Academic Journal of Science and Technology
ISSN: 2771-3032 | Vol. 13, No. 2, 2024

Time-series Anomaly Detection Study Based on Deep Learning

Wenjuan Wu

School of Henan Polytechnic University, Jiaozuo 454000, China

Abstract: With the rapid development of deep learning technology, anomaly detection based on deep learning has become an important research direction. This paper classifies and summarizes anomaly detection methods, covering traditional methods, machine learning-based methods, and deep learning-based methods. It particularly introduces typical deep learning models, analyzing their respective advantages and disadvantages. Experiments were conducted on two public datasets, and the performance of each model in anomaly detection was compared in detail.

Keywords: Anomaly detection; Deep learning; Time series.

1. Introduction

Time series data is an ordered collection of data that changes over time, widely existing in various fields such as financial trading, industrial operations, environmental monitoring, and network traffic monitoring. In time series data, there are always some data that differ significantly from the overall characteristics of the dataset, which are known as anomalies. Timely detection of these anomalies can effectively reduce losses in industrial production and can even prevent the occurrence of some disasters. Therefore, research into anomaly detection has very important practical significance. Currently, research on time series anomaly detection mainly focuses on traditional statistical models and machine learning models, but there are issues such as low detection accuracy and weak generalization ability. With the rapid development of deep learning technology, researchers are gradually applying some deep learning algorithms, including Convolutional Neural Networks (CNN) [1] and Long Short-Term Memory networks (LSTM) [2], to anomaly detection in fields such as medical image processing [3], pest detection [4], and natural language processing [5].

2. Introduction to Anomaly Detection

2.1 Classification of Anomalies

Time series data is a collection of data points arranged in chronological order, with each time point corresponding to an observation. It is commonly used to describe events that change over time. In time series anomaly detection, an anomaly refers to individual data points that are significantly deviant from the sample dataset and lie outside a specific range. Anomalies are generally classified into three types: point anomalies, contextual anomalies, and collective anomalies:

(1) Point anomalies: These are values that are different from most of the data points in the dataset. Point anomalies can cause biases in data models, as these anomalies represent errors in data collection or recording, or they may indicate true rare events within the dataset.

(2) Contextual anomalies: These have relatively larger or smaller values within a context or subset, but not globally. Contextual anomalies are useful for understanding how data behaves under different conditions, but they can also lead to the omission of important local information in global analysis.

(3) Collective anomalies: These are composed of multiple related data points that, as a whole, are anomalous with respect to the entire dataset. Collective anomalies represent new patterns or trends in the data, or they may indicate some kind of systemic problem.

2.2 Challenges Faced in Anomaly Detection

Time series data is characterized by its continuity and complexity. In the vast amount of data, accurately and efficiently detecting anomalies is crucial for system safety. Currently, time series modeling methods based on deep learning have shown good results in real-time status detection, fault detection, and have effectively improved operational efficiency. Against this backdrop, a large number of deep learning research results have emerged in the field of time series anomaly detection, while also facing many challenges:

(1) The recall rate of anomaly detection is low. Due to the rarity and diversity of anomalous events, it is difficult to identify all anomalies. Even many normal instances are misclassified as anomalies, while true anomalies are overlooked. Existing methods have a high false positive rate, and reducing this rate and improving the recall rate is an important challenge.

(2) Insufficient high-dimensional data processing capability. In low-dimensional space, anomalous features are more apparent, but in high-dimensional space, they become hidden and less obvious. This increases the difficulty of anomaly detection.

(3) Susceptibility to noise. Many weakly supervised or semi-supervised anomaly detection methods assume that the labels in the training data are correct, but such data may contain noise samples that are incorrectly marked as another category, thereby affecting the detection results.

(4) Limitations in model training time and computational resources. Anomaly detection methods based on deep learning typically require longer training times and a large amount of computational resources, which can be a limiting factor in situations with limited resources.

3. Classification of Anomaly Detection Methods

The anomaly detection method can be divided into three categories, supervised, unsupervised and semi-supervised according to whether the data set has labels:

(1) Supervised anomaly detection means that all the data in the training set have labels, but it is very difficult to obtain the accurate labeled training data set, which usually requires manual annotation, so the application scope of supervised anomaly detection is small;

(2) Unsupervised anomaly detection means that the data in the training set are unmarked, and unsupervised anomaly detection is widespread;

Semi-supervised abnormality detection means that some samples in the training set have labels, and the use of labeled data can improve the detection performance of the model. Therefore, the semi-supervised anomaly detection is also widely used.

Anomaly detection can be roughly divided into three categories according to the way of processing data: statistical based methods, machine learning based methods and deep learning based methods.

3.1 Statistics-based Methods

The advantages of statistics-based anomaly detection methods are low complexity and fast calculation speed, which are suitable for scenarios without historical data. This method ignores the temporal indexing in the time series and takes the points in the time series as statistical sample points. Assuming that the data follows some statistical model, the data points with a small probability of occurrence are anomalies. The statistics-based abnormality detection algorithm [5] is an early and mature technology, and n-Sigma and Boxplot are the most common statistical methods. The n-Sigma assumes that the data follow a normal distribution, and the data whose distance from the mean exceeds n times the standard deviation are labeled as abnormal. Classical parametric statistical models such as mobile autoregressive models also perform anomaly detection by assuming that the basic distribution of normal data fits the preset distribution. The Gaussian mixture model estimates the probability density of normal class data, and introduces a confidence measure to estimate the reliability of normal data. When the confidence takes a large value, the algorithm updates the parameter once. Non-parametric statistical models are based on kernel functions, and they learn the underlying distribution of normal behavior directly from a given data, but are difficult for the processing of high-dimensional data. Statistics-based methods are highly hypothetical, and the efficiency of abnormality detection will decrease when the amount of data and dimensionality increase.

3.2 Machine Learning-based Methods

Machine learning-based anomaly detection algorithms can be classified into clustering-based, classification-based, and density-based anomaly detection algorithms. Clustering-based anomaly detection combines clustering with anomaly detection algorithms, using existing clustering algorithms directly or indirectly to alleviate the problem of low detection efficiency in high-dimensional sparse data. This algorithm considers data that is isolated or far from the cluster center as anomalies [6-8]. For example, the DBSCAN algorithm clusters data in wireless sensor networks and considers low-density areas as anomalies. The K-means algorithm clusters traffic data [9], and makes judgments on normal and abnormal results based on the assumption principle. Anomaly detection based on Bayesian networks [10] estimates the posterior probability distribution of normal and abnormal data by constructing a naive Bayesian network, aggregates the posterior probability distribution of data attributes, and extends the univariate classification algorithm to multivariate anomaly detection tasks. It has already been applied in areas such as network intrusion and image anomaly detection, but due to its conditional independence assumption, it often performs poorly in practical applications. Clustering-based anomaly detection algorithms are an unsupervised detection method, with the greatest advantage being the lack of labeled data and ease of understanding, but the detection effect and computational complexity decrease with increasing dimensions.

Classification-based anomaly detection algorithms assume that normal data can be smoothly mapped to a feature subspace, and data points that cannot be mapped to the subspace are considered as anomalies. The Isolation Forest anomaly detection algorithm uses a binary tree data structure to classify anomalies at the root of the tree and normal data into deeper nodes. The Support Vector Machine (SVM) anomaly detection algorithm [11-12] maps normal and anomalous data to a low-dimensional space through SVM, and finds a decision boundary in the feature space to separate normal and anomalous data. Due to the difficulty in obtaining negative class samples in anomaly detection, a variant of this algorithm, One-Class SVM, has been developed that can make use of normal data for anomaly detection. However, this algorithm also has some drawbacks, such as the computation of its kernel function requiring a significant amount of computational resources.

Density-based anomaly detection considers high-density areas as normal and low-density areas as abnormal. The fundamental idea is to detect anomalies by comparing the density of data with its neighboring points. In this category of algorithms, the Local Outlier Factor (LOF) algorithm views anomalies not as a binary attribute but as a measure, with a higher LOF value indicating a greater likelihood of the data being anomalous. Its variant, the Connectivity-based Outlier Factor (COF), calculates the density of different data regions within the dataset and considers regions with lower density as outliers. Normal data points have strong connectivity with other data points within their neighborhood, whereas anomalous points have weaker connectivity. The classic K-nearest neighbor (KNN) anomaly detection [13] employs pruning techniques to improve running speed when dealing with high-dimensional data. Density-based bias sampling algorithms combine probabilistic analysis with density calculations to reduce the time and space complexity of the algorithm, enhancing the efficiency of anomaly detection. Although density-based anomaly detection algorithms do not require labels, they have high complexity and are sensitive to the choice of parameters such as the anomaly factor threshold.

In addition to the more common machine learning-based anomaly detection methods, there are also anomaly detection methods based on information theory. Assuming that anomalies have a greater impact on content information compared to normal points, algorithms determine whether the removed data is anomalous based on the extent of change in content information after data removal. Information entropy

anomaly detection uses information entropy to identify anomalies. If the entropy of the dataset decreases after removing a data point, then the removed point is considered an anomaly. The information bottleneck method defines anomalous data clusters using the information bottleneck and detects anomalies based on the difference between the distribution of anomalous data clusters and normal data clusters. Anomaly detection algorithms based on principal component analysis use orthogonal transformations to map data into a low-dimensional subspace, where the features in the low-dimensional space retain the key components of the original data to the greatest extent. Vector-based anomaly detection uses the variance of vector angles as a criterion for identifying anomalies, which to some extent alleviates the difficulties of dealing with high-dimensional data.

3.3 Deep Learning-based Methods

Deep learning-based time series anomaly detection has become a popular research direction in recent years, using various neural networks to extract data features and identify anomalies that do not conform to patterns from these features. It can be divided into three categories: prediction-based time series anomaly detection algorithms, reconstruction-based anomaly detection algorithms, and generation-based anomaly detection algorithms.

Prediction-based time series anomaly detection uses neural networks to predict time series and determines anomalies by comparing the distance between predicted values and original data. Convolutional anomaly detection [15] learns the geometric shape of data features by using the Jacobian matrix output from a regularized encoder, achieving a smoother data representation space, and the algorithm can more effectively detect anomalies from normal data. LSTM anomaly detection [16-17] predicts time series, taking adjacent data segments as true values, and identifies anomalies by comparing predicted values with true values. Multilayer convolutional neural networks [18] reduce the dimensionality of the original data with multiple layers of CNNs and then increase it again, training the network so that the result after dimensionality increase approximates the original data segments, and anomaly detection is performed by comparing the differences before and after training.

Reconstruction-based time series anomaly detection is achieved by minimizing reconstruction errors, with the most classic reconstruction algorithm being the autoencoder and its variants. Spectral encoding anomaly detection [19] constructs a denoising autoencoder based on bidirectional LSTM, mapping data to spectral features, with the decoder reconstructing the original data and determining whether the input data is anomalous based on the difference between the reconstructed data and the original data. Group-robust deep autoencoders decompose data into reconstructed data and anomalous noise data, using sparse regularization terms to constrain the anomalous noise data, thereby achieving robust anomaly detection results. Bayesian convolutional autoencoders [20] show a significant difference between the reconstructed data and the original data when anomalous data is input into the autoencoder. Ensemble methods integrate different types of autoencoders to detect anomalous data, replacing fully connected autoencoders with autoencoders of different structures and connection methods, reducing the computational complexity of the algorithm. Deep anomaly detection based on variational autoencoders [21] uses variational inference mechanisms to learn the generative distribution information of normal data. Improved variational lower bound anomaly detection trains the model with an improved version of the variational lower bound and uses Markov chain Monte Carlo strategies to detect anomalies in the data. The variational deviation network model uses variational autoencoders to generate reference scores, offering better scalability and stronger interpretability for anomalous data. However, this model also has certain limitations, as it uses a normal distribution to fit the probability distribution of normal samples. When training data and test data come from different data distributions, variational autoencoders cannot be used for anomaly detection on the data.

Generation-based time series anomaly detection [22] adopts a game-theoretic approach to learn the marginal distribution information of normal data. Detection generative adversarial networks [23] use convolutional generative adversarial networks to learn the population distribution of normal data, and when the model is input with anomalous data, the generated anomalous data differs significantly from the original data, with the difference serving as an anomaly score. Conditional generative anomaly detection [24] uses conditional generative adversarial networks to learn the distribution of the data generation space and the data inference space, constructing data encoders and data decoders to extract features and reconstruct data, respectively, while other encoders learn the latent space representation of the reconstructed data. Active learning anomaly detection constructs multiple generators to generate different potential outlier data to avoid the problem of mode collapse. Adversarial inference anomaly detection [25] is an inference-based anomaly detection algorithm that uses bidirectional generative adversarial networks to infer generated data and learn the joint distribution of normal and anomalous data. Anomalies are determined by calculating the reconstruction error of data features.

4. Deep Anomaly Detection Model

4.1 Deep Anomaly Detection Model Based on Transformer

The Transformer model is a deep learning model architecture for sequence-to-sequence tasks in natural language processing, introducing self-attention mechanism and multi-head attention mechanism to escape the sequence dependence problem inherent in traditional recurrent neural networks. It is mainly composed of four parts: input part, multi-layer encoder, multi-layer decoder and output part.

(1) Input part: generate a position vector for each position of the input sequence, so that the model can understand the position information in the sequence.(2) Encoder part: composed of N encoder layers. Each encoder layer consists of two sub-layer connected structures: multi-head self-attention sublayer and feedforward fully connected sublayer.(3) Decoder part: stacked of N decoder layers. Each decoder layer consists of three sub-layer connected structures: a masked multi-head self-attention sub-layer, a multi-head attention sub-layer, and a feedforward fully connected sub-layer. Each sublayer is connected by a normalized layer and a residual connection.(4) Output part: convert the vector of the output of the decoder into the linear layer of the final output dimension and the softmax layer of converting the output of the linear layer into the probability distribution.

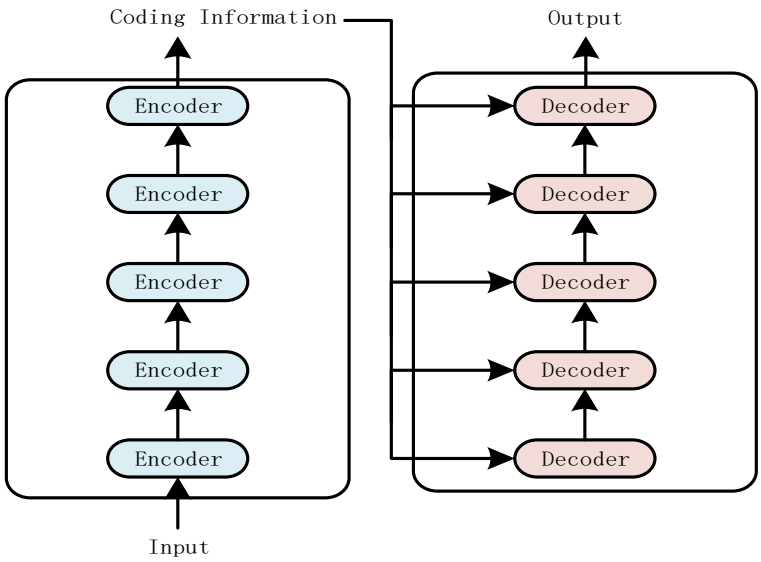

Figure 1. Transformer Brief diagram of the structure

4.2 Deep Anomaly Detection Model Based on Autoencoder

The autoencoder consists of two components, the encoder and the decoder, where the input is compressed into the latent spatial representation through the encoder, and the decoder uses these latent spatial representations as inputs for the reconstruction output. By adding constraints to the autoencoder, the input and output are approximately identical to effectively learn the data features. The autoencoder is a widely used data compression technology that can not only be used in both for dimension reduction and feature learning, but also as a good data noise reduction algorithm. Autoencoder and its variants have been widely used in anomaly detection and data generation, and it is an unsupervised deep learning method.

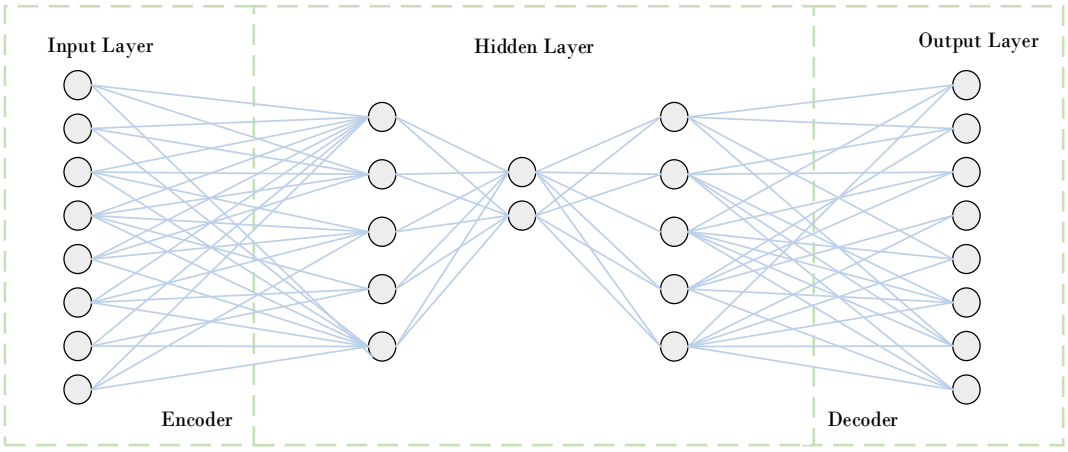

Figure 2. Structure diagram of the autoencoder

In unsupervised training, the loss function helps to correct errors generated by the model, with the goal to make the inputs and outputs of the model as close as possible. As shown in the following formula, the input is sent to the autoencoder and reconstructed, the encoder by the function f and the decoder by the function g:

$$h(x) = f(w_1 x + b_1)$$
$$o(x) = g(w_2 h(x) + b_2)$$

Where x is the input; w_1 and w_2 are the weight matrices of the input-output layers and the hidden layers, respectively; and b_1 and b_2 are the bias corresponding to each stage.

The autoencoder structure is clear and easy to understand, suitable for different types of data, and many powerful variants are derived. However, abnormalities in the training data may bias the learned feature representation. Moreover, the objective function of data reconstruction is mainly aimed at dimension reduction and data compression, not specifically for anomaly detection. This causes the learned feature representation tends to generalize latent patterns and is not optimized for anomaly detection.

4.3 Deep Anomaly Detection Model Based on LSTM

As a variant of recurrent neural network (RNN), LSTM inherits the ability of RNN to process time series data, which can make use of historical data for predict, while overcoming the problems of short-term memory and gradient disappearance of RNN. Through its unique memory module, LSTM is able to learn and retain long-term dependent

information, thus enabling the model to perform well when processing time-series data. Recently, models based on LSTM models and their variants have been widely used in various research fields, solving many problems that are difficult to overcome by traditional AI algorithms.

For a given time series input x_t, LSTM jointly controls its unit state c_t and output h_t by forgetting the gate, input gate and output gate:

$$f_t = \sigma(W_{xf} \cdot x_t + W_{hf} \cdot h_{t-1} + b_f)$$
$$i_t = \sigma(W_{xi} \cdot x_i + W_{hi} \cdot h_{t-1} + b_i)$$
$$o_t = \sigma(W_{xo} \cdot x_t + W_{ho} \cdot h_{t-1} + b_o)$$
$$c_t = f_t \otimes c_{t-1} + i_t \otimes tanh(W_{xc} \cdot x_t + W_{hc} \cdot h_{t-1} + b_c)$$
$$h_t = o_t \otimes tanh(c_t)$$

Where W and b represent the weight matrix and the bias vector, respectively. The forgetting gate is responsible for discarding the useless information existing in the past cell state of the LSTM, retaining the information added to the current cell state of the LSTM, and determining the information output. Due to the presence of these gates, LSTM can more easily simulate long-term change patterns in time series.

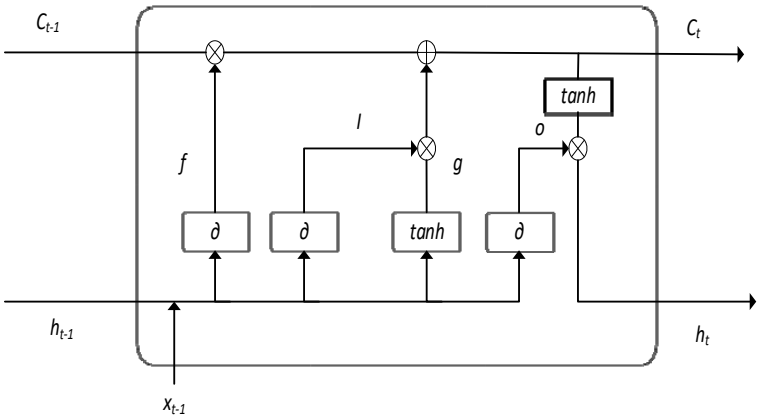

Figure 3. Structural diagram of the LSTM cells

LSTM network model of time series in the study of various time series, which can learn adaptively from a large number of normal samples and realize real-time anomaly detection. There are also some deficiencies, including high computational costs, overfitting risks, complex parameter adjustments, and insufficient sensitivity to some abnormal types, which may affect their performance and efficiency in practical applications.

4.4 Deep Anomaly Detection Model Based on GAN

Generating Generative Adversarial Network (GAN) is mainly composed of the generator and the discriminator, in which the discriminator is responsible for the global judgment, and the generator focuses on the generation of local details. The goal of the generator is to maximize capturing the characteristics of the training sample to generate realistic samples sufficient to fool the discriminator. The purpose of the discriminator is to compare the two, to distinguish the authenticity of the input data as much as possible, and to narrow the deviation between the generated sample and the real sample.

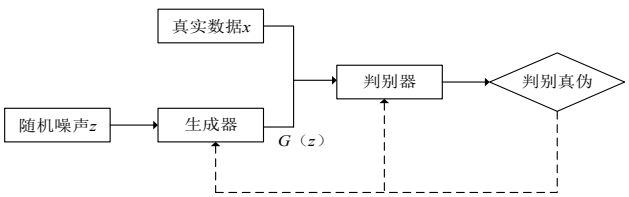

Figure 3. GAN structure diagram

In the training process, while the generator improves the falsification ability, the discriminator also improves the discrimination ability. The confrontation between the generator and the discriminator forms a dynamic game process. Nash equilibrium is reached when the generator can generate data that the discriminant has difficulty distinguishing between true and false. The GAN optimization objective is to maximize discriminant parameters, while minimizing generator parameters. The objective function formula is as follows:

$$Min_G Max_D V(D,G) = E_{x \sim P_{data}(x)}[logD(x)] + E_{x \sim P_z(z)}[log(1 - D(G(z))]$$

Where z represents the random noise of the input, P_z (z) represents the distribution of the generated network, x represents the real data, and P_{data} (x) represents the distribution of the real data. The discriminant will bring $D(G(z))$ to 0, the generator will bring $D(G(z))$ to 1, and when $D(G(z))$ =0.5, the generated sample can be false, theoretically reaching Nash equilibrium.

GAN has been very mature in image data processing, and a large number of existing GAN correlation models and theories provide a theoretical basis for anomaly detection. However, there are still some defects in GAN-based abnormality detection: the GAN training process may face the non-convergence problem, which increases the training difficulty; the generator network may be misled when the true distribution of the data set is complex or the training data contains inaccurate outliers.

4.5 Deep Anomaly Detection Model Based on CNN

When processing images, CNN treats the images as two-dimensional matrices, and gradually extracts and combines features through multi-layer convolution operation and pooling operation to realize the efficient processing of images. In the time series analysis, the time series can be regarded as a one-dimensional vector of 1*n. The advantage of applying convolutional neural networks in time series prediction is that their multi-layer network structure enables massively parallel processing. This structure can build a deep learning network, improving performance while also saving time.

5. Experiment

5.1 Dataset

To compare the effects of different models in anomaly detection applications, this paper selected two public datasets to conduct experiments on various models.

One of the datasets is the KPI dataset released by the AIOPS data competition [26], which consists of multiple KPI curves, and the anomaly labels come from several internet companies including Sogou, Tencent, and eBay. Most of the KPI curves have data points with a 1-minute interval, while some have a 5-minute interval. These datasets cover time series with different time intervals and a wide range of patterns, and are commonly used to evaluate the performance of time series anomaly detection. The other dataset comes from Huawei's NAIE platform, which is a KPI anomaly detection dataset based on Huawei's real business. It includes a labeled training set and an unlabeled testing set, and its data format is basically consistent with the AIOPS dataset.

5.2 Evaluation Metrics

In evaluating model performance, Precision, Recall, F1-score, and accuracy (Acc) are commonly used metrics, which are defined as follows:

$$Acc = \frac{T_p + T_n}{T_p + F_p + T_n + F_n}$$

$$P = \frac{T_p}{T_p + F_p}$$

$$R = \frac{T_p}{T_p + F_n}$$

$$F_1 = 2 \times \frac{PR}{P + R}$$

Among them, T_p represents the number of instances correctly detected as normal and are actually normal, F_p represents the number of instances detected as normal but are actually abnormal, T_n represents the number of instances correctly detected as abnormal and are actually abnormal, and F_n represents the number of instances detected as abnormal but are actually normal.

5.3 Experimental Analysis

The experiment compares the effectiveness of five representative deep learning anomaly detection algorithms, which are introduced in detail in the text, on time series data. To ensure the consistency and comparability of the experiment, the evaluation was conducted on two public KPI anomaly detection datasets. This ensures the applicability and representativeness of the research results, providing a reliable basis for technology selection in different application scenarios.

Experiments conducted on the KPI dataset released by the AIOPS data competition yielded a comparison of performance metrics including precision, recall, F1 score, and accuracy, as shown in the following table:

Table 1. Results of Models' Anomaly Detection Metrics

Detection Methods	Precision	Recall	F1-score	Accuracy
AE	0.80	0.78	0.85	0.77
Transformer	0.73	0.80	0.83	0.75
LSTM	0.87	0.90	0.87	0.86
GAN	0.85	0.88	0.86	0.83
CNN	0.82	0.87	0.86	0.85

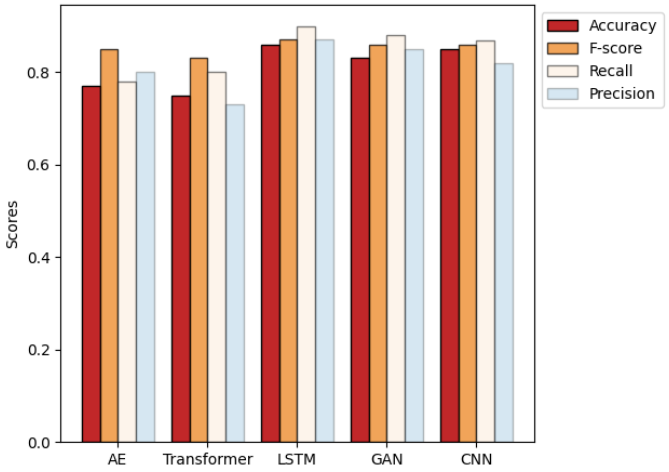

Figure 5. Detection Results of Models on KPI Data

On the KPI anomaly detection dataset released by Huawei's NAIE platform, only the F1 score performance metric was evaluated. It can be observed that different models perform relatively well on the AIOPS dataset.

Table 2. The F1-Score of Models on the Huawei Dataset

Detection Methods	HUAWEI
AE	0.73
Transformer	0.72
LSTM	0.74
GAN	0.73
CNN	0.69

The experimental results from two datasets indicate that LSTM, GAN, and CNN perform well in anomaly detection tasks, achieving high scores in evaluation metrics such as precision, recall, F1 score, and accuracy. Different types of deep learning models have their own focuses in anomaly detection tasks. LSTM is effective in capturing complex features and long-term dependencies in the data; GAN has an advantage in generating deceptive anomaly samples; while LSTM and CNN excel in handling the sequential and local features of time series data. Additionally, although the performance of the five algorithms on Huawei's KPI dataset is slightly inferior, the average F1 score exceeds 0.7. This suggests that deep learning-based anomaly detection algorithms have good generalization capabilities and can be somewhat applied to real-world production environments.

6. Summary

Time series anomaly detection has important practical value and is a subject that has been studied extensively. In actual work, it is often difficult to obtain high-quality labels for supervised learning anomaly detection algorithms, so unsupervised or semi-supervised deep learning anomaly detection algorithms have higher universality and a wider range of application scenarios.

This paper provides a classification overview of existing anomaly detection methods and focuses on introducing five classic deep learning models. The performance of different deep learning time series anomaly detection algorithms was evaluated on two public KPI anomaly detection datasets. The experimental results show that LSTM, GAN, and CNN perform well in the anomaly detection task. Meanwhile, the average F1 score of the five deep learning algorithms exceeded 0.7, proving that the deep learning-based anomaly detection model has good generalization ability. Deep learning technology has certain application value in time series anomaly detection, but there are also certain limitations, such as high training cost, complex model, and insufficient sensitivity to certain anomaly types.

References

[1] Imani M. Collaborative representation based unsupervised CNN for hyperspectral anomaly detection[J]. Infrared Physics & Technology, 2024, 141: 105498.

[2] Cai Y, Tu Y X, Teng Y T, et al. Anomaly detection of earthquake precursor data using long short-term memory networks [J]. Applied Geophysics, 2019, 16(03): 257-266.

[3] Xue J, Yan S, Qu JH, et al. Deep Membrane Systems for Multitask Segmentation in Diabetic Retinopathy[J]. Knowledge-Based System, 2019, 183(1): 1-10.

[4] Wang Q, Qi F, Sun M, et al. Identification of Tomato Disease Types and Detection of Infected Areas Based on Deep Convolutional Neural Networks and Object Detection Techniques[J]. Computational Intelligence and Neuroscience, 2019, 2019(2): 1-15.

[5] Pittino F, Puggl M, Moldaschl T, et al. Automatic anomaly detection on in-production manufacturing machines using statistical learning methods[J]. Sensors, 2020, 20(8): 2344.

[6] Wu S, Fang L, Zhang J, et al. Unsupervised Anomaly Detection and Diagnosis in Power Electronic Networks: Informative Leverage and Multivariate Functional Clustering Approaches[J]. IEEE Transactions on Smart Grid, 2023.

[7] Wang C, Zhou H, Hao Z, et al. Network traffic analysis over clustering-based collective anomaly detection[J]. Computer Networks, 2022, 205: 108760.

[8] Enayati E, Mortazavi R, Basiri A, et al. Time series anomaly detection via clustering-based representation[J]. Evolving Systems, 2024, 15(4): 1115-1136.

[9] Li, M., Xu, D., Zhang, D. et al. The seeding algorithms for spherical k-means clustering[J]. Glob Optim,2020: 695–708.

[10] Chen J, Pi D, Wu Z, et al. Imbalanced satellite telemetry data anomaly detection model based on Bayesian LSTM[J]. Acta Astronautica, 2021, 180: 232-242.

[11] Hosseinzadeh M, Rahmani A M, Vo B, et al. Improving security using SVM-based anomaly detection: issues and challenges[J]. Soft Computing, 2021, 25(4): 3195-3223.

[12] Li Y, Lei M, Liu P, et al. A novel framework for anomaly detection for satellite momentum wheel based on optimized SVM and Huffman-Multi-Scale entropy[J]. Entropy, 2021, 23(8): 1062.

[13] Djenouri Y, Belhadi A, Lin J C W, et al. Adapted k-nearest neighbors for detecting anomalies on spatio–temporal traffic flow[J]. Ieee Access, 2019, 7: 10015-10027.

[14] Ruff L, Kauffmann J R, Vandermeulen R A, et al. A unifying review of deep and shallow anomaly detection[J]. the IEEE, 2021, 109(5): 756-795.

[15] Zhao H, Liu M, Qiu S, et al. Satellite unsupervised anomaly detection based on deconvolution-reconstructed temporal convolutional autoencoder[J]. IEEE Transactions on Consumer Electronics, 2023.

[16] Kong F, Li J, Jiang B, et al. Integrated generative model for industrial anomaly detection via bidirectional LSTM and attention mechanism[J]. IEEE Transactions on Industrial Informatics, 2021, 19(1): 541-550.

[17] Su Y, Zhao Y, Niu C,et al. Robust anomaly detection for multivariatetime series through stochastic recurrent neural network[C]//Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery &data mining.2019:2828-2837.

[18] Zhao P, Chang X, Wang M. A novel multivariate time-series anomaly detection approach using an unsupervised deep neural network[J]. IEEE Access, 2021, 9: 109025-109041.

[19] Yao X, Zhu J, Jiang Q, et al. RUL prediction method for rolling bearing using convolutional denoising autoencoder and bidirectional LSTM[J]. Measurement Science and Technology, 2023, 35(3): 035111.

[20] Zou L, Zhuang K J, Zhou A, et al. Bayesian optimization and channel-fusion-based convolutional autoencoder network for

fault diagnosis of rotating machinery[J]. Engineering Structures, 2023, 280: 115708.

[21] Moon J, Noh Y, Jung S, et al. Anomaly detection using a model-agnostic meta-learning-based variational auto-encoder for facility management[J]. Journal of Building Engineering, 2023, 68: 106099.

[22] de Albuquerque Filho J E, Brandão L C P, Fernandes B J T, et al. A review of neural networks for anomaly detection[J]. IEEE Access, 2022, 10: 112342-112367.

[23] Zhang X, Mu J, Zhang X, et al. Deep anomaly detection with self-supervised learning and adversarial training[J]. Pattern Recognition, 2022, 121: 108234.

[24] Chen Z, Duan J, Kang L, et al. Supervised anomaly detection via conditional generative adversarial network and ensemble active learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 45(6): 7781-7798.

[25] Schlegl T, Seeböck P, Waldstein S M, et al. f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks[J]. Medical image analysis, 2019, 54: 30-44.

[26] Li Z, Zhao N, Zhang S, et al. Constructing large-scale real-world benchmark datasets for Aiops[J]. arXiv preprint arXiv:2208.03938, 2022.