ANOMALYBERT: SELF-SUPERVISED TRANSFORMER FOR TIME SERIES ANOMALY DETECTION USING DATA DEGRADATION SCHEME

Yungi Jeong, Eunseok Yang, Jung Hyun Ryu, Imseong Park, Myungjoo Kang* Numerical Computing & Image Analysis Lab Seoul National University, Seoul, Republic of Korea {jyg9628, mayth24, jhryu30, parkis, mkang}@snu.ac.kr

ABSTRACT

Mechanical defects in real situations affect observation values and cause abnormalities in multivariate time series, such as sensor values or network data. To perceive abnormalities in such data, it is crucial to understand the temporal context and interrelation between variables simultaneously. The anomaly detection task for time series, especially for unlabeled data, has been a challenging problem, and we address it by applying a suitable data degradation scheme to selfsupervised model training. We define four types of synthetic outliers and propose the degradation scheme in which a portion of input data is replaced with one of the synthetic outliers. Inspired by the self-attention mechanism, we design a Transformer-based architecture to recognize the temporal context and detect unnatural sequences with high efficiency. Our model converts multivariate data points into temporal representations with relative position bias and yields anomaly scores from these representations. Our method, AnomalyBERT, shows a great capability of detecting anomalies contained in complex time series and surpasses previous state-of-the-art methods on five real-world benchmarks. Our code is available at https://github.com/Jhryu30/AnomalyBERT.

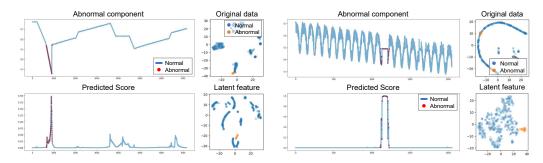


Figure 1: Examples of anomaly detection for abnormal time series in SWaT (*left*) and SMAP (*right*) datasets. Our method, AnomalyBERT, predicts anomaly scores that quantify abnormalities of data points in time series. We also visualize the original data and the final latent features in our model using t-SNE and show that our method separates abnormal points effectively.

1 Introduction

In many industrial environments, time series data is mainly dealt with to monitor machines, IT devices, spacecrafts, or engines. Anomaly detection is one of the essential tasks for time series analysis, which can find defects in machines and prevent potential harm. Recently, many deep learning-based approaches have been applied to this work. Several studies design recurrent neural

^{*}Corresponding author.

network (RNN) models (Hundman et al., 2018; Park et al., 2018; Su et al., 2019) to treat multivariate data in the temporal order. Some researchers try to adopt a graph or Transformer architecture (Vaswani et al., 2017) to focus on relationships between variables and data points (Deng & Hooi, 2021; Xu et al., 2021). These approaches take account of the temporal characteristics of data and successfully adapt deep neural networks to the field of time series analysis.

Most datasets do not provide ground truth labels for the training set in this area. In other words, it is unknown whether a point is anomalous or not in the training set. Therefore, previous studies have developed unsupervised learning methods for anomaly detection. Some of them insert an autoencoder or an adversarial network (Goodfellow et al., 2014) into their model to understand data distribution efficiently. However, it is still a hard problem to detect anomalies from the temporal context without supervision. Moreover, abnormalities may display unexpected behavior and be related to multiple variables, which makes the detection task more difficult in real situations.

In this paper, we design a Transformer-based architecture and propose *AnomalyBERT*, a self-supervised method for time series anomaly detection. Inspired by BERT (Devlin et al., 2018) in the natural language processing (NLP) field, we modify the masked language modeling (MLM) by replacing a random portion of input data and training a model to find the degraded part. This *data degradation scheme* helps detect varied unnatural sequences in real-world time series, as shown in Figure 1. Furthermore, we apply 1D relative position bias (Raffel et al., 2019) to self-attention modules to insert appropriate temporal information into data. AnomalyBERT outperforms previous detection methods by achieving the highest F1-scores on five real-world benchmark datasets. We demonstrate that our data degradation scheme enables the Transformer-based model to understand the temporal context, and our method has strong capability in detecting real-world anomalies.

2 RELATED WORKS

Time Series Anomaly Detection. Anomaly detection problems have been handled using various statistical and machine learning-based (ML-based) methods. A classical method, Local Outlier Factor (LOF) (Breunig et al., 2000), is proposed for density-based outlier detection. Following the LOF, several statistical methods (Tang et al., 2002; Kriegel et al., 2009) and ML-based methods (Tax & Duin, 2004; Ruff et al., 2018; Liu et al., 2008) have been proposed. For example, DAGMM (Zong et al., 2018) combines Gaussian Mixture Model (GMM) with a deep neural network. Meanwhile, deep learning-based approaches have appeared in this area with the advances in neural networks. They commonly adopt RNNs to deal with complex time series, such as multiple sensor values from IoT data. Park et al. (2018) propose LSTM-VAE, a variational autoencoder (VAE) model whose feed-forward networks are replaced with Long Short-Term Memory (LSTM). Su et al. (2019) also propose a VAE model, but they use a gated recurrent unit (GRU) to extract latent features. Recently, there have been attempts to employ advanced models. Deng & Hooi (2021) present a graph neural network (GNN) that detects deviations from the relationships between variables, and Xu et al. (2021) propose a Transformer-based model and define the association discrepancy for detection criterion.

Transformer and Its Variants. Transformer (Vaswani et al., 2017) is first introduced in the field of NLP and has achieved big success. It employs an attention mechanism in the multi-head structures and builds a pair of an encoder and a decoder with the attention layers. The concepts of Transformer have been applied to various NLP tasks. For example, BERT (Devlin et al., 2018) uses a Transformer encoder that is pre-trained on large-scale unlabeled datasets. BERT implements MLM in the pre-training phase to understand the context of the sentences. There also have been successful methods for effective generalization. T5 (Raffel et al., 2019) is a text-to-text method using both an encoder and a decoder. SpanBERT (Joshi et al., 2020) introduces a span masking instead of the MLM in BERT, and XLNet (Yang et al., 2019) combines autoencoding and autoregressive modeling. BART (Lewis et al., 2019) uses several noising schemes including the MLM in BERT. The Transformer architecture also has been applied to computer vision tasks recently. Vision Transformer (ViT) (Dosovitskiy et al., 2020) employs a Transformer encoder without CNN architecture and achieves outstanding results in classification tasks.

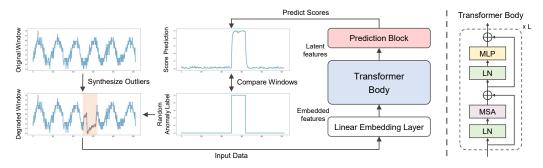


Figure 2: An overview of our framework. We design a Transformer-based model and a self-supervised training strategy. In the training stage, a portion of an input window is randomly replaced and the model is directed to classify the degraded part. The main Transformer body is composed of Transformer layers with 1D relative position bias.

3 METHOD

3.1 Overall Architecture

An overview of our framework is illustrated in Figure 2. Our model is composed of three parts; a linear embedding layer, a Transformer body, and a prediction block. A window of multivariate time series $X = x_{t_0:t_1} \in \mathbb{R}^{N \times D}$ is fed into the model as an input. The linear embedding layer first projects each data patch $x_{t:t+p}$ (a patch consists of several neighboring points) in a window X to an embedded feature f_i . Then the Transformer body takes all embedded features $\{f_i\}_{1 \leq i \leq M}$ from X and yields latent features $\{h_i\}_{1 \leq i \leq M}$. These latent features share information among themselves and reflect the temporal context in the window. The prediction block finally outputs anomaly scores of data points $a_{t_0:t_1} \in [0,1]^N$ of the window. A data point x_t is regarded as more anomalous as the score a_t is closer to 1.

We adopt the Transformer encoder, in which each layer contains a multi-head self-attention (MSA) module and an MLP block, as the main body. A LayerNorm (LN) layer is placed before each module, and GELU activation is used for activation layers. Unlike the original Transformer or ViT, we do not use sinusoidal positional encodings (Vaswani et al., 2017) or absolute position embeddings (Dosovitskiy et al., 2020) to inject positional information. We instead add 1D relative position bias (Raffel et al., 2019; Liu et al., 2021) to each attention matrix to consider the relative positions between features in a window. A self-attention in each head with the relative bias is computed as:

$$\operatorname{Attention}(Q,K,V) = \operatorname{SoftMax}\left(\frac{QK^T}{\sqrt{d}} + B\right)V, \tag{1}$$

where Q, K, and V are query, key, and value of input features, respectively, and d is the dimension of features in an attention head. $B = [b_{i,j}] \in \mathbb{R}^{M \times M}$ is a relative position bias and an element $b_{i,j} = \hat{b}_{j-i}$ is brought from a learnable bias table $\hat{B} = \{\hat{b}_n\}_{-M+1 \le n \le M-1}$. We apply a different position bias to each MSA module as in Liu et al. (2021).

3.2 SYNTHETIC OUTLIERS AND DATA DEGRADATION

As we use unlabeled training data, we create degraded inputs by replacing a portion of a window with an outlier in the training phase (Figure 3). Similar to the span masking in SpanBERT (Joshi et al., 2020), we randomly select an interval $[t_0', t_1'] \subset [t_0, t_1]$ in a window $X = x_{t_0:t_1}$. The selected sequence $X' = x_{t_0':t_1'}$ is replaced with one of the synthetic outliers below.

- A weighted sequence with the outside of the window (Soft replacement).
- A constant sequence (Uniform replacement).
- A lengthened or shortened sequence (Length adjustment).
- A single peak value (Peak noise).

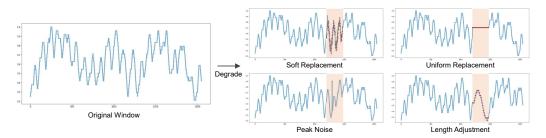


Figure 3: Types of synthetic outliers and degradation examples. We define four types of synthetic outliers, soft replacement, uniform replacement, peak noise, and length adjustment, which are added to input windows for the model training.

The *soft replacement* denotes the replaced sequence fetched from the outside of the window. Technically, it represents the replacement with a weighted sum of the original interval and an external interval. The *uniform replacement* is the replacement with a constant sequence, and the *length adjustment* denotes a lengthened or shortened sequence. Lastly, the *peak noise* is the addition of a single peak value. Unlike the existing method (Lai et al., 2021), our data degradation scheme can be processed without prior knowledge of a given time series.

3.3 Training

We apply the binary cross entropy loss to our objective. For an input window $X = x_{t_0:t_1}$ with a degraded interval $[t'_0, t'_1]$ and predicted anomaly scores $a_{t_0:t_1}$, the objective function is defined as:

$$L = -\frac{1}{N} \sum_{t=t_0}^{t_1} \mathbf{1}_{[\mathbf{t}'_0, \mathbf{t}'_1]}(t) \cdot \log a_t + \left(1 - \mathbf{1}_{[\mathbf{t}'_0, \mathbf{t}'_1]}(t)\right) \cdot \log(1 - a_t), \tag{2}$$

where $N=t_1-t_0+1$ is the window size. The function $\mathbf{1}_{[t_0',t_1']}$ plays a role of artificial labels in this equation. Compared to the MLM in the field of NLP (Devlin et al., 2018; Yang et al., 2019), our model is directed to classify the entire data points in a window into normal/abnormal points at once. At every training step, a synthetic outlier of random type, length, and values is added to an original window under the data degradation scheme. A mini-batch of degraded windows is fed into the model, and the model is trained to detect the degraded parts. The implementation details of the training procedure are described in Appendix A.2.

4 EXPERIMENTS

4.1 DATASETS

We create a simple sine wave dataset that consists of a normal sequence and five abnormal sequences categorized by Lai et al. (2021), and conduct a preliminary experiment on this dataset. We then produce the experimental results on five widely-used benchmark datasets, SWaT, WADI, SMAP, MSL, and SMD (Goh et al., 2017; Ahmed et al., 2017; Hundman et al., 2018; Su et al., 2019). These datasets are collected from multiple sensors in server machines, spacecrafts, or water treatment/distribution systems. Each dataset consists of an unlabeled training set and a labeled test set. The information of datasets is summarized in Table 4 and described in Appendix A.1 in detail.

4.2 Score Prediction and Evaluation Metrics

In the evaluation stage, the trained model takes windows in the test set and predicts anomaly scores of the data points. We use the sliding window strategy (Shen et al., 2020) and average the scores on the overlapped intervals. After the score prediction, we categorize data points whose anomaly scores exceed a threshold as anomalies. We mainly use F1-score (F1) over the ground-truth labels and anomaly predictions to evaluate the performance. We count the number of true positives (TP), false positives (FP), and false negatives (FN), and compute F1 as 2TP / (2TP + FP + FN). In practice,

Table 1: Examination of relationships between the typical types of outliers in Lai et al. (2021) and our proposed synthetic outliers. \bigcirc represents the covering of typical outlier types. It is considered to be covered if both F1 and AUROC score are more than 0.9.

| Typical |] | Point | Pattern | | | |
|---------------------|--------|------------|----------|----------|-------|--|
| Proposed | Global | Contextual | Shapelet | Seasonal | Trend | |
| Soft replacement | 0 | 0 | 0 | 0 | 0 | |
| Uniform replacement | | 0 | 0 | | | |
| Length adjustment | | | | 0 | | |
| Peak noise | 0 | 0 | | | | |

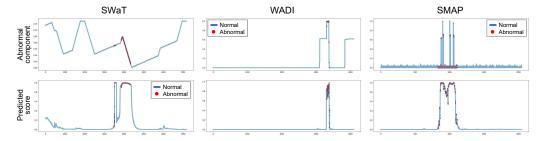


Figure 4: Qualitative results of AnomalyBERT on abnormal sequences from SWaT, WADI, and SMAP datasets. We visualize an abnormal component where the outlier occurs and anomaly scores for each sequence. Without any post-processing, our method finds out diverse types of anomalies.

many approaches process prediction results using the point adjustment (Xu et al., 2018), in which the entire points in an abnormal segment are regarded as anomalous if at least one point is detected as an anomaly. This process, however, has been shown to overestimate the detection performance because it may increase TP but decrease FN dramatically (Kim et al., 2022). Following the protocol in Kim et al. (2022), thresholds that yield the best F1 and F1-score after the point adjustment (F1 $_{\rm PA}$) are obtained, and the best evaluation values are used for comparisons.

4.3 MOTIVATION

Previous work (Lai et al., 2021) categorizes sequential outliers into five specific behavior-driven taxonomy; global, contextual, shapelet, seasonal, and trend outliers. Synthesizing such patterns in time series data is a difficult task because it strictly requires knowledge of patterns that appear in existing data in advance. For example, it is necessary to select a shape to be added to synthesize a shapelet outlier, and the data trend is required to synthesize a trend outlier. However, our synthesis technique covers these five outliers in an easier way because it does not require analysis of the original data. We now refer to these five outlier types as the *typical outlier types*.

In Table 1, we examine the relationships between the typical outlier types and our synthetic outliers using a simplified version of our model and the sine wave dataset. We measure F1 and area under ROC curve (AUROC) for comparisons of the model performances, and consider that a synthetic outlier *covers* a typical outlier type (o mark) if a model trained with the synthetic one achieves both F1 and AUROC over 0.9 on the sine wave including the corresponding typical outlier. As shown in Table 1, the soft replacement covers all typical types of outliers, and the uniform replacement and peak noise partially cover them. Meanwhile, the length adjustment only covers the seasonal outlier. From the simple experiment, we draw inspiration and get ideas for imitating anomalous behavior using synthetic outliers.

4.4 MAIN RESULTS

We report the results of AnomalyBERT on the five real-world datasets introduced in Section 4.1, and compare them to those of the previous works. The reproduced evaluation scores are brought from Kim et al. (2022). In Table 2, we show that AnomalyBERT outperforms the previous methods on

Table 2: F1-scores for various anomaly detection methods and AnomalyBERT on five benchmark datasets. We report the standard F1 and F1-scores after point adjustment (F1_{PA}) following the protocol in Kim et al. (2022). Our method outperforms all existing methods with F1.

| | SW | VaT | WA | ADI | M | SL | SM | IAP | SN | ΔD |
|--------------------|-------|-----------------------------|-------|-----------------------------|-------|-----------------------------|-------|-----------------------------|-------|-----------------------------|
| | F1 | $\mathrm{F1}_{\mathrm{PA}}$ |
| DAGMM (2018) | 0.550 | 0.853 | 0.121 | 0.209 | 0.199 | 0.701 | 0.333 | 0.712 | 0.238 | 0.723 |
| LSTM-VAE (2018) | 0.775 | 0.805 | 0.227 | 0.380 | 0.212 | 0.678 | 0.235 | 0.756 | 0.435 | 0.808 |
| OmniAnomaly (2019) | 0.782 | 0.866 | 0.223 | 0.417 | 0.207 | 0.899 | 0.227 | 0.805 | 0.474 | 0.944 |
| MSCRED (2019) | 0.662 | 0.868 | 0.087 | 0.346 | 0.199 | 0.775 | 0.232 | 0.945 | 0.097 | 0.389 |
| THOC (2020) | 0.612 | 0.880 | 0.130 | 0.506 | 0.190 | 0.891 | 0.240 | 0.781 | 0.168 | 0.541 |
| USAD (2020) | 0.791 | 0.846 | 0.232 | 0.429 | 0.211 | 0.927 | 0.228 | 0.818 | 0.426 | 0.938 |
| GDN (2021) | 0.808 | 0.935 | 0.569 | 0.855 | 0.217 | 0.903 | 0.252 | 0.708 | 0.529 | 0.716 |
| AnomalyBERT (Ours) | 0.854 | 0.925 | 0.580 | <u>0.798</u> | 0.302 | 0.585 | 0.457 | <u>0.914</u> | 0.535 | 0.830 |

Table 3: Results of ablation studies on combinations of the synthetic outlier types. We show the impacts of synthetic outliers by comparing F1-scores on various experimental conditions.

(a) Experimental results on the soft replacement, uniform replace- (b) Experimental results on the length adment, and peak noise on WADI dataset.

| Soft replacement | Uniform replacement | Peak noise | F1 | $\mathrm{F1}_{\mathrm{PA}}$ |
|------------------|---------------------|------------|--------------|-----------------------------|
| 0 | 0 | 0 | 0.580 | 0.798 |
| 0 | 0 | × | 0.504 | 0.756 |
| 0 | × | 0 | <u>0.556</u> | 0.770 |
| × | 0 | 0 | 0.402 | 0.757 |
| 0 | × | × | 0.478 | 0.743 |
| × | 0 | × | 0.403 | 0.706 |
| × | × | 0 | 0.330 | 0.888 |

justment on SWaT dataset.

| Length adjustment | F1 | $F1_{PA}$ |
|-------------------|-------|-----------|
| 0 | 0.854 | 0.925 |
| × | 0.837 | 0.914 |

(c) Experimental results on the length adjustment on **WADI** dataset.

| Length adjustment | F1 | $\mathrm{F1}_{\mathrm{PA}}$ |
|-------------------|-------|-----------------------------|
| 0 | 0.433 | 0.642 |
| X | 0.580 | 0.798 |

all datasets with F1. Our method particularly surpasses the others on MSL and SMAP, which may contain unlabeled outliers in the training set. Despite the difficulty in training networks, our method detects anomalies well in this kind of data. Anomaly BERT also performs well with $F1_{PA}$, although F1_{PA} tends to distort the model performance. Our qualitative results are visualized in Figure 4.

4.5 IMPACT OF SYNTHETIC OUTLIERS

We conduct ablation studies on various synthetic outliers in the training phase and show their impacts on the model performance. In Table 3a, we first report the model performance affected by the three types of outliers, soft replacement, uniform replacement, and peak noise on WADI dataset. We set a baseline by mixing all three outlier types and set experimental conditions by excluding outliers from the baseline one by one. The sum of all probability of synthesizing outliers is fixed at 80%. As shown in Table 3a, using all three outliers yields the best F1, and using the soft replacement and peak noise yields the next. The absence of soft replacement obviously reduces the capability of the model. Also noteworthy, mixing other outlier types with soft replacement enhances the performance compared to using it only. This indicates that the uniform replacement and peak noise complement the soft replacement, though each of them does not perfectly cover the typical outliers in Table 1.

On the other hand, the length adjustment has different influences depending on the datasets. In Table 3b and 3c, we report the results of ablation studies on the length adjustment on SWaT and WADI datasets. Using the length adjustment in the training stage enhances the model performance on SWaT dataset, but it degrades that on WADI dataset. Because the length adjustment specializes in detecting abnormal frequencies as shown in Table 1, it may confuse the model if data contains various frequencies (SMAP, MSL) or low-frequencies (WADI).

5 Conclusion

This paper presents AnomalyBERT, a novel method for time series anomaly detection that uses a data degradation scheme to train a Transformer-based model in a self-supervised manner. We design an appropriate Transformer architecture with 1D relative position embeddings for temporal data and propose four types of synthetic outliers that can cover all typical types of anomalies. Exploiting the synthetic outliers in the training phase, our proposed model can learn to distinguish anomalous behavior. We finally show that our method outperforms previous works and has a strong capability in detecting real-world anomalies in complex time series. Our data degradation scheme has the potential to improve the model performance by revising degradation algorithms to mimic real-world anomalies naturally or mixing proper types of outliers according to data characteristics. Therefore, future studies could be demonstrated on detailed analysis of outlier synthesis processes.

ACKNOWLEDGMENTS

This work was partly supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [NO.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University)], the NRF grant [2021R1A2C3010887], and the ICT R&D program of MSIT/IITP [1711117093, 2021-0-00077] and MOTIE [P0014715].

REFERENCES

- Chuadhry Mujeeb Ahmed, Venkata Reddy Palleti, and Aditya P Mathur. Wadi: a water distribution testbed for research in the design of secure cyber physical systems. In *Proceedings of the 3rd international workshop on cyber-physical systems for smart water networks*, pp. 25–28, 2017.
- Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A Zuluaga. Usad: Unsupervised anomaly detection on multivariate time series. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3395–3404, 2020.
- Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000.
- Ailin Deng and Bryan Hooi. Graph neural network-based anomaly detection in multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 4027–4035, 2021.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- Jonathan Goh, Sridhar Adepu, Khurum Nazir Junejo, and Aditya Mathur. A dataset to support research in the design of secure water treatment systems. In *International conference on critical information infrastructures security*, pp. 88–99. Springer, 2017.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. Advances in neural information processing systems, 27, 2014.
- Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 387–395, 2018.

- Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020.
- Siwon Kim, Kukjin Choi, Hyun-Soo Choi, Byunghan Lee, and Sungroh Yoon. Towards a rigorous evaluation of time-series anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7194–7201, 2022.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.
- Hans-Peter Kriegel, Peer Kröger, Erich Schubert, and Arthur Zimek. Loop: local outlier probabilities. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 1649–1652, 2009.
- Kwei-Herng Lai, Daochen Zha, Junjie Xu, Yue Zhao, Guanchu Wang, and Xia Hu. Revisiting time series outlier detection: Definitions and benchmarks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pretraining for natural language generation, translation, and comprehension. *arXiv* preprint arXiv:1910.13461, 2019.
- Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In 2008 eighth ieee international conference on data mining, pp. 413–422. IEEE, 2008.
- Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.
- Daehyung Park, Yuuna Hoshi, and Charles C Kemp. A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. *IEEE Robotics and Automation Letters*, 3(3):1544–1551, 2018.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.
- Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pp. 4393–4402. PMLR, 2018.
- Lifeng Shen, Zhuocong Li, and James Kwok. Timeseries anomaly detection using temporal hierarchical one-class network. Advances in Neural Information Processing Systems, 33:13016–13026, 2020.
- Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2828–2837, 2019.
- Jian Tang, Zhixiang Chen, Ada Wai-Chee Fu, and David W Cheung. Enhancing effectiveness of outlier detections for low density patterns. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 535–548. Springer, 2002.
- David MJ Tax and Robert PW Duin. Support vector data description. *Machine learning*, 54(1): 45–66, 2004.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, et al. Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In *Proceedings of the 2018 world wide web conference*, pp. 187–196, 2018.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. *arXiv preprint arXiv:2110.02642*, 2021.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V Chawla. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 1409–1416, 2019.

Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International conference on learning representations*, 2018.

A EXPERIMENTAL DETAILS

A.1 DETAILED INFORMATION OF DATASETS

We create the simple sine wave dataset according to the synthesizing criteria introduced by Lai et al. (2021). Based on a noised sine wave, we split the wave into a long length of a normal sequence for training and five slices of abnormal sequences for testing. Each abnormal slice contains one of the typical types of outliers; global, contextual, shapelet, seasonal, and trend outliers.

Table 4: A summary of five real-world benchmark datasets. * indicates the average length of all sub-datasets in the case of SMD.

| Dataset | Train length | Test length | Anomaly % in test | Dimension |
|-------------|--------------|-------------|-------------------|-----------|
| SWaT (2017) | 495,000 | 449,919 | 12.13% | 51 |
| WADI (2017) | 784,537 | 172,801 | 5.77% | 123 |
| MSL (2018) | 58,317 | 73,729 | 10.53% | 55 |
| SMAP (2018) | 153,183 | 427,617 | 12.79% | 25 |
| SMD (2019) | 25,300* | 25,301* | 4.16% | 38 |

The five real-world benchmark datasets are summarized in Table 4 and described below.

Secure Water Treatment (SWaT) (Goh et al., 2017). SWaT is collected from a water treatment testbed for seven days under normal conditions and four days with physical attacks. The data is composed of 51 sensor values. In practice, we ignore the eleventh column in the entire data because unseen patterns labeled as normal in the testing part arise occasionally out of the range of values in the training part. (Our method detects these kinds of normal patterns as anomalies.)

Water Distribution Testbed (WADI) (Ahmed et al., 2017). WADI is a dataset collected from a water distribution testbed with 123 sensors. It contains a normal sequence of 14 days and an abnormal sequence of two days with attack scenarios. For the same reason as SWaT, we ignore the 102nd column in the entire data.

Mars Science Laboratory (MSL) (Hundman et al., 2018). MSL is telemetry data collected by the NASA spacecraft. It consists of datasets from 55 telemetry channels, and each test set is labeled from Incident Surprise Anomaly reports. Following Su et al. (2019), we concatenate all datasets into a pair of training and testing sets and yield a single result. We also prevent windows containing data from multiple channels for score prediction.

Soil Moisture Active Passive (SMAP) (Hundman et al., 2018). SMAP is also collected by the NASA spacecraft and has a similar characteristic to MSL. All sub-datasets are concatenated and produce a single result as in MSL.

Server Machine Dataset (**SMD**) (Su et al., 2019). SMD is a collection of sub-datasets from 28 different machines provided by a large Internet company. Each sub-dataset is equally divided into two parts, the first half for training and the second half for testing. The training and evaluation procedures are carried out on 28 sub-datasets separately, and the averaged results are used for comparisons.

A.2 TRAINING AND EVALUATION SETTINGS

We use one linear layer as the embedding layer and 2-layer MLPs as the prediction block. The Transformer body has six layers of the embedding dimension of 512 and eight attention heads. The prediction block contains one hidden layer of 2,048 neurons with GELU activation in between. The sequence length of embedded features is 512 but window sizes (and patch sizes) of input data vary with datasets. For the preliminary experiment in Section 4.3, we simplify our model by halving the embedding dimension and the number of Transformer layers and shortening the window size and patch size to 100 and 1, respectively.

In the training stage, we train the model with the mini-batch size of 16 for the maximum training steps of 150K. Input windows are selected randomly from the training set at every step. External sequences for the soft replacement are also selected randomly from the same training set. When degrading a window, an average of 30% of columns are degraded but the remaining columns are left. A synthetic outlier type is selected as the soft replacement in 50% probability, uniform replacement in 15% probability, peak noise in 15% probability, and length adjustment in 10% probability. However, the length adjustment may not be used for several datasets because it may reduce the model capability depending on the datasets. The other settings depending on the datasets are presented in Table 5. We employ the AdamW optimizer (Kingma & Ba, 2014) with a learning rate of 1×10^{-4} , and use a learning rate warmup for 10% of training steps and a cosine learning rate decay. To prevent exploding gradients, gradient clipping is applied at a norm of 1.0. For the score prediction of the test set, we slide input windows with the sliding step of 16.

Table 5: Different training settings for five benchmark datasets.

| Dataset Setting Dataset | SWaT | WADI | MSL | SMAP | SMD |
|--------------------------|-------|-------|-------|-------|-------|
| Patch size | 14 | 8 | 2 | 4 | 4 |
| Window size | 7,168 | 4,096 | 1,024 | 2,048 | 2,048 |
| Max length % of outlier | 20% | 15% | 20% | 15% | 20% |
| Use of length adjustment | 0 | × | × | × | 0 |

A.3 DETAILED RESULTS OF SECTION 4.3

We examine relationships between the typical types of outliers and our proposed synthetic outliers in Section 4.3 through F1 and AUROC. The detailed results of F1 and AUROC are presented in Table 7 and Table 6, respectively.

Table 6: AUROC results of the preliminary experiment in Section 4.3.

| | Typical | Point | | Pattern | | |
|----------------|---------|--------|------------|----------|----------|-------|
| Proposed | | Global | Contextual | Shapelet | Seasonal | Trend |
| Soft replaceme | ent | 1.000 | 1.000 | 1.000 | 0.997 | 1.000 |
| Uniform repla | cement | 0.419 | 1.000 | 1.000 | 0.927 | 0.899 |
| Length adjusts | ment | 0.990 | 0.965 | 0.894 | 0.998 | 0.551 |
| Peak noise | | 1.000 | 1.000 | 0.863 | 0.847 | 0.800 |

Table 7: F1 results of the preliminary experiment in Section 4.3.

| Typical | Point | | Pattern | | |
|---------------------|--------|------------|----------|----------|-------|
| Proposed | Global | Contextual | Shapelet | Seasonal | Trend |
| Soft replacement | 1.000 | 1.000 | 1.000 | 0.952 | 1.000 |
| Uniform replacement | 0.000 | 1.000 | 1.000 | 0.825 | 0.889 |
| Length adjustment | 0.667 | 0.400 | 0.556 | 0.947 | 0.625 |
| Peak noise | 1.000 | 1.000 | 0.588 | 0.824 | 0.824 |

B VISUAL RESULTS

We visualize the prediction results of AnomalyBERT and present the 2D projections of the original data and latent features in the model. Similar to Figure 1, we select three abnormal windows in each of SWaT, WADI, and SMAP test sets and fetch the corresponding anomaly scores after the score prediction process. We then project the original data points and the last latent features from the abnormal windows into 2D planes using t-SNE. As shown in Figure 5, our method distinguishes anomalies successfully and separates abnormal data points from normal points well.

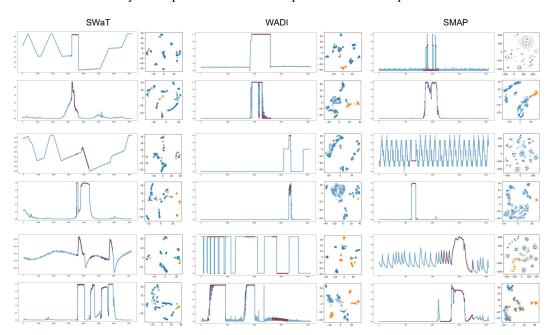


Figure 5: Visualization of anomaly score predictions on abnormal sequences in SWaT, WADI, and SMAP datasets. In each subfigure, graphs of an abnormal component where an outlier occurs (top left) and the corresponding anomaly scores (bottom left) are presented, and the data points (top right) and the last latent features (bottom right) are visualized in 2D planes. Blue lines/dots represent normal values and red/orange dots represent abnormal values.