Application of Machine Learning Methods for Timely Detection and Analysis of Anomalies

Anzhelika Stakhova, Yurii Kyrychuk and Nataliia Nazarenko

October 18, 2024

Application of machine learning methods for timely detection and analysis of anomalies

1st Anzhelika Stakhova
department of system analysis and information technology
Mariupol State University
Kiev, Ukraine
ORCID: 0000-0001-5171-6330

2st Yuriy Kyrychuk
department of automation and non-destructive testing systems
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kiev, Ukraine
ORCID: 0000-0001-8638-6060

3st Nataliia Nazarenko
department of automation and non-destructive testing systems
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kiev, Ukraine
ORCID: 0000-0001-6533-7323

***Abstract*—This work investigates the application of machine learning for anomaly detection in control systems, focusing on autoencoders as a tool for identifying and analyzing anomalies. The relevance of the research is driven by the need to enhance cybersecurity and the reliability of systems managing critically important production processes. The developed approach allows not only to determine the presence of an anomaly but also to identify key parameters contributing to its occurrence. The results demonstrate the effectiveness of using machine learning to improve the safety and reliability of automated systems. The contribution of this work lies in the development of a methodology for interpreting autoencoder data, providing a deep analysis of the causes of anomalies in technological processes.**

Keywords—statistical methods, machine learning, anomaly analysis, anomaly detection algorithms, security threat assessment, autoencoders

I. Introduction

In the modern world, processing large volumes of data is becoming increasingly important as data analysis plays a key role in decision-making and trend forecasting. The automation of data extraction and processing is one of the main factors of success in various fields. The growth in data volumes requires the use of complex analysis methods, where artificial intelligence, machine learning, Big Data, and cloud computing play a crucial role [1,2]. These technologies enable efficient data processing, pattern detection, and future trend prediction, helping companies to enhance competitiveness and improve product quality. However, as data volumes increase, so does the complexity of their processing, presenting new challenges in the areas of data storage, analysis, and security.

The performance of any system depends on its ability to operate within set parameter ranges, which may vary across different system sections or sensors. Key is the system's ability to function within these limits. Deviations indicating potential issues must be promptly detected and analyzed, whether they concern individual sensors or system sections [3]. Analyzing such deviations is crucial for system reliability and efficiency, involving identifying causes, assessing impact, and developing corrective measures to prevent future issues [4].

In the field of data analysis, there are various approaches to anomaly detection, including outlier detection and the search for novelty/out-of-distribution (OOD) detection [5]. Outliers are data points that stand out from the overall series and can be caused by noise or exhibit irregular behaviour [6]. The search for novelty and its subsequent study are critically important for the development of artificial intelligence (AI) algorithms, making them more informed and adaptable to new conditions.

Novelty in data refers to objects that differ from those in the training sample and emerge afterwards, making them crucial for detection by machine learning models [7]. These systems can identify anomalies, including both singular and collective anomalies, that exceed established norms [6]. However, limiting learning algorithms to only analyzed data can reduce their ability to adapt to novelty, potentially leading to incorrect classification of new data as anomalous [8]. Novelty can arise from various scenarios, such as equipment failures or cyber-attacks, and presents a challenge for prediction and identification due to its wide range of forms and contexts [9,10].

Effective novelty forecasting in data requires analyzing large volumes of diverse information, including historical data on anomalies. This task involves identifying known types of anomalies and adapting to new patterns, necessitating the creation of flexible machine learning models for continuous learning. The study highlights the relevance of anomaly detection, reviewing methods and approaches, including statistical methods and machine learning, their advantages and limitations. Challenges in the field and the prospects for developing new algorithms to tackle the increasing volume of complex data are also discussed.

II. Anomaly Detection Methods

Research requires data with clear criteria for normal behavior to train algorithms and test hypotheses. Anomaly detection, a key element in many areas, involves identifying deviations from expected behavior [11-13]. Figure 1 shows the methods and approaches to anomaly detection and their classification.

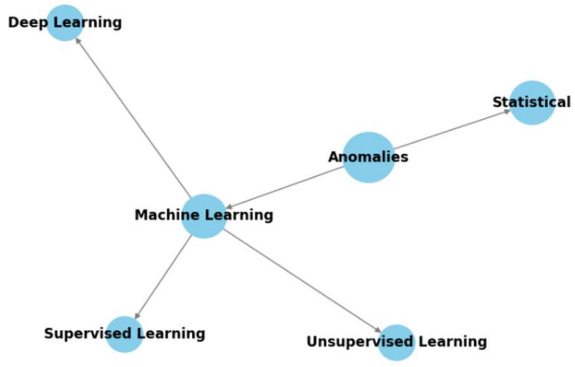

Fig. 1. Scheme of Anomaly Detection Methods.

Figure 2 presents a visual diagram that vividly demonstrates the various methods of detecting anomalies in data. The diagram includes the main approaches and algorithms used in the field of anomaly detection, providing a clear understanding of how these methods are classified and interact with each other. Let's consider some common methods of anomaly detection.

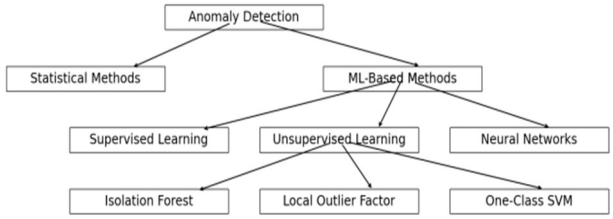

Fig. 2. Hierarchical Structure of Anomaly Detection Methods.

A. Statistical Methods

Statistical anomaly detection methods [14-15] are crucial for identifying deviations that may indicate problems, errors, or fraud, based on data's statistical characteristics. These methods assume most data follow a certain distribution, often Gaussian, identifying outliers as observations significantly diverging from this distribution. A common approach is statistical quality control [16], which analyzes production process characteristics to detect deviations, using indicators like mean, standard deviation, and variation coefficient. Let's consider some of them:

Z-score or standard deviation. This statistical tool allows assessing how far a value in a dataset deviates from the mean, expressed in units of standard deviation [15,17]. It is very useful for identifying outliers or anomalous values in data.

$$Z = \frac{(X-\mu)}{\sigma}, \quad (1)$$

where X - value in the dataset, μ - the mean (expected value) of all values in the dataset, σ - the standard deviation of all values in the dataset.

This method is especially effective in datasets where the distribution approximates a normal (Gaussian) distribution, and can be used for automated anomaly detection in large datasets across various applications.

The outlier test refers to a method using the interquartile range (IQR) [15] to determine and identify outliers in data. This method is widely used in statistical analysis for anomaly detection. The ACS method is particularly useful because it is resistant to the outliers themselves - the calculation of quartiles is not as heavily influenced by extreme values as, for example, the arithmetic mean. This makes IQR a preferred choice for analyzing data with anomalies. Let's take a closer look at how it works.

The ACS [15,18] is defined as the difference between the 75th percentile (third quartile, Q_3) and the 25th percentile (first quartile, Q_1) in a dataset. The formula for IQR:

$$IQR = Q_3 - Q_1 \quad (2)$$

Based on the IQR, boundaries for detecting outliers can be defined. Typically, the follow in grules are used [15]: the lower boundary is Q_1 - 1.5×IQR, and the upper boundary is Q_3 + 1.5×IQR. These boundaries help to identify values that are considered too low or too high compared to the main distribution of the data. Values that fall outside these boundaries are classified as outliers. This method is widely used in various fields for data preprocessing, to exclude or further investigate anomalous values before conducting further analysis.

Histograms play a key role in data analysis, allowing for the visualization of the distribution of values in a dataset [19]. The main idea of this method is to build a histogram that divides the data into a series of intervals or "bins," and shows the frequency with which values fall into each of these bins [20]. This provides valuable insights about data distribution trends, enabling easy identification of areas where data are densely grouped, as well as detecting potential anomalies or outliers that deviate from the main bulk of the data. The use of histograms is particularly useful for preliminary analysis when a quick overview of the data distribution character is needed. However, this method has its limitations. One of the drawbacks is that histograms do not consider the correlation between different variables in the dataset. This means that in the presence of complex relationships between variables, which may affect whether a specific observation is anomalous, histograms may not provide a complete picture of the situation. Additionally, the choice of the number and size of bins can significantly influence the perception of data distribution, requiring the analyst to carefully adjust these parameters to ensure the most accurate and informative visualization.

Control charts, developed by Walter A. Shuhart [21], are used for monitoring and analyzing processes. They help identify anomalies in data that differ from the usual behavior of the system, which may indicate the presence of safety threats or intrusions. In the context of cybersecurity, control charts are used to track network activity, access to system resources, or any other indicators that may evidence unauthorized activity or security breaches. The central line of the control chart reflects the average value of the parameter being analyzed [21], and the upper and lower control limits, usually set at a distance of ±3σ (where σ is the standard deviation), help determine when activity deviates from normal behavior [21]. Exceeding these limits may serve as a signal for further investigation to identify and mitigate potential threats.

Regression analysis represents a powerful statistical tool [15,22] that finds wide application in cybersecurity for modeling and predicting system behavior. Using the linear regression equation $y = ax + b$, where y is the dependent variable we try to predict, x is the independent variable based on which the prediction is made, a is the slope of the line indicating the amount of change in y for a change in x, and b is the y -intercept, it is possible to analyze and predict various security aspects [22].

In cybersecurity, regression analysis determines relationships between certain network traffic frequencies and security incidents, identifies correlations between incident response times and their system impacts, or predicts vulnerabilities based on current threat trends. This approach enables security experts to analyze vast data volumes, uncovering hidden patterns and trends for more effective attack prevention and risk management. Determining parameters a and b in the regression equation, based on

available data, facilitates quantitative assessments of relationships among various cybersecurity variables.

Statistical methods used for time series analysis play a crucial role in forecasting and detecting anomalies in various fields, including cybersecurity. Methods such as moving averages and exponential smoothing allow for analyzing and predicting trends based on historical data, which can be critically important for preventing or timely responding to attacks and threats in the field of information security.

However, the application of statistical methods in time series analysis encounters a number of limitations. In the context of cybersecurity, these limitations can affect the ability to promptly detect and respond to anomalous events or threats, requiring specialists to seek more advanced methods of analysis and monitoring. Therefore, it is important to integrate statistical methods with other approaches, such as machine learning and big data analysis, to increase the accuracy and efficiency of anomaly detection in modern technological and information systems.

B. Machine Learning Methods

Machine learning methods represent advanced approaches for detecting anomalies in technological processes and data, offering powerful tools for automatically identifying unusual patterns [23-24] that may indicate potential problems or threats. However, like any technology, they have their own characteristics and challenges. One of the key aspects of using machine learning for anomaly detection is that, although algorithms can effectively identify anomalous data [24], an important issue remains understanding why specific observations were classified as anomalous. This is because many machine learning models act as a "black box," especially in the case of complex algorithms such as deep learning. Understanding the influence of specific parameters on the results of the model can be difficult without additional analysis.

Furthermore, tuning the model parameters is a critically important stage in the process of creating an effective anomaly detection system. Model parameters selected manually may require adjustments depending on the specifics of the data and the objectives of anomaly detection, which can become a challenge without sufficient experience and knowledge in the field of machine learning.

In the context of cybersecurity and managing technological processes, using machine learning for anomaly detection offers significant advantages, including the ability to quickly process large data volumes and identify complex threats. However, the success of these systems largely depends on the correct choice of algorithms, parameter settings, and constant adaptation to changing conditions.

Classification algorithms, such as One-Class SVM and Isolation Forest, are popular tools in the field of machine learning for detecting anomalies in data [25-26]. They offer unique approaches to identifying unusual patterns that may indicate significant deviations in the data or potential threats. Let's examine them in more detail:

One-Class Support Vector Machines (One-Class SVM) [26] is a specialized machine learning algorithm used to determine whether new data is anomalous or normal. This is achieved by training the algorithm on data considered "normal" with the goal of isolating these data into a separate class. Successful operation of One-Class SVM requires careful data preparation and the selection of an appropriate kernel function so that the model can effectively separate "normal" data from potential anomalies, defining the separation boundary.

Isolation Forest is an anomaly detection algorithm [26] based on the concept of isolating data points using an ensemble of isolation trees. It is effective in detecting anomalies because anomalous data is usually easier to isolate from the rest of the data. The algorithm randomly selects features and splits the data, building trees until each data point is isolated. Anomalous data typically gets isolated with fewer splits, resulting in a shorter path in the tree. Isolation Forest is well-suited for working with data containing noise or multiple clusters where traditional outlier detection methods might fail.

Clustering algorithms play a crucial role in anomaly detection by allowing the grouping of data based on their similarities and highlighting those that do not conform to common patterns. Let's take a closer look at two popular clustering algorithms: k-Nearest Neighbors (k-NN) and DBSCAN [27].

The k-NN method assumes data of the same class are closer together, while anomalies are far from most normal data. For each object, the algorithm calculates the distance to its k nearest neighbors. If the distance to the neighbors exceeds a threshold, the object may be an anomaly. k-NN is simple to implement and can be effective for small datasets. However, its performance may degrade for high-dimensional data or complex structures due to the "curse of dimensionality." The method also requires choosing the optimal number of neighbors and threshold, which may not be trivial.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that groups points based on their density connectivity [28] and is capable of detecting clusters of arbitrary shapes as well as separating noise and anomalies. The algorithm starts with a random point, identifies all points in its "neighborhood" based on specified parameters (eps and MinPts), and, if the point has enough neighbors, it becomes part of the cluster. The process is repeated until all points are visited. DBSCAN is excellent for data with noise [28] and can detect anomalies as outliers not belonging to any cluster. However, the algorithm may face challenges when dealing with data where clusters vary in density since choosing global parameters for eps and MinPts may not simultaneously suit all clusters. Additionally, the algorithm can be computationally expensive for very large datasets.

Algorithmic methods are important tools for finding interesting patterns and associations between different elements in large datasets. They are particularly useful in anomaly detection tasks, allowing the identification of unusual or infrequently occurring combinations of data that may indicate anomalies. Let's delve into three mentioned algorithms: Apriori, FP-Growth, and ECLAT [29-30].

The Apriori algorithm iteratively discovers frequently occurring sets of elements, starting with one-element sets and gradually increasing the size of sets. At each step, the algorithm uses sets found in the previous step to generate new candidate sets of larger size [31]. It then determines their frequency of occurrence and filters out sets whose frequency is below a specified threshold.

FP-Growth (Frequent Pattern Growth) uses a tree structure to efficiently store frequently occurring combinations [29-30]. It compresses the database into an FP tree, which is then used to generate frequent items without the need to generate candidates.

ECLAT (Equivalence Class Clustering and bottom-up Lattice Traversal) operates by transitioning from a vertical to a horizontal data representation, utilizing the concept of equivalence classes [29-30]. It searches for associative rules by performing a bottom-up traversal of the lattice of nested sets, enabling efficient identification of frequently occurring items. The advantage of ECLAT lies in its ability to process data efficiently without repeated scanning and with minimal memory requirements. It is faster than Apriori but may be less efficient compared to FP-Growth when dealing with very large datasets.

Unsupervised learning algorithms are pivotal for anomaly detection, enabling data analysis without predefined labels. Among such algorithms, Principal Component Analysis (PCA) and the Local Outlier Factor (LOF) [31] hold special significance, each with its applications and limitations. The PCA transforms original data features into orthogonal principal components [31]. These components are ordered so that the first few capture most of the variance in the data. When using PCA for anomaly detection, the extent to which data deviates from the first few principal components is analyzed. Large deviations may indicate anomalies. The Mahalanobis distance is used to measure the degree of deviation of each data point from the model constructed using principal components. If this distance for a point exceeds a predefined threshold, the point may be considered anomalous. The LOF assesses local data density [31]. It compares the density of each object's surroundings with the densities of its neighbors. Objects with significantly lower density than their neighbors are considered outliers. LOF is well-suited for detecting anomalies in data with varying density and complex structure, where anomalous points may not be as explicitly delineated [32].

Neural networks are powerful tools in the field of machine learning, especially when it comes to anomaly detection in data [33]. Their ability to learn and generalize complex patterns makes them ideal for identifying deviations that may indicate potential issues or unusual states in various processes and systems. Let's explore the main types of neural networks used for anomaly detection:

Autoencoder (AE) is trained to reconstruct input data after compressing it into a smaller hidden space [34]. The efficiency of signal reconstruction on new, normal data is high, while for anomalous data, the reconstruction results are typically poorer, enabling anomaly detection. Careful selection of hyperparameters and training on normal data are required for accurate anomaly detection.

LSTM (Long Short-Term Memory) networks are capable of processing and remembering information over long periods [34], making them ideal for analyzing time series and detecting anomalies based on historical data. They are particularly useful in conditions of unbalanced and dependent time series, where it's crucial to account for both short-term and long-term dependencies.

GAN (Generative Adversarial Network) are trained to generate data similar to the normal data, while a discriminator tries to distinguish between real and generated data [35]. Anomalies can be detected by analyzing how well new data can be reproduced or differentiated by the network. They are suitable for anomaly detection by creating data that differs from the learned distribution of normal data.

SOM (Self-Organizing Map) are used for clustering data in a lower-dimensional space while preserving the topological properties of the original space [35]. Anomalies can be identified by analyzing the distances between nodes and clusters. They are effective for data visualization and clustering, which allows for the detection of anomalies as points that deviate from common clusters.

The use of neural networks for anomaly detection offers a flexible and powerful approach, capable of adapting to various types of data and application scenarios. It's important to emphasize that successful application of these methods requires a deep understanding of the data specifics and process, as well as careful selection and tuning of model parameters. Combining different approaches can enhance the effectiveness of anomaly detection, providing a more reliable and accurate identification of potential problems.

III. Result and Discussion

Automated control systems (ACS) play a key role in various industries, enhancing efficiency and reducing costs. However, their increasing use raises vulnerabilities to cyberattacks that can have serious consequences. Vulnerabilities may arise from software errors, architectural flaws, and unauthorized access, making system components targets for attacks. A particular problem is the inadequate protection of network protocols, many of which are outdated in terms of cybersecurity, allowing malicious actors to intercept and modify data, potentially disrupting equipment operation and management processes. ACS connected to corporate networks or the internet for remote access increase the risk of cyberattacks. Vulnerabilities in network infrastructure can be exploited to affect operations. Even isolated ACS are at risk from data carriers or internal threats like malicious employees. A comprehensive approach combining technical and organizational measures is essential for ACS security.

This study proposes the use of Autoencoders (AEs) for detecting anomalies in technological processes, with a key aspect being not just the identification of anomalies but understanding their causes. AEs effectively detect anomalies by analyzing deviations of reconstructed data from the original. However, one of the primary tasks is to identify specific features or parameters that caused the anomalies. This requires the development of additional algorithms or techniques capable of analyzing and interpreting the AE's results to determine which changes in the data are critical.

One approach to addressing this issue involves applying machine learning or deep learning techniques that can analyze relationships between various parameters in the data and identify those most strongly associated with anomalies. For example, classification or clustering algorithms could be used to analyze data processed by an autoencoder to identify groups of features characteristic of abnormal states.

Another approach involves using Explainable Artificial Intelligence (XAI) techniques, which can help interpret the results obtained from complex deep learning models like AEs. XAI methods can provide insights into which features contribute most significantly to the model's decision-making

process, allowing for the identification of parameters associated with anomaly detection in the case of AEs.

Interpreting the results obtained from AEs in large-scale Industrial Control Systems (ICS) indeed poses a significant challenge due to the complexity and volume of data generated by numerous sensors and mechanisms. These systems can encompass thousands or even millions of different parameters, making the analysis of anomalies particularly challenging for understanding and interpretation. Using simulated data to assess anomaly detection systems, especially in cybersecurity and industrial monitoring, is crucial. It helps refine models before applying them to complex real-world data. Autoencoders, trained on normal behavior data without anomalies, effectively detect deviations when exposed to real data. This approach reduces false positives by learning the characteristics of normal behavior. After training on simulated data, the model's quality assessment occurs by checking its ability to accurately reconstruct input data. This step is critical to ensure that the model is sensitive enough to anomalies without losing the ability to correctly interpret normal data.

Next, applying the model to real data opens the possibility for the practical use of AEs in anomaly detection. Here, the key point is the analysis and interpretation of the results that the model provides when encountering potential threats or non-standard operating conditions of the system. Using a model trained on data without anomalies, with appropriate hyperparameters, allows for the precise identification of moments when deviations from usual behavior occur, which is the basis for detecting attacks or technical malfunctions.

The importance of this approach is amplified when comparing the model's predictions with data containing anomalies. This allows for an assessment of how effectively the model can detect real incidents, and how accurately it can predict potential threats based on its training. Calculating the errors or losses between the input data and its reconstruction by the model provides insight into the quality of data reconstruction and sensitivity to anomalies.

The structure of an autoencoder, which includes an encoder for compressing input data into a hidden representation and a decoder for reconstructing data from this representation, plays a key role in the training process and anomaly detection. By training to minimize the reconstruction error, it aims to reproduce the input data as accurately as possible, while being highly sensitive to any deviations that may indicate issues within the system.

The proposed approach to determining the contribution of each parameter to an anomaly when using AEs highlights the importance of understanding the dynamics of reconstruction errors in the context of multidimensional data. In the process of training the autoencoder, the focus is on minimizing the data reconstruction error, directing attention to the use of a loss function based on Mean Squared Error (MSE). The MSE loss function, denoted as L_{mse}, plays a crucial role in optimizing the training process of the autoencoder, providing a mechanism for assessing the quality of the model's data reconstruction.

The formula for L_{mse} is as follows:

$$L_{mse} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2 \quad (3$$

where n is the total number of parameters (dimensions) in the data, x_i represents the true value of the *i*-th parameter, and $\hat{x}_i$ is the predicted value of the *i*-th parameter, reconstructed by the autoencoder.

Based on the assumption that an anomaly might be caused by only a small subset of all parameters, it's crucial to identify which specific parameters contribute most significantly to the anomaly. The introduction of a vector g, indicating the degree of contribution of each parameter to the anomaly, allows for more detailed analysis of how changes in parameter values affect the anomaly assessment by the model. If the value of a parameter increases relative to its normal state, the corresponding element in vector g becomes positive, and vice versa.

The minimization task proposed for determining the degree of contribution of each parameter to the anomaly aims to reduce the MSE, which, in turn, should lower the overall anomaly score provided by the autoencoder as the parameter values approach their normal operating values. This is suitable for filtering out those parameters that are truly important for detecting and interpreting anomalies, reducing the likelihood of misinterpretation due to the influence of anomalies on reconstruction errors across all measurements. Thus, this approach not only improves the accuracy of anomaly detection but also promotes a deeper understanding of the contribution of individual parameters to the emergence of anomalous situations.

Interpreting the operation of an autoencoder in the context of anomaly detection represents a key aspect when analyzing data from complex control systems. An autoencoder trained on training data without anomalies incorporates architectural components such as an input layer, several fully connected layers, normalization layers, and activation functions. This ensures its ability to efficiently learn and generate accurate reconstructions of input data. Minimizing the mean absolute error and using the MSE as a metric to assess the quality of the reconstruction, along with the application of early stopping, helps prevent overfitting.

To interpret the results of the autoencoder and identify features that most influence anomalies, the gradient descent method is applied. This method allows calculating how changing each input feature affects the reconstruction error, which is valuable information for localizing and understanding the causes of anomalies. After training on data in a normal state, the weights in the autoencoder are frozen to investigate the model's response to data corresponding to the moment of an attack.

The interpretation process involves feeding a portion of the test data representing the moment of attack into the trained autoencoder, after which the MSE between the reconstructed outputs and the original inputs is calculated. Then, using gradient descent, those input features whose change most significantly affects the reconstruction error are identified. This approach not only helps to detect anomalies but also allows understanding which specific changes in the data are most significant for their occurrence. Such an approach can be particularly useful for diagnosing and preventing future attacks in cybersecurity systems.

Thus, the use of AEs in cybersecurity represents a complex but extremely promising approach to ensuring the safety and reliability of industrial systems. It allows not only to detect potential threats but also to analyze their nature to develop

effective measures to prevent attacks and technical malfunctions in the future.

IV. Conclusions

During the study, it was found that anomalies can occur anytime and anywhere in the process, simultaneously affecting various system stages. This significantly complicates their detection and localization. Moreover, the technological process can be disrupted not only by attacks from malicious actors but also by other reasons, such as equipment malfunctions, incorrect settings of sensors and equipment parameters, and changes in parameters caused by force majeure circumstances. All these events negatively impact the production process, highlighting the relevance of applying machine learning methods for timely detection of anomalies and determining their causes. The use of machine learning allows not only detecting anomalies in real-time but also predicting potential threats, minimizing the risk of critical situations and ensuring reliable protection of production processes.

Acknowledgment

We would like to extend my heartfelt gratitude to the Department of System Analysis and Information Technology at Mariupol State University. Their unwavering support, invaluable guidance, and extensive resources have been instrumental in the success of my work.

References

[1] S. Mishra and A. K. Tyagi, "The Role of Machine Learning Techniques in Internet of Things-Based Cloud Applications," in Artificial Intelligence-Based Internet of Things Systems, 2022, pp. 105-135.

[2] A. A. A. Gad-Elrab, "Modern Business Intelligence: Big Data Analytics and Artificial Intelligence for Creating the Data-Driven Value," in E-Business-Higher Education and Intelligence Applications, vol. 135, 2021.

[3] V. Chatzigiannakis and S. Papavassiliou, "Diagnosing Anomalies and Identifying Faulty Nodes in Sensor Networks," IEEE Sensors Journal, vol. 7, no. 5, pp. 637-645, 2007.

[4] R. K. Pearson, "Outliers in process modeling and identification," IEEE Transactions on Control Systems Technology, vol. 10, no. 1, pp. 55-63, Jan. 2002, doi: 10.1109/87.974338.

[5] H. Hojjati, T. K. K. Ho, and N. Armanfard, "Self-supervised anomaly detection in computer vision and beyond: A survey and outlook," Neural Networks, vol. 2024, no. 106106, 2024.

[6] D. Velasquez et al., "A Hybrid Machine-Learning Ensemble for Anomaly Detection in Real-Time Industry 4.0 System," IEEE Access, vol. 10, pp. 72024-72036, 2022.

[7] G. Kim et al., "Open-World Continual Learning: Unifying Novelty Detection and Continual Learning," 2023. [Online]. Available: https://www.researchgate.net/publication/370152933_Open-World_Continual_Learning_Unifying_Novelty_Detection_and_Continual_Learning.

[8] S. Ahmad et al., "Unsupervised real-time anomaly detection for streaming data," Neurocomputing, vol. 262, pp. 134-147, 2017.

[9] L. Martí et al., "Anomaly detection based on sensor data in petroleum industry applications," Sensors, vol. 15, no. 2, pp. 2774-2797, 2015.

[10] M. A. Pimentel et al., "A review of novelty detection," Signal Processing, vol. 99, pp. 215-249, 2014.

[11] A. Patcha and J. M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," Computer Networks, vol. 51, no. 12, pp. 3448-3470, 2007.

[12] S. Thudumu et al., "A comprehensive survey of anomaly detection techniques for high dimensional big data," Journal of Big Data, vol. 7, 2020, Art. no. 1-30.

[13] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Computing Surveys (CSUR), vol. 41, no. 3, Article 15, 2009.

[14] T. Stibor, J. Timmis, and C. Eckert, "A comparative study of real-valued negative selection to statistical anomaly detection techniques," in Proc. 4th Int. Conf. on Artificial Immune Systems (ICARIS 2005), Banff, Alberta, Canada, Aug. 2005, vol. 4, pp. 262-275.

[15] P. J. Rousseeuw and M. Hubert, "Anomaly detection by robust statistics," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 8, no. 2, e1236, 2018.

[16] D. Selvamuthu and D. Das, Introduction to Statistical Methods, Design of Experiments and Statistical Quality Control. Singapore: Springer Singapore, 2018.

[17] E. J. Jamshidi et al., "Detecting outliers in a univariate time series dataset using unsupervised combined statistical methods: A case study on surface water temperature," Ecological Informatics, vol. 69, 101672, 2022.

[18] H. P. Vinutha, B. Poornima, and B. M. Sagar, "Detection of outliers using interquartile range technique from intrusion dataset," in Proc. 6th Int. Conf. on FICTA, Singapore, 2018, pp. 511-518.

[19] P. Mishra et al., "Detecting network anomalies using machine learning techniques: A comparative study," in Proc. Int. Conf. on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020, pp. 1-7.

[20] L. M. Manevitz and M. Yousef, "One-class SVMs for document classification," Journal of Machine Learning Research, vol. 2, pp. 139-154, Dec. 2001.

[21] T. Fawcett and F. Provost, "Adaptive fraud detection," Data Mining and Knowledge Discovery, vol. 1, no. 3, pp. 291-316, 1997.

[22] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," ACM Computing Surveys (CSUR), vol. 46, no. 4, Article 44, 2014.

[23] A. Lazarevic et al., "A comparative study of anomaly detection schemes in network intrusion detection," in Proc. of the 2003 SIAM Int. Conf. on Data Mining, San Francisco, CA, 2003, pp. 25-36.

[24] R. Aggarwal and K. Subbian, "Data clustering algorithms and applications," in Data Clustering: Algorithms and Applications, Boca Raton, FL, USA: CRC Press, 2013, ch. 1, pp. 1-29.

[25] G. Shu Fuhnwi, V. Adedoyin, and J. O. Agbaje, "An Empirical Internet Protocol Network Intrusion Detection using Isolation Forest and One-Class Support Vector Machines," 2023.

[26] N. Seliya, A. Abdollah Zadeh, and T. M. Khoshgoftaar, "A literature review on one-class classification and its potential applications in big data," Journal of Big Data, vol. 8, no. 1, pp. 1-31, 2021.

[27] A. Putina and D. Rossi, "Online anomaly detection leveraging stream-based clustering and real-time telemetry," IEEE Transactions on Network and Service Management, vol. 18, no. 1, pp. 839-854, 2020.

[28] A. Toshniwal, K. Mahesh, and R. Jayashree, "Overview of anomaly detection techniques in machine learning," in 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), October 2020, pp. 808-815.

[29] J. Heaton, "Comparing dataset characteristics that favor the Apriori, Eclat or FP-Growth frequent itemset mining algorithms," in SoutheastCon 2016, March 2016, pp. 1-7.

[30] V. Srinadh, "Evaluation of Apriori, FP growth and Eclat Association rule mining algorithms," International Journal of Health Sciences, vol. II, pp. 7475-7485, 2022.

[31] " M. Goldstein and S. Uchida, "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data," PloS one, vol. 11, no. 4, e0152173, 2016.

[32] M. M. Breunig and H. P. Kriegel, "LOF: Identifying Density-Based Local Outliers," ACM SIGMOD Record, vol. 29, pp. 93-104, 2000.

[33] A. B. Nassif et al., "Machine learning for anomaly detection: A systematic review," IEEE Access, vol. 9, pp. 78658-78700, 2021.

[34] E. Mushtaq et al., "A two-stage intrusion detection system with auto-encoder and LSTMs," Applied Soft Computing, vol. 121, 108768, 2022.

[35] I. H. Sarker, "Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective," SN Computer Science, vol. 2, no. 3, 154, 2021.