Article

Anomaly Detection of Metallurgical Energy Data Based on iForest-AE

Zhangming Xiong [1], Daofei Zhu [1,*], Dafang Liu [2], Shujing He [1] and Luo Zhao [1]

[1]   Faculty of Metallurgical and Energy Engineering, Kunming University of Science and Technology, Kunming 650093, China
[2]   Chuxiong Dianzhong Nonferrous Metals Co., Ltd., Chuxiong 675000, China
*   Correspondence: zdf_kmust@126.com

Abstract: With the proliferation of the Internet of Things, a large amount of data is generated constantly by industrial systems, corresponding in many cases to critical tasks. It is particularly important to detect abnormal data to ensure the accuracy of data. Aiming at the problem that the training data are contaminated with anomalies in autoencoder-based anomaly detection, which makes it difficult to distinguish abnormal data from normal data, this paper proposes a data anomaly detection method that combines an isolated forest (iForest) and autoencoder algorithm. In this method (iForest-AE), the iForest algorithm was used to calculate the anomaly score of energy data, and the data with a lower anomaly score were selected for model training. After the test data passed through the autoencoder trained by normal data, the data whose reconstruction error was larger than the threshold were determined as an anomaly. Experiment results on the electricity consumption dataset showed that the iForest-AE method achieved an F1 score of 0.981, which outperformed other detection methods, and a significant advantage in anomaly detection.

Keywords: metallurgical energy data; data anomaly detection; iForest algorithm; autoencoder algorithm

1. Introduction

The metallurgical industry is an essential component of the manufacturing industry and an essential support for maintaining world economic growth [1]. However, since almost all equipment in metallurgical enterprises usually lie in harsh environments [2], data collection is susceptible to external interference [3]. Sensor failures, network interruptions, and electromagnetic interference can also cause abnormalities in the collected data, thus affecting the modeling and analysis of the data. This also leads to incorrect decisions and guidance for the field dispatchers of metallurgical enterprises, causing serious safety accidents and irreparable economic losses.

Data anomaly detection plays a pivotal role in data management. The detection and diagnosis of key energy data in metallurgical enterprises are necessary for the enterprise's safe production and scientific dispatch. Therefore, developing a set of anomaly detection algorithms for data collected by a metallurgical enterprise information system is of great significance in ensuring the safe production of metallurgical enterprises and improving the economic benefits of enterprises.

As one of the important research directions of machine learning [4], anomaly detection has become a hotspot and been applied in many important areas. According to the different ideas of anomaly detection, it can be broadly categorized into: statistical-based [5], nearest neighbor-based [6], clustering-based [7], and iForest-based [8] methods. Wang [9] proposed an improved KNN algorithm for log data anomaly detection. He used an existing mean-shift clustering algorithm to select a training set from log data, and assigned various weights to samples with various distances, which lessened the negative effect of an imbalanced distribution of the log samples. Vanem [10] came up with a clustering-based anomaly detection method. The idea of the method was to identify clusters in sensor data under

normal operating situations and determine if new observations belong to any of these clusters. Experimental results showed that cluster-based methods performed better than other methods. LI [11] developed an GA-iForest method for numerical data anomaly detection. This method improved the accuracy of anomaly detection by removing some duplicate, similar, and poor detection isolation trees. Experimental results indicated that the proposed method could not only increase the accuracy of anomaly detection, but also decrease the number of isolation trees. However, the above methods have randomness and a lack of robustness, and it is easy to cause a dimensional disaster in high-dimensional data space.

In recent years, with the wide application of machine learning in speech recognition [12], machine translation [13], and data analysis [14], some machine learning models have also been gradually applied to the research of data anomaly detection [15]. Among various anomaly detection models based on machine learning, the autoencoder shows excellent performance due to its strong generalization and no dimensional disaster. Borghesi et al. [16] proposed an autoencoder-based approach for high performance computing systems anomaly detection and it can be used to distinguish abnormal conditions from normal conditions by learning the normal state of the supercomputer nodes. Chen et al. [17] built a model that projects the training data into a lower-dimensional subspace during the training phase, then optimized weights to minimize reconstruction error. It also used the reconstruction error of test data to determine whether the wireless transmission of data is abnormal. Although many attempts have been made, the performance of anomaly detection remains low, especially when the training data are contaminated with anomalies. It is difficult to distinguish abnormal data from normal data because the reconstruction error between the two is not very large.

To solve this problem, a data anomaly detection method by combining the iForest and autoencoder algorithm is proposed in this paper. This model uses the actual energy data of a copper smelting enterprise in southwest China to verify the effectiveness and competitiveness of this method.

2. Related Work

An autoencoder algorithm is one of the most effective algorithms for anomaly detection at present, but it is affected by the training set. If the training set is contaminated with anomalies, the detection effect will be poor, while the iForest algorithm can roughly screen out normal data to train the autoencoder model to solve this problem.

2.1. iForest Algorithm

The iForest algorithm is an anomaly detection algorithm proposed by Mr. Zhou in 2008 [18]. The algorithm first uses a random hyperplane to cut the data space, and each time it is cut, two subspaces are generated. The subspace is repeatedly cut using the random hyperplane until there is only one data point remaining in each subspace. Since a data set with high density requires numerous cuts to stop, the point with a low density will stay in a subspace very early after cutting [19], so the low density point is the abnormal point.

There are multiple trees in the iForest and the tree in the algorithm is called isolated tree(iTree) [20]. The following is the process of establishing the iTree.

1. Randomly select sub samples from training data as root nodes in trees.
2. Arbitrarily select a dimension to generate a cut point p in the current node data.
3. The cut point p separates the current node data space into two subspaces. The data that are less than the cut point are placed in the left subtree of the current node, and the data greater than or equal to the cut point are placed in the right subtree of the current node.
4. Perform steps 2 and 3 continuously to generate new sub nodes until the iTree reaches a finite height or there is only one data on the child node.

Then, the training data are detected by the generated iForest. $h(x)$ is the number of edges that the sample point x traverses from the root node of the iTree to the leaf nodes. Given a dataset with n samples, the tree's average path length is:

$$c(n) = 2H(n-1) - 2(n-1)/n \tag{1}$$

$H(i)$ is the harmonic function and it can be estimated by $In(i) + 0.5772$. $c(n)$ is the average of the path lengths for a given number of samples n, used to normalize the path length $h(x)$. The anomaly score of the sample x is calculated as:

$$s(x,n) = 2^{\frac{E(h(x))}{c(n)}} \tag{2}$$

$E(h(x))$ is the mathematical expectation of the path length of the sample x in a batch of iTree, from which the following inferences can be drawn:

When $E(h(x)) \to 0$, $s \to 1$, the probability of x being an outlier is high.

When $E(h(x)) \to c(n)$, $s \to 0.5$, whether x is an outlier is uncertain.

When $E(h(x)) \to n-1$, $s \to 0$, the probability of x being normal is high.

2.2. Autoencoder Algorithm

Autoencoder is a neural network whose training goal is to minimize the reconstruction error between the output and input [21]. It is frequently utilized in feature extraction [22], image classification [23], and noise reduction [24]. Autoencoder is also commonly used in data compression. It is made up of two components: encoder network and decoder network. An autoencoder can encode data X into a low-dimensional hidden vector Z, and a decoder can reconstruct the hidden vector into data X'. The following is its mathematical formula.

$$Z = \sigma_e(WX + b) \tag{3}$$

$$X' = \sigma_d(W'Z + b') \tag{4}$$

σ_e and σ_d are the activation functions for the encoder and decoder. W and b are respectively the weight and bias.

Since the autoencoder is trained on several normal data, its aim is to minimize the reconstruction error [25], so the trained autoencoder has a good ability to reconstruct normal data, but insufficient ability for anomalous data because it has not been trained. Considering that the reconstruction error of normal data will be smaller while those of anomaly data will be relatively larger [26], the reconstruction error can be used to distinguish whether it is an abnormality. The calculation formula of the reconstruction error is shown in Formula 5 below.
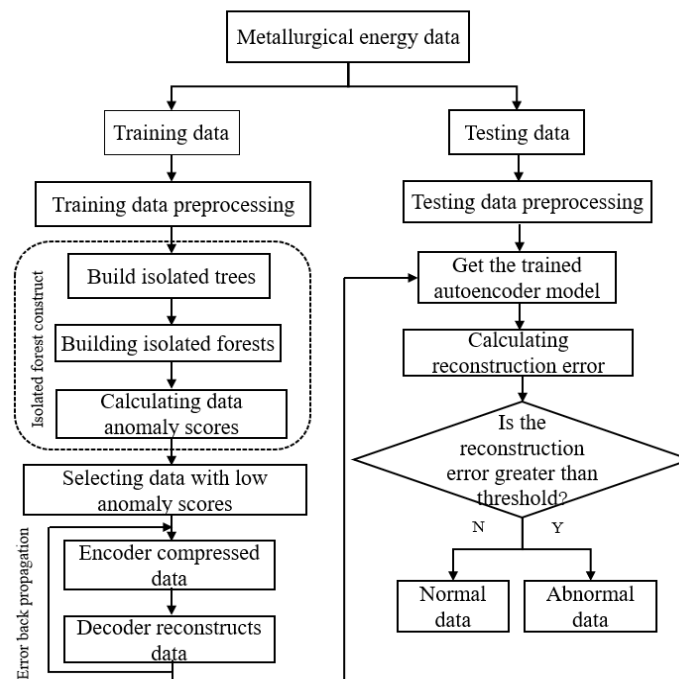
$$e = X - X' \tag{5}$$

2.3. Anomaly Detection Process Based on the iForest-AE Method

The anomaly detection process based on the AE-iForest is shown in Figure 1, which is separated into two phases: model training and model detection.

The data collected by the metallurgical information collection system are divided into two parts: training data and test data. In order to speed up the convergence of the model, the two parts of data are normalized by the following equation [27].

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{6}$$

x' is the normalized data, x is the input data, x_{max} and x_{min} are the maximum and minimum values of the input data.

Figure 1. The anomaly detection process based on the AE-iForest.

2.3.1. Model Training

The model training phase is separated into two parts: calculating the data anomaly score and training the autoencoder. Firstly, iTree is constructed according to the above rules; then, iForest is constructed by the iTree. For each training data, the anomaly score is calculated by Formula (2), and the data with a lower anomaly score are selected for model training to avoid the training data being anomaly-contaminated.

The training performance of the model is evaluated by root mean square error (RMSE). The smaller its value, the closer the reconstructed data are to the original data, and the better the training effect of the model. It is calculated as.

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \left(X_k - X'_k\right)^2} \tag{7}$$

In the training phase, the model is only trained using normal data. Additionally, the autoencoder iteratively updates the weight and bias parameters of each hidden layer by minimizing the reconstruction error. As a result, by several trainings, the model can capture the normal pattern of the data [28].

2.3.2. Data Detection

In the detection phase, the test data are fed into the autoencoder model and the reconstruction error of these test data is calculated. Additionally, the test data whose reconstruction error is larger than the threshold are determined as an anomaly.

In the anomaly detection based on the autoencoder, the selection of the threshold is extremely critical. We selected several values as thresholds, and used the F1 score to assess the effect of anomaly detection. The threshold that corresponds to the maximum F1 score was selected as the final threshold.

3. Experiment and Result Analysis

3.1. Experiment Environment

The experimental platform was a Windows 10 host equipped with 3.60 GHz CPU and 8.00 GB of ram. All of the iForest-AE model code was programmed in python3.6.

3.2. Data Preparation

In this paper, we selected the power consumption data of a copper smelting enterprise in Yunnan Province from 1 June 2021 to 25 June 2021, including 7200 data as the training set, and finally we selected 6810 data with an abnormal score lower than 0.6 to train the autoencoder. A total of 620 data from 24 June 2021 to 25 June 2021 were selected as the validation set and 4900 data from 26 June 2021 to 10 July 2021 were selected as the test set.

3.3. Experimental Procedure and Result Analysis

The iForest algorithm was selected to calculate the anomaly score for 7200 metallurgical energy data. According to the characteristics of electric energy data of 1# workshop, the best parameters were selected by manual experience [29], and the results are shown in Table 1.

Table 1. Parameter settings for the iForest algorithm.

Parameter	Value
n_estimators	100
max_features	1
contamination	0.1
max_samples	auto
n_jobs	None

Figure 2 shows the result of calculating the anomaly score of partial data through the iForest algorithm. Among the 7200 training data, 684 data with an anomaly score higher than 0.60 were uncertain data. This part of the data was excluded, and the remaining part of the data was selected for model training.
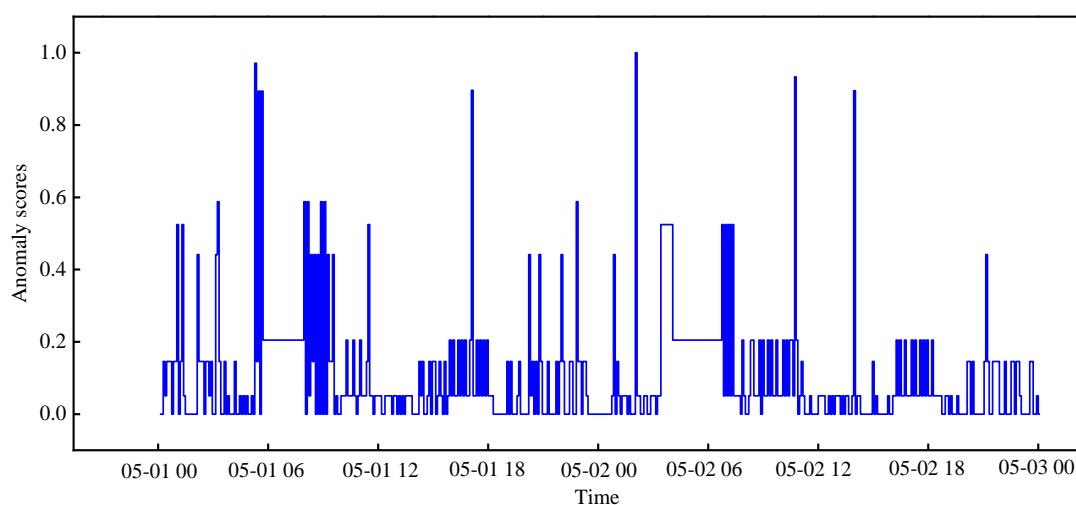

Figure 2. Anomaly scores of data.

The autoencoder was used to detect abnormality in metallurgical energy data. The random search algorithm [30] was used to find the best parameters, and the group of parameters with the minimum training error was taken as the final parameters. The results are shown in Table 2.

Table 2. Parameter settings for the autoencoder algorithm.

| Parameter	Values
Batch_size	128
Epoch	80
Loss	Mean square error
Optimizer	Adam

The training and validation sets were fed into the autoencoder network. Their loss leveled off at 1.976×10^{-4} and 2.684×10^{-4} after 80 training iterations, as shown in Figure 3, indicating that the trained model could effectively reconstruct normal data.
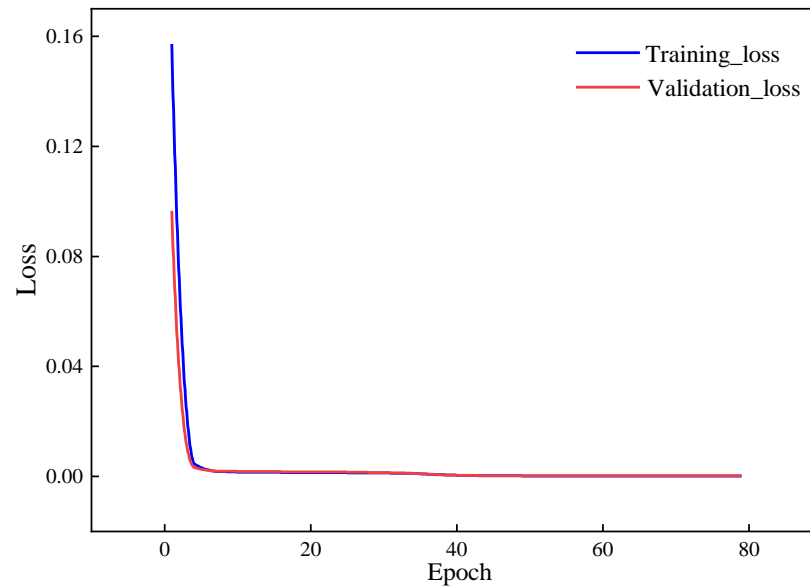
Figure 3. Training error of training set and validation set.

To evaluate the effect of anomaly detection, recall, precision, accuracy, and F1 score were used to evaluate the model. The F1 score was used for it is a comprehensive metrics.

Since the reconstruction error was used as the anomaly score, the selection of the threshold had a great influence on the detection result, which was related to the model detection accuracy. We selected several values as the threshold. For each value, five experiments were carried out and the average value of the F1 score of the five experiments was calculated. The threshold corresponding to the maximum F1 score was selected as the final threshold. The results of the experiments are shown in Figure 4.
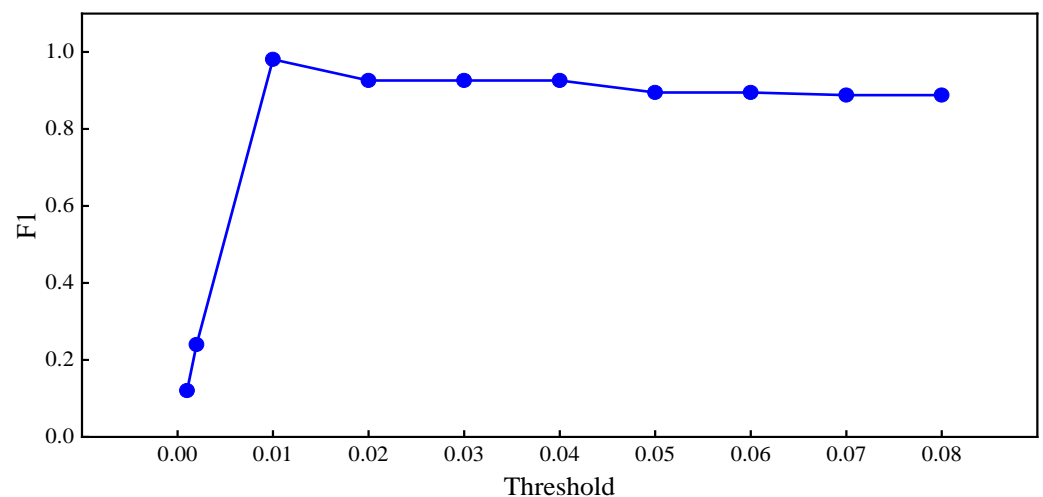

**Figure 4.** Relationship between threshold selection and detection effect.

It can be seen from the above experiments' results that when the threshold was 0.01, the F1 score obtained the maximum value of 0.981, and the anomaly detection effect was best at this time. When the threshold was too small, the detection effect was poor. The reason is that the threshold was too small, which caused the model to regard the normal reconstruction error fluctuation generated by the normal energy data as abnormal, so some

normal data were judged as abnormal data. When the threshold was too large, the detection effect of the model also decreased. The reason is that the threshold selection was too large, which relaxed the requirements for the reconstruction error of normal data. Therefore, abnormal data were determined as normal data.

In the test data, the electricity consumption data of metallurgical enterprise 1# workshop from 26 June 2021 to 12:00 on 28 June 2021 were taken as an example. After data normalization, the electricity consumption data were shown in Figure 5. The normal data were screened out by the iForest algorithm for model training. The model encoded and decoded those data by the autoencoder, and the reconstructed data are shown in Figure 6.
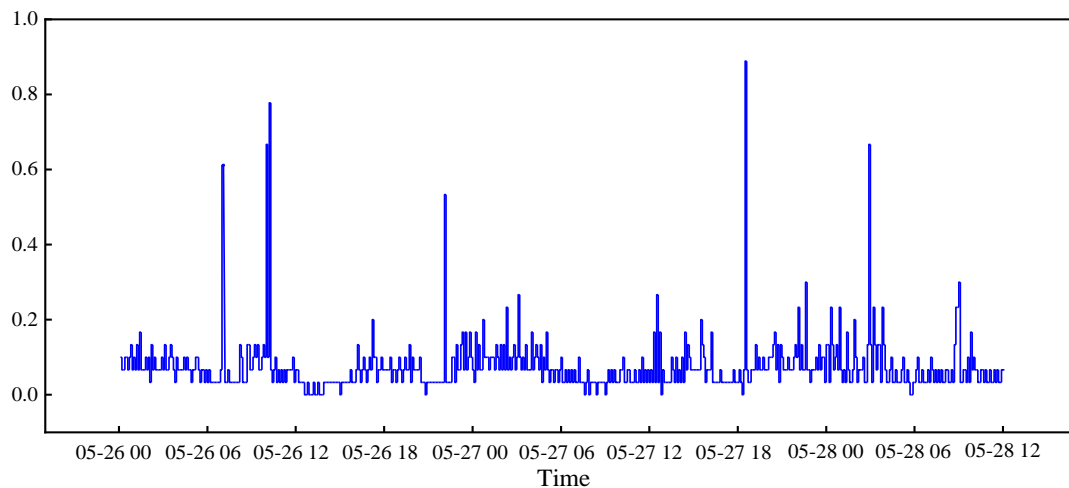

Figure 5. Electricity consumption of 1# workshop.
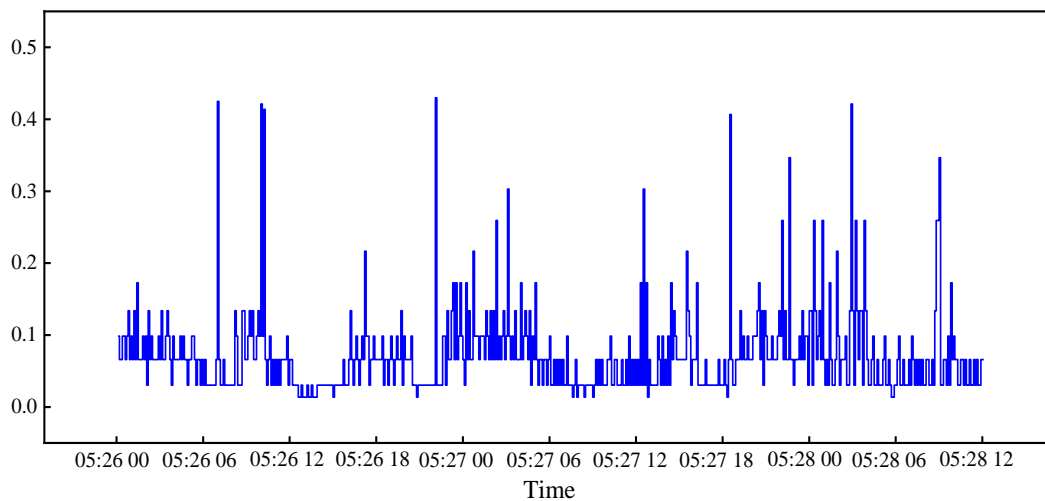
Figure 6. Electricity consumption reconstruction data of 1# workshop.

Normal electricity consumption data fluctuated in a low range. However, due to sensor failures and interruptions in field communication transmissions, the collected data did conform to the normal data distribution pattern, and the abnormal data had a large error after passing through the autoencoder trained by the normal data. The data with reconstruction error exceeding the threshold are marked with red dots, representing abnormal data. The detection results are shown in Figure 7.
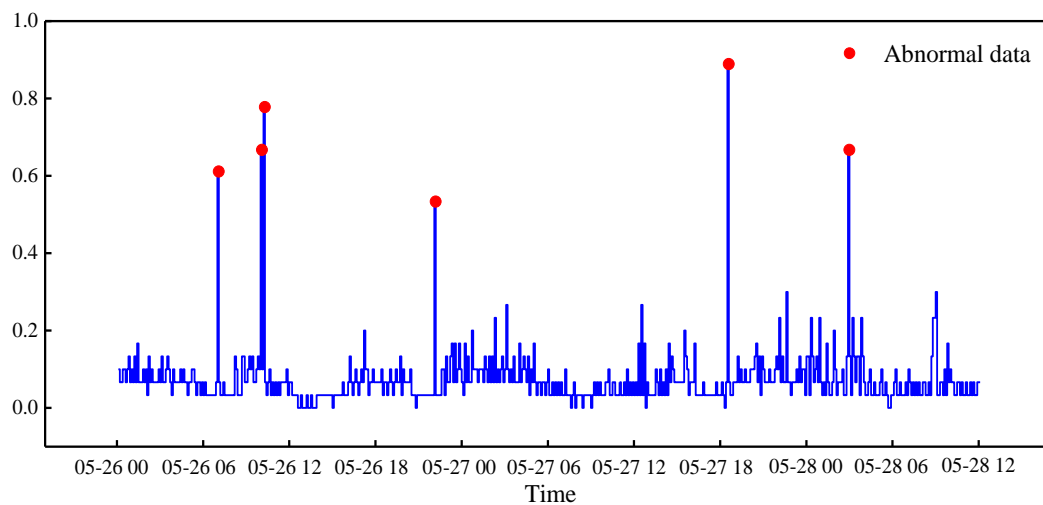
Figure 7. Anomaly detection results of 1# Workshop.

In order to test whether the training set screened by the iForest algorithm had an impact on the autoencoder-based anomaly detection, we trained the autoencoder model with the training set screened by the iForest algorithm and without the algorithm screening. The detection results are shown in Table 3.

Table 3. Comparison of test results.

Model	Accuracy	Precision	Recall	F1 Score
Autoencoder	0.989	1.000	0.799	0.888
iForest-AE	0.998	1.000	0.963	0.981

In the single autoencoder model, since the training set contains abnormal data, the reconstruction error of the abnormal data also became smaller, which made the model judge part of the abnormal data as normal data and fail to identify all the abnormal data.

To further analyze the effectiveness of the proposed method, iForest and SVM algorithms were used to perform experiments on the same data set. The experimental results are listed in Table 4 and illustrated in Figure 8.

Table 4. Experimental results of the three methods.

Model	Accuracy	Precision	Recall	F1 Score
iForest	0.975	0.789	0.723	0.755
SVM	0.922	1.000	0.858	0.924
iForest-AE	0.998	1.000	0.963	0.981

From Table 4 and Figure 8, it can be seen that the accuracy of this method was 2.359% and 0.604% higher than the SVM and iForest algorithms, indicating that the proposed algorithm had a good recognition effect for both abnormal and normal data. At the same time, the precision of this method also reached 1, illustrating that all the detected abnormal data were real abnormal data. The recall rate of this method reached 0.963, which was higher than the other two algorithms, showing that most abnormal data could be detected. Compared with the two methods, the F1 score was increased by 29.934% and 6.189%, proving that this method had a better comprehensive effect than the other two methods. To sum up, the proposed method had a high accuracy, precision, recall, and F1 score, and it was generally better than other algorithms. Accordingly, it achieved the state-of-the-art performance on metallurgical energy data anomaly detection.
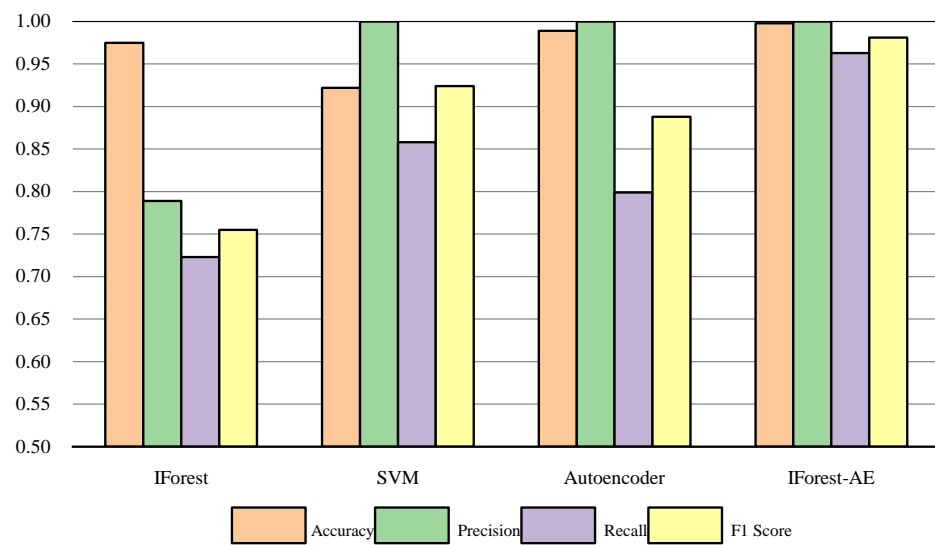
Figure 8. Comparison of different methods for anomaly detection.

In order to solve the problem that the training data is anomaly-contaminated by autoencoder-based anomaly detection, which makes it difficult to distinguish abnormal data from normal data, this paper proposed an anomaly detection method for metallurgical energy data that combined iForest and autoencoder. The electricity consumption data from 1# workshop of a copper smelting enterprise were used for verification, and the experiment result shows that the F1 score of the proposed method was as high as 0.981, which was 29.934%, 6.189%, and 10.473% higher than the iForest, SVM, and ordinary autoencoder.

4. Discussion

Autoencoder or iForest models separately have worse performance because training data are anomaly-contaminated. For the autoencoder-based anomaly detection model, since the training data are contaminated with anomalies, it also performs well in reconstructing anomalous data. Additionally, it is difficult to distinguish abnormal data from normal data because the reconstruction error between the two is not large. For the iForest-based anomaly detection model, the data were recursively divided, and the length of the path from data to the root node is chosen as the anomaly score. Since metallurgical energy data have a high proportion of abnormal data, the length of the path from abnormal data to the root node is similar to the normal ones, which will lead to a poor anomaly detection performance. The SVM-based anomaly detection algorithm is a supervised algorithm that needs to be carried out with a large amount of labeled data. It can behave well in detecting abnormal data that have been trained, while failing to do so with abnormal data that the model has not encountered.

The proposed algorithm first used the iForest algorithm to calculate the anomaly score of the data, and then selected the data with a lower anomaly score to train the autoencoder model, which avoided the worse detection effect caused by the abnormal data in the training set. The reconstruction error of normal data and abnormal data became larger, so it was easy to distinguish normal data from abnormal data.

5. Conclusions

In this paper, an anomaly detection method based on iForest-AE was proposed. The normal data were selected by the iForest algorithm to train the autoencoder model to improve the detection effect. It was applied in the data collection system of copper smelting enterprises in Yunnan Province because of its high detection accuracy and convenience. Due to the strong robustness and generalization of the proposed algorithm, it can also be applied to the data collection system of other enterprises such as mechanical manufacturing and chemical production to ensure the accuracy of the collected data.

In the research method of this paper, the parameters of the iForest and the autoencoder algorithm were selected through multiple experiments to achieve the best results. Future research is required to use some group optimization algorithms to find the best parameters to improve the model performance.

With the advent of the era of big data, the amount of data is increasing, and the deep iForest is often used to solve the corresponding problems. In order to reduce the time cost of the deep iForest algorithm, we need to make changes in the form of the isolated forest in the future.

References

1. Wu, W.; An, S.; Wu, C.; Tsai, S.; Yang, K. An empirical study on green environmental system certification affects financing cost of high energy consumption enterprises-taking metallurgical enterprises as an example. *J. Clean. Prod.* **2020**, *244*, 118848. [CrossRef]
2. Chuah, L.F.; Mokhtar, K.; Bakar, A.A.; Othman, M.R.; Osman, N.H.; Bokhari, A.; Mubashir, M.; Abdullah, M.A.; Hasan, M. Marine environment and maritime safety assessment using Port State Control database. *Chemosphere* **2022**, *304*, 135245. [CrossRef] [PubMed]
3. Chuah, L.F.; Mohd Salleh, N.H.; Osnin, N.A.; Alcaide, J.I.; Abdul Majid, M.H.; Abdullah, A.A.; Bokhari, A.A.; Jalil, E.E.; Klemeš, J.J. Profiling Malaysian ship registration and seafarers for streamlining future Malaysian shipping governance. *Aust. J. Marit. Ocean Aff.* **2021**, *13*, 225–261. [CrossRef]
4. Dogo, E.M.; Nwulu, N.I.; Twala, B.; Aigbavboa, C. A survey of machine learning methods applied to anomaly detection on drinking-water quality data. *Urban Water J.* **2019**, *16*, 235–248. [CrossRef]
5. Krawiec, P.; Junge, M.; Hesselbach, J. Comparison and adaptation of two strategies for anomaly detection in load profiles based on methods from the fields of machine learning and statistics. *Open J. Energy Effic.* **2021**, *10*, 37–49. [CrossRef]
6. Batchanaboyina, M.R.; Devarakonda, N. Design and evaluation of outlier detection based on semantic condensed nearest neighbor. *J. Intell. Syst.* **2019**, *29*, 1416–1424. [CrossRef]
7. Yang, Z. An efficient automatic gait anomaly detection method based on semisupervised clustering. *Comput. Intell. Neurosci.* **2021**, *2021*, 8840156. [CrossRef]
8. Li, C.; Guo, L.; Gao, H.; Li, Y. Similarity-measured isolation forest: Anomaly detection method for machine monitoring data. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–12. [CrossRef]
9. Wang, B.; Ying, S.; Cheng, G.; Wang, R.; Yang, Z.; Dong, B. Log-based anomaly detection with the improved K-nearest neighbor. *Int. J. Softw. Eng. Knowl. Eng.* **2020**, *30*, 239–262. [CrossRef]
10. Vanem, E.; Brandsæter, A. Unsupervised anomaly detection based on clustering methods and sensor data on a marine diesel engine. *J. Mar. Eng. Technol.* **2021**, *20*, 217–234. [CrossRef]
11. Li, K.X.; Li, J.; Liu, S.J.; Li, Z.; Bo, J.; Liu, B. GA-iForest: An efficient isolated forest framework based on genetic algorithm for numerical data outlier detection. *Trans. Nanjing Univ. Aeronaut. Astronaut.* **2019**, *36*, 1026–1038.
12. Lim, S.S.; Kwon, O.W. Frame augment: A simple data augmentation method for encoder–decoder speech recognition. *Appl. Sci.* **2022**, *12*, 7619. [CrossRef]
13. Xie, S.; Xia, Y.; Wu, L.; Huang, Y.; Fan, Y.; Qin, T. End-to-end entity-aware neural machine translation. *Mach. Learn.* **2022**, *111*, 1181–1203. [CrossRef]
14. Zhang, D.; Zhang, H.; Zhao, Y.; Chen, Y.; Ke, C.; Xu, T.; He, Y. A brief review of new data analysis methods of laser-induced breakdown spectroscopy: Machine learning. *Appl. Spectrosc. Rev.* **2022**, *57*, 89–111. [CrossRef]
15. Li, Y.; Xu, Y.; Cao, Y.; Hou, J.; Wang, C.; Guo, W.; Li, X.; Xin, Y.; Liu, Z.; Cui, L. One-class LSTM network for anomalous network traffic detection. *Appl. Sci.* **2022**, *12*, 5051. [CrossRef]

16. Borghesi, A.; Bartolini, A.; Lombardi, M.; Milano, M.; Benini, L. A semisupervised autoencoder-based approach for anomaly detection in high performance computing systems. *Eng. Appl. Artif. Intell.* **2019**, *85*, 634–644. [CrossRef]

17. Chen, Z.; Yeo, C.K.; Lee, B.S.; Lau, C.T. Autoencoder-based network anomaly detection. In Proceedings of the 2018 Wireless Telecommunications Symposium (WTS), Phoenix, AZ, USA, 17–20 April 2018; pp. 1–5.

18. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; pp. 413–422.

19. Zhang, W.; Chen, L. Web log anomaly detection based on isolated forest algorithm. In Proceedings of the 2019 IEEE 14th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Dalian, China, 14–16 November 2019; pp. 755–759.

20. Mao, W.; Cao, X.; Zhou, Q.; Yan, T.; Zhang, Y. Anomaly detection for power consumption data based on isolated forest. In Proceedings of the 2018 International Conference on Power System Technology (POWERCON), Guangzhou, China, 6–8 November 2018; pp. 4169–4174.

21. Ji, Y.; Lu, Z. The theoretical breakthrough of self-supervised learning: Variational autoencoders and its application in big data analysis. *J. Phys. Conf. Ser.* **2021**, *1955*, 012062. [CrossRef]

22. Zabalza, J.; Ren, J.; Zheng, J.; Zhao, H.; Qing, C.; Yang, Z.; Du, P.; Marshall, S. Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging. *Neurocomputing* **2016**, *185*, 1–10. [CrossRef]

23. Patel, H.; Upla, K.P. A shallow network for hyperspectral image classification using an autoencoder with convolutional neural network. *Multimed. Tools Appl.* **2022**, *81*, 695–714. [CrossRef]

24. Li, X.; Feng, S.; Hou, N.; Wang, R.; Li, H.; Gao, M.; Li, S. Surface microseismic data denoising based on sparse autoencoder and Kalman filter. *Syst. Sci. Control Eng.* **2022**, *10*, 616–628. [CrossRef]

25. Wang, H.; Liu, X.; Ma, L.; Zhang, Y. Anomaly detection for hydropower turbine unit based on variational modal decomposition and deep autoencoder. *Energy Rep.* **2021**, *7*, 938–946. [CrossRef]

26. Cheng, Z.; Wang, S.; Zhang, P.; Wang, S.; Liu, X.; Zhu, E. Improved autoencoder for unsupervised anomaly detection. *Int. J. Intell. Syst.* **2021**, *36*, 7103–7125. [CrossRef]

27. Seokheon, Y. Performance analysis of construction cost prediction using neural network for multioutput regression. *Appl. Sci.* **2022**, *12*, 9592. [CrossRef]

28. Zhou, H.; Yu, K.; Zhang, X.; Wu, G.; Yazidi, A. Contrastive autoencoder for anomaly detection in multivariate time series. *Inf. Sci.* **2022**, *610*, 266–280. [CrossRef]

29. Liang, D.; Wang, J.; Gao, X.; Wang, J.; Zhao, X.; Wang, L. Self-supervised Pretraining Isolated Forest for Outlier Detection. In Proceedings of the 2022 International Conference on Big Data, Information and Computer Network (BDICN), Sanya, China, 20–22 January 2020; pp. 306–310.

30. Bergstra, J.; Bengio, Y. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* **2012**, *13*, 281–305.