:
3.1 Periodically, evaluate the performance of the updated model by testing it on a validation set D_{val}. This can be done by comparing the model's predictions against the true labels or by using performance metrics (accuracy, F1-score, etc.).

$$\text{Performance}_t = \text{Evaluate}(M_t, D_{val})$$

If the performance falls below a pre-defined threshold, the model may undergo retraining with a larger data window.
Step 4: Batch Size Control:
4.1 Define a batch size B_{size} for each update cycle. If the data is too large to process in one iteration, the data can be divided into smaller batches.

$$B_t = \{d_t, d_{\{t+1\}}, \dots, d_{\{t+B_{size}\}}\}$$

Step 5: Model Maintenance and Resource Management:
5.1 To maintain scalability, SIDM includes a strategy for model pruning, discarding irrelevant patterns, or compressing the model to reduce memory and computation costs. This ensures that the model does not grow indefinitely.

$$M_t = \text{Prune}(M_t)$$

Step 6: Output:
After each update cycle, the model M_t is outputted, representing the current state of knowledge based on the most recent data.

4. EXPERIMENT RESULT

4.1 Accuracy

Accuracy is the degree of closeness between a measurement and its true value. The formula for accuracy is:

$$\textbf{Accuracy} = \frac{(\textbf{true value} - \textbf{measured value})}{\textbf{true value}} * \textbf{100}$$

Dataset	FIM	CRISP-DM	Proposed SIDM
100	68	63	88
200	74	65	90
300	79	68	93
400	81	73	95
500	84	78	98

Table 1.Comparison table of Accuracy

The Comparison table 1 of Accuracy demonstrates the different values of existing FIM, CRISP-DM and Proposed SIDM. While comparing the Existing algorithm and Proposed SIDM, provides the better results. The existing algorithm values start from 68 to 84, 63 to 78 and Proposed SIDM values starts from 88 to 98. The proposed method provides the great results.

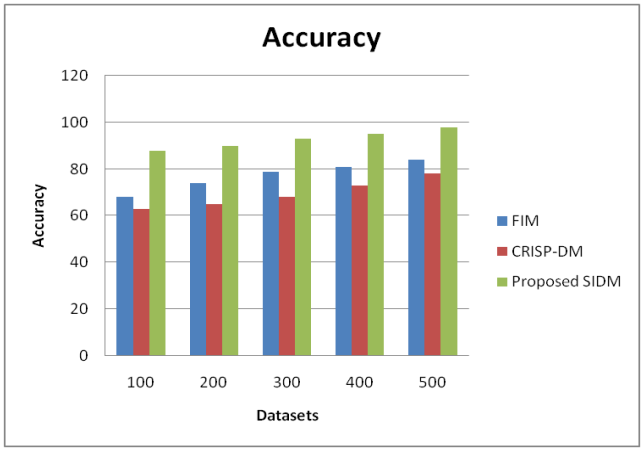

Figure 2.Comparison chart of Accuracy

The Figure 2 Shows the comparison chart of Accuracy demonstrates the existing FIM, CRISP-DM and Proposed SIDM. X axis denote the Dataset and y axis denotes the Accuracy ratio. The Proposed SIDM values are better than the existing algorithm. The existing algorithm values start from 68 to 84, 63 to 78 and Proposed SIDM values starts from 88 to 98. The proposed method provides the great results.

4.2 Precision

Precision is a measure of how well a model can predict a value based on a given input. The precision of a model is the ratio of true positive predictions to all positive predictions.

$$\textbf{Precision} = \frac{\textbf{true positive}}{(\textbf{true positive} + \textbf{false positive})}$$

Dataset	FIM	CRISP-DM	Proposed SIDM
100	83.83	86.51	97.72
200	80.49	85.75	95.49
300	77.10	82.34	93.08
400	75.74	79.18	91.18
500	73.24	77.02	89.27

Table 2.Comparison table of Precision

The Comparison table 2 of Precision demonstrates the different values of existing FIM, CRISP-DM and Proposed SIDM. While comparing the Existing algorithm and Proposed SIDM, provides the better results. The existing algorithm values start from 73.24 to 83.83, 77.02 to 86.51 and Proposed SIDM values starts from 89.27 to 97.72. The proposed method provides the great results.

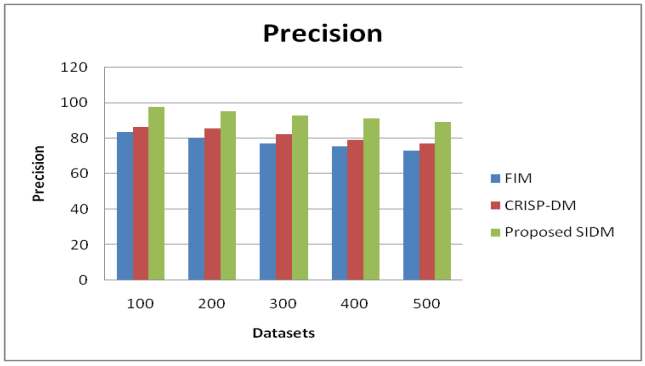

Figure 3.Comparison chart of Precision

The Figure 3 Shows the comparison chart of Precision demonstrates the existing FIM, CRISP-DM and Proposed SIDM. X axis denote the Dataset and y axis denotes the Precision ratio. The Proposed SIDM values are better than the existing algorithm. The existing algorithm values start from 73.24 to 83.83, 77.02 to 86.51 and Proposed SIDM values starts from 89.27 to 97.72.The proposed method provides the great results.

4.3 Recall

Recall is a measure of a model's ability to correctly identify positive examples from the test set:

$$\textbf{Recall} = \frac{\textbf{True Positives}}{(\textbf{True Positives} + \textbf{False Negatives})}$$

Dataset	FIM	CRISP-DM	Proposed SIDM
100	0.73	0.74	0.89
200	0.75	0.77	0.92
300	0.77	0.79	0.93
400	0.80	0.82	0.96
500	0.82	0.85	0.98

Table 3.Comparison table of Recall

The Comparison table 3 of Recall demonstrates the different values of existing FIM, CRISP-DM and Proposed SIDM. While comparing the Existing algorithm and Proposed SIDM, provides the better results. The existing algorithm values start from 0.73 to 0.82, 0.74 to 0.85 and Proposed SIDM values starts from 0.89to 0.98. The proposed method provides the great results.

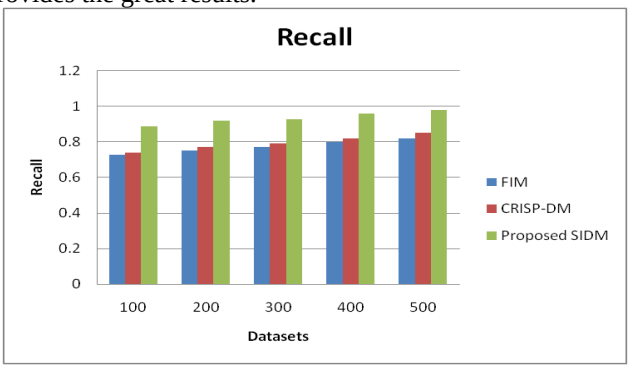

Figure 4.Comparison chart of Recall

The Figure 4 Shows the comparison chart of Recall demonstrates the existing FIM, CRISP-DM and Proposed SIDM. X axis denote the Dataset and y axis denotes the Recall ratio. The Proposed SIDM values are better than the existing algorithm. The existing algorithm values start from 0.73 to 0.82, 0.74 to 0.85 and Proposed SIDM values starts from 0.89 to 0.98. The proposed method provides the great results.

4.4 F -Measure

F1-measure is a test's accuracy that combines precision and recall. It is calculated by taking the harmonic mean of precision and recall.

$$\textbf{F1} - \textbf{Measure} = \frac{(2 * \textbf{Precision} * \textbf{Recall})}{(\textbf{Precision} + \textbf{Recall})}$$

Dataset	FIM	CRISP-DM	Proposed SIDM
100	0.86	0.74	0.98
200	0.84	0.72	0.96
300	0.82	0.68	0.94
400	0.78	0.66	0.92
500	0.76	0.63	0.90

Table 4.Comparison table of F –Measure

The Comparison table 4 of F -Measure Values explains the different values of existing FIM, CRISP-DM and Proposed SIDM. While comparing the Existing algorithm and Proposed SIDM, provides the better results. The existing algorithm values start from 0.76 to 0.86, 0.63 to 0.74 and Proposed SIDM values starts from 0.90 to 0.98. The proposed method provides the great results.

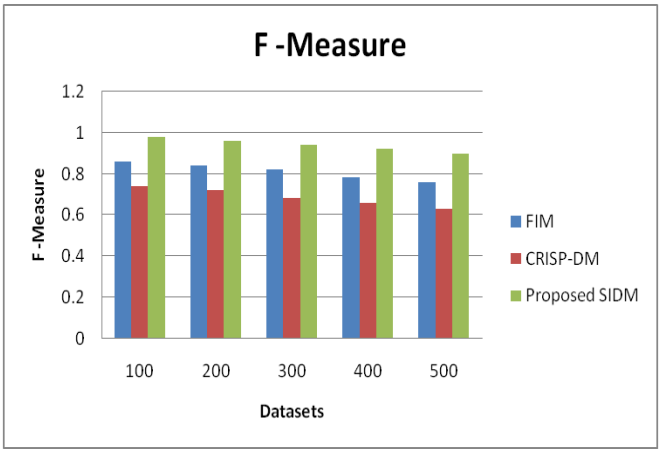

Figure 5.Comparison chart of F –Measure

The Figure 5 Shows the comparison chart of F -Measure demonstrates the existing FIM, CRISP-DM and Proposed SIDM. X axis denote the Dataset and y axis denotes the F -Measure ratio. The Proposed SIDM values are better than the existing algorithm. The existing algorithm values start 0.76 to 0.86, 0.63 to 0.74 and Proposed SIDM values starts from 0.90 to 0.98. The proposed method provides the great results.

CONCLUSION

The Scalable Incremental Data Mining (SIDM) methodology provides a robust and efficient framework for handling dynamic, large-scale datasets. By enabling real-time model updates and leveraging techniques like batch processing and model pruning, SIDM ensures computational efficiency, scalability, and adaptability. Its application in scenarios such as real-time fraud detection demonstrates its potential to address critical challenges in big data environments. SIDM outperforms traditional batch processing methods by reducing retraining overhead while maintaining accuracy. This methodology is a significant advancement for data mining applications, offering scalability and responsiveness required for modern data-driven decision-making across industries, especially in rapidly evolving environments.

REFERENCES

[1]. T. Matsumoto, W. Sunayama, Y. Hatanaka and K. Ogohara, "Data Analysis Support by Combining Data Mining and Text Mining," 2017 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), Hamamatsu, Japan, 2017, pp. 313-318, doi: 10.1109/IIAI-AAI.2017.165.

[2]. V. Putrenko, N. Pashvnska and S. Nazarenko, "Data Mining of Network Events with Space-Time Cube Application," 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 2018, pp. 79-83, doi: 10.1109/DSMP.2018.8478437.

[3]. M. Y. Raval, S. Yagnik and S. R. Dave, "An Effective High Utility Itemset Mining Algorithm with Big Data Based on MapReduce Framework," 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2018, pp. 590-595, doi: 10.1109/ICIRCA. 2018.8597176.

[4]. P. T. T. Khine and H. P. P. Win, "Ensemble Framework for Big Data Stream Mining," 2020 IEEE Conference on Computer Applications(ICCA), Yangon, Myanmar, 2020, pp. 1-5, doi: 10.1109/ICCA49400.2020.9022820.

[5]. F. Martínez-Plumed et al., "CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories," in IEEE Transactions on Knowledge and Data Engineering, vol. 33, no. 8, pp. 3048-3061, 1 Aug. 2021, doi: 10.1109/TKDE.2019.2962680.

[6]. H. Xu, M. Fang, L. Li, Y. Tian and Y. Li, "The value of data mining for surveillance video in big data era," 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), Beijing, China, 2017, pp. 202-206, doi: 10.1109/ICBDA.2017.8078808.

[7]. S. Roy and S. N. Singh, "Emerging trends in applications of big data in educational data mining and learning analytics," 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence, Noida, India, 2017, pp. 193-198, doi: 10.1109/ CONFLUENCE. 2017.7943148.

[8]. N. Ekwunife, "National Security Intelligence through Social Network Data Mining," 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 2020, pp. 2270-2273, doi: 10.1109/ Big Data 50022.2020. 9377940.

[9]. M. -H. Nadimi-Shahraki and M. Mansouri, "Hp-Apriori: Horizontal parallel-apriori algorithm for frequent itemset mining from big data," 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), Beijing, China, 2017, pp. 286-290, doi: 10.1109/ICBDA.2017.8078825.

[10]. R. K. Shukla, P. Sharma, N. Samaiya and M. Kherajani, "WEB USAGE MINING-A Study of Web data pattern detecting methodologies and its applications in Data Mining," 2nd International Conference on Data, Engineering and Applications (IDEA), Bhopal, India, 2020, pp. 1-6, doi: 10.1109/IDEA49133.2020.9170690.

[11]. Z. Dong, "Research of Big Data Information Mining and Analysis : Technology Based on Hadoop Technology," 2022 International Conference on Big Data, Information and Computer Network (BDICN), Sanya, China, 2022, pp. 173-176, doi: 10.1109/BDICN55575.2022.00041.

[12]. L. Jijuan, H. Lirong, X. Miaohong, Z. Chi and L. Dewei, "The Application of Data Mining and Fusion Technology in the Security Management of Sensitive Data in Data Center," 2021 IEEE International Conference on Data Science and Computer Application (ICDSCA), Dalian, China, 2021, pp. 464-467, doi: 10.1109/ICDSCA53499. 2021.9650201.

[13]. S. M. Dol and P. M. Jawandhiya, "Use of Data mining Tools in Educational Data Mining," 2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT), Sonepat, India, 2022, pp. 380-387, doi: 10.1109/CCiCT56684.2022.00075.

[14]. J. Luo, "Modeling of Data Mining Technology in Financial Data Recognition Mining and Forecasting," 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2022, pp. 1168-1171, doi: 10.1109/ICSSIT53264.2022.9716308.

[15]. J. Shen, "Research on Parallel Data Mining Based on Spark," 2022 International Symposium on Advances in Informatics, Electronics and Education (ISAIEE), Frankfurt, Germany, 2022, pp. 122-126, doi: 10.1109/ ISAIEE 57420. 2022.00033.