Expert Systems With Applications 136 (2019) 94–104

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

ALSR: An adaptive label screening and relearning approach for interval-oriented anomaly detection

Jingyu Wang, Yuhan Jing*, Qi Qi, Tongtong Feng, Jianxin Liao*

The State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China

ARTICLE INFO

Article history:
Received 27 January 2019
Revised 28 May 2019
Accepted 14 June 2019
Available online 17 June 2019

Keywords:
Anomaly detection
Multi-type KPIs
Machine learning
Interval-oriented

ABSTRACT

Anomaly detection using KPIs (Key Performance Indicators) is a key part of AIOps (Artificial Intelligence for IT Operations). Recent anomaly detection approaches have adopted Machine Learning to detect anomalies on the perspective of individual time points more than events. These approaches do not effectively utilize the labels of continuous anomaly intervals, nor do they pay attention to the differences among anomaly points. The detection performances are therefore not precise enough to be applied in practice, and the differences in length of anomaly intervals also cause loss of performance. In this paper, we propose an anomaly detection approach named ALSR, which uses a label screening model and a relearning model to analyze and utilize the continuous anomaly intervals of KPIs in finer granularity. The label screening model takes advantage of the continuity of anomaly intervals to remove some unnecessary data from the training set, making it more suitable for interval-oriented anomaly detection. The relearning model based on random forest reclassifies the true/false positive points within domain of detected anomalies, thus effectively reduces the number of false positive points. ALSR uses several features extracted by sliding windows, and the feature set is proved to better describe the characteristics of KPI time series. Finally, we conduct comprehensive experiments on 25 KPIs. The total F-score of ALSR is 0.965, which outperforms state-of-the-art anomaly detection approaches.

1. Introduction

Artificial Intelligence for IT Operations (AIOps) applies artificial intelligence, especially machine learning to the field of operation. It can further handle some problems that cannot be solved by DevOps (Combination of Development and Operation), which uses preset parameters and processes for automated operation. Anomaly detection is a key part of operation, and is also valued in AIOps. KPIs (Key Performance Indicators), such as SRT (search response time) (Chen, Mahajan, Sridharan, & Zhang, 2013) in search engines, page access traffic and number of online users, are generally used for anomaly detection, and they can be obtained from various Web applications. KPIs are important indicators to measure the working status of services. When an anomaly occurs, the corresponding KPI, which is always time series, is likely to show a deviation from normal pattern, and that is the basis for KPI anomaly detection.

Before the appearance and development of AIOps, many algorithms of time series analysis (Ömer Aydin & Kurnaz, 2017; Laptev, Amizadeh, & Flint, 2015; Nguyen & Goulet, 2018) have played a role in anomaly detection of KPIs in time series format. In addition, approaches based on time series prediction (He et al., 2012; Pena, Assis, & Proenca, 2017; Qi, Chu, & He, 2018) have been proposed to improve the performance of anomaly detection. In practice, however, the traditional detectors which need manually and frequently tuning of internal parameters and thresholds (Dao, Liu, Sim, Tull, & Wu, 2018) cannot be well applied to different types of KPIs. They can only be applied to specific type of KPIs after time-consuming manual optimization. Due to such limitations, operators simply choose static thresholds in most cases, resulting in unsatisfactory detection results on various actual KPIs.

With the widely use of machine learning, AIOps based on machine learning has become an effective solution for anomaly detection. Compared with traditional anomaly detection approaches, AIOps generally combines many different detectors, and therefore achieves more flexibility and adaptability. Traditional anomaly detection approaches have been further inherited and synthesized as effective methods for feature extraction in AIOps. Supervised learning (Fontugne, Borgnat, Abry, & Fukuda, 2010; Hoyle et al., 2018; Lavin & Ahmad, 2015; Liu et al., 2015; Shanbhag & Wolf, 2009;

* Corresponding author.
E-mail addresses: wangjingyu@bupt.edu.cn (J. Wang), weatherjyh@163.com (Y. Jing), qiqi8266@bupt.edu.cn (Q. Qi), ftt@bupt.edu (T. Feng), liaojx@bupt.edu.cn (J. Liao).

https://doi.org/10.1016/j.eswa.2019.06.028
0957-4174/

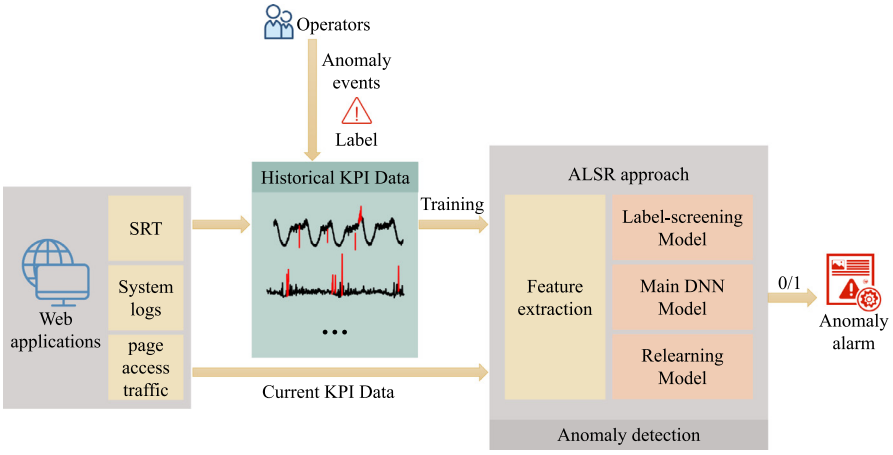

Fig. 1. The use of ALSR in AIOps.

Thi, Cao, & Le-Khac, 2018; Xu et al., 2018) and unsupervised learning (Erfani, Rajasegarar, Karunasekera, & Leckie, 2016; Kim, Yang, Chung, & Cho, 2017; Miao, Liu, Zhao, & Li, 2018) are two main approaches in this field. The philosophy of unsupervised learning is to focus on normal pattern rather than anomaly. Time series of KPIs are composed mainly of normal data, not anomalies, so the model can be effectively trained even without labels. The problem is that the normal patterns of KPIs are difficult to define. Some unsupervised learning models, such as one-class SVM (Erfani et al., 2016; Miao et al., 2018), and generative models (Goodfellow et al., 2014; Kim et al., 2017) could imitate the normal data to some extent. However, differences between imitated normal data and original normal data cannot be ignored. In contrast, supervised learning approaches use KPI data labeled by operators to train the model, and take use of both normal and anomalous data, so they are easier to obtain accurate results. Nevertheless, the performance of supervised learning depends largely on the accuracy of labels. Anomaly detection approaches generally consider the given labels as ground truth, and inappropriate labels may have a huge impact on the performance of anomaly detection.

In general anomaly detection scenarios, operators recognize the anomaly in the perspective of events, and labeled anomalies are usually continuous intervals. On the contrary, existing anomaly detection approaches (Liu et al., 2015; Shanbhag & Wolf, 2009; Thi et al., 2018) based on supervised machine learning use individual time points for both training and detection, and thus do not suit to the detection criterion of anomaly intervals. They neither make effective use of interval related labels, nor take advantage of the differences among anomaly points. The main problem is that when the operators label an interval as anomalous, it only means that an anomaly event lasts for a period of time occurs, but it cannot be therefore defined that all points in the interval have the same importance to be regarded as anomaly points. Existing anomaly detection approaches generally ignore the differences among labeled anomalies, but simply use them for training. The use of unprocessed and point-oriented labels reduces the effectiveness of labels. Moreover, ignoring the differences among anomaly points limits the generalization quality of detection approaches.

To solve these problems, a supervised learning approach ALSR (Adaptive Label Screening and Relearning Approach) is proposed in this paper, which provides an idea for interval-oriented anomaly detection. ALSR uses the label screening model, which takes advantage of characteristic of continuous anomaly intervals directly at the algorithm level, to filter the training data. It can be better applied to interval-oriented anomaly detection. Besides, the relearning model based on random forest, which analyzes the differences among detected anomalies in finer granularity, is used for reducing the false positive rate.

The use of ALSR in AIOps is presented in Fig. 1. Historical KPI data are obtained from Web applications, and then operators label the anomaly events in them. The ALSR approach, which consists of four components of feature extraction, label screening model, the main DNN model, and relearning model, uses the historical data for training. While detecting, current KPI data are feed into ALSR and the classification results (0 for normal and 1 for anomalous) are used for anomaly alarm.

The contributions of ALSR can be summarized as follows.

- The label screening model is proposed, which makes use of the different importance of different points within anomaly intervals to filter the training data. It takes the characteristic of continuous anomaly intervals into consideration at the level of algorithm and is better suited to interval-oriented anomaly detection criteria.
- Lightweight relearning algorithm is used in ALSR, which analyzes the difference among detected anomalies. It reclassifies the false positive points and the true positive points, so as to increase the precision rate while do not reduce the recall rate.
- Statistical characteristics and time series models are used for feature extraction. The 12 features carefully selected can be applied to mainstream types of KPIs with different changing characteristics, and moderate scale can well avoid unnecessary costs of redundant features.

The ALSR is proved to work well and stably on three different types of KPIs, and the total performance can reach an F-score of 0.965, which outperforms state-of-the-art supervised anomaly detection approaches.

2. Related works

For anomaly detection of KPIs in time series format, many algorithms of time series analysis can play a role, such as time series decomposition (Chen et al., 2013), filtering algorithm (Nguyen & Goulet, 2018), wavelet analysis (Ömer Aydin & Kurnaz, 2017), and periodic analysis (Laptev et al., 2015). In addition, approaches based on time series prediction (He et al., 2012; Pena et al., 2017; Qi et al., 2018) are used to improve the performance of anomaly detection. The ARIMA algorithm is used by (Krishnamurthy, Sen, Yin, & Yan, 2003) to predict the expected normal traffic pattern and the result is compared with actual traffic, and the difference shows the anomalous activities. Argus (He et al., 2012) uses

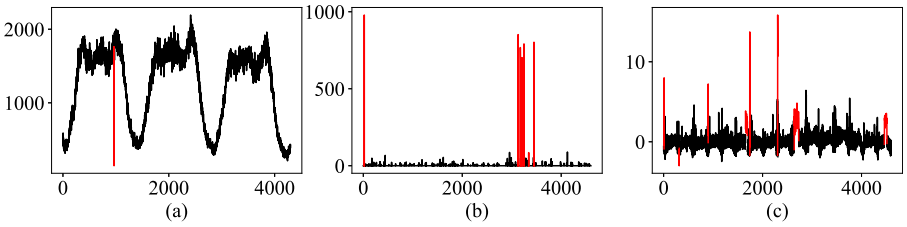

Fig. 2. Examples of strong-seasonal KPI, stable KPI and fluctuant KPI.

the Holt-winters algorithm to detect the change-points of anomalies. However, these algorithms typically have specific assumptions (Ömer Aydin & Kurnaz, 2017) for KPIs, and thresholds need to be determined manually, so it is difficult to propose a universal approach.

Artificial Intelligence for IT Operations (AIOps) based on machine learning is a new field proposed to solve the problems of traditional anomaly detection. Approaches of AIOps no longer require the KPIs to comply with strict rule. Opprentice (Liu et al., 2015) works on the given PV, SR and SRT datasets, and Donut (Xu et al., 2018) can be applied to most seasonal KPIs.

Approaches based on machine learning (Fontugne et al., 2010; Liu et al., 2015; Shanbhag & Wolf, 2009; Thi et al., 2018) usually use outputs of traditional detectors as features, and thus inherit the advantages of multiple time series models. A variety of detectors including time series prediction are used by (Liu et al., 2015) as feature extractors, and the random forest algorithm is used to obtain the final anomaly detection result. A real-time parallel anomaly detection system has been proposed in (Shanbhag & Wolf, 2009), which aggregates normalized anomaly metrics of several algorithms to obtain anomaly scores and has lower false positive rate and false negative rate than separate anomaly detection algorithms.

Existing supervised learning approaches use the unprocessed given labels, which greatly depend on annotation precision, for training, and inappropriate labels affect the performance obviously. Unsupervised learning approaches work with no need of labels, but only the normal data are used. The anomalous data is useless, or even harmful for acquisition of normal mode. In ALSR, differently, the effectiveness of interval related labels is taken into consideration. Both normal and anomaly data are used, while the impact of unimportant anomaly points is weaken by label screening.

In general anomaly detection scenarios, the anomalies are continuous intervals, namely, anomaly events. Existing machine learning approaches using unprocessed labels do not well suit to the interval-oriented anomaly detection. Many detectors (Ömer Aydin & Kurnaz, 2017; Erfani et al., 2016; Fontugne et al., 2010; Hoyle et al., 2018; Laptev et al., 2015; Lavin & Ahmad, 2015; Liu et al., 2015; Miao et al., 2018; Nguyen & Goulet, 2018; Pena et al., 2017; Qi et al., 2018; Shanbhag & Wolf, 2009; Thi et al., 2018) simply use individual points for both training and detection. Donut (Xu et al., 2018) regards the detection of any anomaly point in the interval as the detection of the whole anomaly interval, however, its classifier still use all points in anomaly intervals indiscriminately as the training data. Some other detectors (Alkasassbeh, 2018; He et al., 2012; Tan, Gan, & Shao, 2017) identify the starting and ending points of anomaly intervals. These approaches use only obvious change-points for anomaly detection, and the anomaly points in the middle of anomaly intervals are ignored. ALSR uses anomaly points filtered by label screening model, rather than simple change-points. Besides, the reclassification of detected anomalies in relearning model also analyzes the obviousness of anomaly point in finer granularity than individual points or change-points detection.

3. Problem statement

3.1. KPI anomalies and detection

The KPI data can be collected from system logs, web access logs, SNMP (Simple Network Management Protocol) and other data sources. The KPI data used in this paper are series containing timestamps and values. They have been labeled by operators (0 for normal and 1 for anomalous) for each timestamp.

KPI data can be described as the superposition of normal pattern and occasional anomalous volatilities. The normal patterns of different KPIs have different changing characteristics, so it is impossible to apply a single predefined rule (Alkasassbeh, 2018; Chen et al., 2013; Mahimkar et al., 2010) for anomaly detection to every KPI. Fig. 2 shows three typical types of KPIs which have different changing modes, where anomalies are marked in red.

(a) Strong-seasonal KPI: The values of strong-seasonal KPI vary periodically in terms of weeks, days or hours. The amplitude of periodic variations is much larger than which of random noises.
(b) Stable KPI: The values of stable KPI remain constant for most of the time, though sometimes they contain small random noises.
(c) Fluctuant KPI: Weak-seasonal KPI which contains periodic variations and random noises with similar amplitudes, and random KPI which has only random noises in it are both classified as fluctuant KPI.

Despite the different shapes of KPI curves, it can be intuitively believed that there are many similarities between them. The amplitude of anomalies may be larger than which of noises, and the mean and variance of anomaly points may be different from those of normal points. These similarities make it possible for us to design a detection approach suitable for multi-type KPIs. In this paper, it is assumed that there is no concept drift (Sammut & Harries, 2017) in the KPI data used.

KPI anomaly detection can be described as the following discriminant model: for each timestamp t, the sliding window of current and historical values $\boldsymbol{x_t} = \{x_{t-W+1}, \cdots, x_t\}$ (where W is the length of the sliding window) is used to extract features, and the classification result of y_t ($y_t = 1$ for anomaly points and $y_t = 0$ for normal points) is obtained from the discriminant model.

3.2. Challenges

Points and Intervals of Anomalies. In practice, operators generally focus on anomaly events, which correspond to continuous intervals in KPI data. Two different types of anomaly detection are considered in existing papers. One is change-point detection which aims at the starting and ending points of anomaly, and it is the mainstream in traditional time series models. The other is point-oriented anomaly detection used in most existing machine learning approaches. In change-point detection, points in the middle of anomaly intervals are totally ignored. In point-oriented anomaly detection, every points in anomaly intervals are regarded as equal. Both methods do not make full use of the anomaly points within

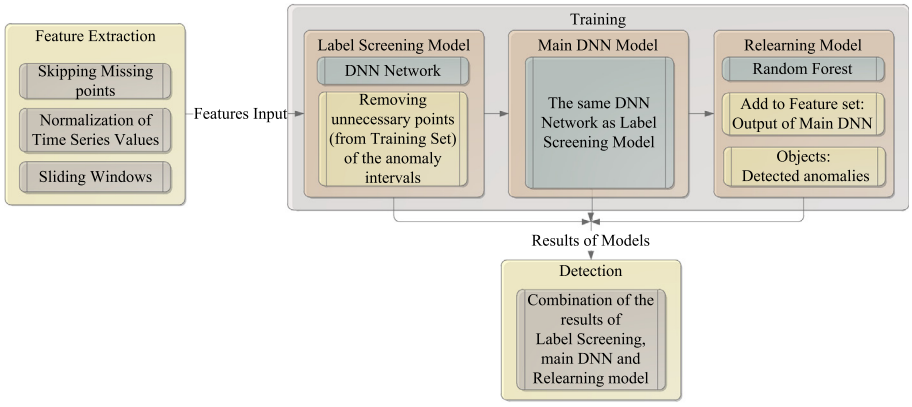

Fig. 3. The architecture of ALSR.

anomaly intervals. Inside intervals labeled as anomalous in KPI data, some points (not only change-points but also some prominent points in the middle of anomaly intervals) have more distinct features, while others are less discriminative and more easily to be confused with normal data. The not so necessary anomaly points cause some redundancy and confusion for the anomaly detection. The label screening model in ALSR is proposed to adaptively distinguish these redundant anomaly points within anomaly intervals. In this way, the difference among anomaly points is fully considered, and as many useful anomaly points as possible are used.

False Positive Points within Detected Anomalies. Detected anomalies are points which are classified as anomalous by detectors. In most cases, there are inevitably some false positive points in detected anomalies. The normal/anomalous classification ability of detector is always limited. That can be partly explained due to the ignoring of differences among detected anomalies. We intuitively guess that the false positive points are less obvious than true positive points. Based on this assumption, the detected anomalies could be reclassified by a new classifier which is more sensitive to true/false positive points than the original detector. In ALSR, the relearning model is proposed to figure out false positive points from true positive points, which works in smaller domain than the normal/anomalous classification model. The precision of detection is increased by using the relearning model.

ALSR approach based on supervised learning is proposed for aforementioned challenges. It makes full use of information of the whole anomaly intervals, and takes the differences among anomaly points into consideration. The label screening model analyzes the characteristics of continuous anomaly intervals, and removes points that are not conducive to interval-oriented anomaly detection. Besides, the relearning model is used to decrease the false positive rate, which pays more attention to the differences among anomaly points.

4. ALSR methodology

4.1. Architecture

The values of KPI data are one-dimensional time series (as described in Section 3.1). Current and historical data $\boldsymbol{x_t} = \{x_{t-W+1}, \cdots, x_t\}$ in sliding window for each time point t are standardized for feature extraction. The features mainly include KPI values, statistical features, time series prediction features and wavelet analysis features.

The overall structure of ALSR is shown in Fig. 3. Firstly, missing points are skipping in data processing, and features are extracted using sliding windows. Then the features are used to train the label screening model. According to the outputs of label screening model, some unnecessary data are removed from training set. The screened training set is used for training of the main DNN model. Then, detected anomalies obtained from the main DNN model are used for training of relearning model, and the classification probabilities that the main DNN model outputs are added to original feature set and used in relearning model. The purpose of relearning is to reduce false positive points from detected anomalies.

The main contributions of ALSR are the adaptive label screening model used for filtering labels and relearning model used for reclassifying detected anomalies.

4.2. Criterion of anomaly detection

Refer to the detection criterion in *iops.ai* (Netman, 2018) (the source of KPI data used in this paper), the following interval-oriented performance criterion is applied to anomaly detection approaches.

For a continuous labeled anomaly interval. If an anomaly detection approach detects the continuous anomaly interval no later than T timestamps after the start of the continuous anomaly interval, it can be defined that the anomaly detection approach successfully detects the entire continuous anomaly interval, so each point of the anomaly intervals is counted as an interval-oriented true positive(TP_{io}) point. Otherwise, each anomaly point in the continuous anomaly interval is counted as a false negative (FN_{io}) point.

For a point which is labeled as normal. If the anomaly detection approach misreports a point which is actually normal to be anomalous, it is counted as a false positive (FP_{io}), otherwise counted as a true negative (TN_{io}).

We use the following equation to calculate precision, recall and F-score of the algorithm.

$$precision = \frac{TP_{io}}{TP_{io} + FP_{io}} \tag{1}$$

$$recall = \frac{TP_{io}}{TP_{io} + FN_{io}} \tag{2}$$

$$F - score = \frac{2 * precision * recall}{precision + recall} \tag{3}$$

Except for F-score, the **AUCPR** (the area under the PR curve) is also used as a measure metric. The more frequently-used AUC of ROC (receiver operating characteristic curve) is very close to 1 due to the large proportion of normal data. Instead, The PR (precision vs recall) curves which more depend on the anomaly data are used in this paper to measure the performance of anomaly detectors. AUCPR ranges from 0 to 1 and larger AUCPR means better

Table 1
The features used in the label screening model and the main DNN model. Some abbreviations are EWMA (Exponentially Weighted Moving-Average), AR (Autoregressive) and Db (Daubechines).

Group	Feature name
Values	The original values standardized.
Statistical features	Mean, standard deviation, range, upper/lower four quantiles, first order difference.
Time series prediction features	EWMA..... (Qi et al., 2018) prediction, AR (Pena et al., 2017) prediction.
Wavelet analysis features	Db2 wavelet (Singh & Mehra, 2017) decomposition.

performance of anomaly detection. The approach whose PR curve is more close to the top right corner is more likely to achieve high recall and precision, and its AUCPR is therefore more close to 1.

A series of thresholds (th) are required to calculate precision and recall several times, and then they are connected as the PR curves. The concrete calculation will be explicated in Section 4.6.

4.3. Data preprocessing and feature extraction

Missing data. Since the missing data are only about 2%~3% in the dataset, we no longer try to take advantage of these points, but directly exclude the impact of them. Windows containing the missing data are discarded during the feature extraction.

Data standardization. The Minmax algorithm is used to standardize the raw values to (0, 1) and then the sliding windows with standardized data are used for feature extraction.

Feature extraction. 12 features used in ALSR are shown in Table 1. Expect for the original values, statistical features of sliding windows are selected to analyze short-term change of KPIs. Time series prediction models represent the likelihood of anomaly to some extent, which have been widely used in traditional anomaly detection. And wavelet features analyze the KPI data in frequency dimension. The performance of these features are further discussed in Section 5.4.

4.4. Label screening model and the main DNN model

4.4.1. Network structure

A DNN (Deep Neutral Network) (Wu, Swietojanski, Veaux, Renals, & King, 2015) with fully connected structure is used as the fundamental model in both the label screening model and the main DNN model.

Three layers of hidden units are contained in the model of DNN, each of which uses ReLU (Wang, Teng, Computer, & University, 2018) as the activation function. The output layer, to the contrary, does not use any activation function.

4.4.2. Training

The training set obtained by feature extraction is used for training the label screening model. After the training of label screening model, the anomaly points regarded as unnecessary will be removed from the original training set.

The results of label screening model can be divided into four categories: True Positive, False Positive, True Negative, and False Negative. These terms are all point-oriented, and can be named as TP_{po}, FP_{po}, TN_{po} and FN_{po} in order to distinguish from the interval-oriented terms in Section 4.2. The differences between point-oriented and interval-oriented criterion are discussed in this section.

TP_{po} and FN_{po} can be presented using Eqs. (4) and (5). Subscript withinT" presents points within T timestamps after the start of anomaly intervals, and afterT" presents which after T timestamps. Subscript tpI" and fnI" respectively present points inside the anomaly intervals which do or do not contain $TP_{po,\ withinT}$ points.

$$TP_{po} = TP_{po,withinT} + TP_{po,afterT} = TP_{po,withinT} + TP_{po,afterT,tpI} + TP_{po,afterT,fnI} \quad (4)$$

$$FN_{po} = FN_{po,withinT} + FN_{po,afterT} = FN_{po,withinT,tpI} + FN_{po,withinT,fnI} + FN_{po,afterT,tpI} + FN_{po,afterT,fnI} \quad (5)$$

Similarly, the components of TP_{io} and FN_{io} as well as their differences from TP_{po} and FN_{po} can be described as:

$$TP_{io} = TP_{po,withinT} + TP_{po,afterT,tpI} + FN_{po,withinT,tpI} + FN_{po,afterT,tpI} = TP_{po} + FN_{po,withinT,tpI} + FN_{po,afterT,tpI} - TP_{po,afterT,fnI} \quad (6)$$

$$FN_{io} = FN_{po,withinT,fnI} + FN_{po,afterT,fnI} + TP_{po,afterT,fnI} = FN_{po} + TP_{po,afterT,fnI} - FN_{po,withinT,tpI} - FN_{po,afterT,tpI} \quad (7)$$

FP and TN are simply the same in point-oriented and interval-oriented criterion, that is:

$$FP_{io} = FP_{po}, \quad TN_{io} = TN_{po} \quad (8)$$

In interval-oriented criterion, detection of TP_{po} points within T timestamps after the start of the anomaly represents the detection of the whole interval, while TP_{po} points detected later than T timestamps are ignored. Therefore, some FN_{po} points (which are inside the anomaly intervals with at least one anomaly point detected within T timestamps after the start of the anomaly interval) are added to TP_{po}, while other TP_{po} points (which are detected later than T timestamps after the start of the anomaly interval) are removed from TP_{po}. In this way, TP_{io} points are built up, as described in Eq. (6) (the second line). According to this equation, if some TP_{po} points change to $FN_{po,\ withinT,\ tpI}$ or $FN_{po,afterT,tpI}$ points, it has no effect to TP_{io} points. But in Eq. (6) (the first line) and Eq. (7) (the first line), it can be seen that change of $TP_{po,\ withinT}$ to $FN_{po,\ withinT,\ fnI}$ will be synchronized to TP_{io} and FN_{io}, and therefore has adverse impact to interval-oriented detection.

Based on the above analysis, $FN_{po,\ withinT,\ tpI}$ and $FN_{po,afterT,tpI}$ points have no impact in the detection of anomaly intervals, as for they are regarded as false negative in point-oriented anomaly detection but true positive in interval-oriented anomaly detection. These points are certainly not so distinguishing as other anomaly points. Among them, we assume that $FN_{po,\ withinT,\ tpI}$ points represent anomaly interval not easy to detected and should be reserved. However, $FN_{po,afterT,tpI}$ are regarded as unrepresentative points inside anomaly intervals, and we try to reduce their impacts. Label screening is proposed according to this assumption.

The algorithm of label screening is described in Algorithm 1. Two methods following are adopted in it to improve performance and avoid adverse effects.

A. According to the analysis above, only $FN_{po,afterT}$ points, while $FN_{po,\ withinT}$ excluded, are regarded as unnecessary anomaly points and will be removed from the original training set.
B. The screened training set is used in the training of the main DNN model, but the direct output of label screening model which uses the unscreened training set is also reserved. Combining the results of both models is proved to both guarantee the effectiveness of label screening and make full use of the anomaly points of small amount.

Algorithm 1 Label screening.

Input: The feature vector $\boldsymbol{f_t}$ and the corresponding label l_t.

```
//label screening
for each f_t and l_t in training set do
    Train f_t and l_t in label screening model
end for
for each f_t and l_t in training set do
    Classify f_t using label screening model and get the classification result y_t
end for
for each y_t in result of training set do
    Index y_t with the order i_t in its real anomaly interval
    if y_t ∈ FN_po and i_t > T then
        Remove f_t and l_t from training set
    end if
end for
//main DNN
for each f_t and l_t in training set (screened) do
    Train f_t and l_t in main DNN model
end for
```

4.5. Relearning

The inputs of both the label screening model and the main DNN model are large amounts of normal data and much less anomalous data, and the target of classification is to distinguish anomalies from normal points. Differently, the input of relearning is points classified as anomaly by the main DNN model. Most of these detected anomalies correspond to real anomalies, but some of them are undesirable false positive points. According to the general understanding, the former should be more prominent than the later, and there are discernible differences between them, which provide possibility for us to apply the relearning algorithm.

The relearning model in ALSR mainly uses TP_{po} and FP_{po} points (collectively called detected anomalies) obtained from results of the main DNN model as training set. This model aims at true/false positive classification. Ideally, all FP_{po} points should be detected while no TP_{po} points affected. In practice, however, the reducing of FP_{po} points also make a little TP_{po} points change to FN_{po} points. This problem is alleviated in interval-oriented anomaly detection. As described in Section 4.4, except for $FN_{po,\ withinT,\ fnI}$, the other $FN_{po,\ withinT,\ tpI}$, $FN_{po,afterT,tpI}$ and $FN_{po,afterT,fnI}$ will basically not affect the operators' judgement of the entire anomaly event, because $FN_{po,\ withinT,\ tpI}$, $FN_{po,afterT,tpI}$ are classified as TP_{io} in interval-oriented anomaly detection, and $FN_{po,afterT,fnI}$ is totally affected by $FN_{po,\ withinT,\ fnI}$.

For the obtaining of training set, sometimes there are a lot of problems related to quantity, such as small amount of data, unbalanced proportion and so on, because the number of TP_{po} and FP_{po} points obtained from each KPI varies. The data needed for relearning are therefore randomly sampled according to determined amount. In experiments, some TN_{po} points are used as an auxiliary class for FP_{po} in order to improve the performance, they are both seen as normal (0) class and only TP_{po} points are used as anomalous (1) class.

The *relearning training set* is sampled to a combination as shown in Eq. (9). The *randomof* function means random sampling with replacement from the specific set, and it follows uniform distribution. The *shuffle* function means disturbing the order of the set. And C is a specific constant. We use $C = 500$ in this paper. In fact, the sizes of TP_{po} and FP_{po} in each KPI are a few dozen to a few hundred, and $C = 500$ can achieve sufficient sampling. The total size of *relearning training set* is only 3000 and the relearning process is very fast.

$$\begin{aligned} &relearning\ training\ set = \\ &\quad shuffle\{4C * randomof(TP_{po}) \\ &\quad + C \cdot randomof(FP_{po}) + C \cdot randomof(TN_{po})\} \end{aligned} \tag{9}$$

After some experiments and comparisons, the relearning model finally uses RF (Random Forest) algorithm (Ayyadevara, 2018) for classification. RF is a kind of ensemble algorithm. Compared with multi-layer neural network structure, it has the advantages of fast inference speed and high classification accuracy, and can achieve better performance in relearning process.

The outputs of the main DNN model are classification probabilities. They are used to make a normal/anomalous classification. The classification probabilities describe both the obviousness of the anomaly points and the randomness of the learning process. The two types of information are needed for relearning. Therefore, the feature set of relearning is the original feature set added with the classification probabilities output from the main DNN.

The algorithm of relearning is described in Algorithm 2.

Algorithm 2 Relearning.

```
for each f_t and l_t in training set (screened) do
    Classify f_t using main DNN model and get the classification result y_{t,main}
end for
Add the classification probabilities output from the main DNN model to features of training set (screened)
Sample randomly from TP_po, FP_po and TN_po sets of the results of main DNN model using a specific radio (which is 4:1:1 in this paper), get the relearning training set
for each f_t' and l_t' in relearning training set do
    Train f_t' and l_t' in relearning model
end for
```

4.6. Detection

For a sliding window $\boldsymbol{x_t} = \{x_{t-W+1}, \cdots, x_t\}$ to be detected, the purpose of the anomaly detection is to obtain the detection result y_t (0 for normal and 1 for anomalous), and the classification probability p_{y_t} can be used to describe how likely an anomaly occurs at time t. p_{y_t} ranges from 0 to 1 and larger p_{y_t} means greater probability of anomaly.

The classification probability $p_{y_t,ls}$, $p_{y_t,main}$, and $p_{y_t,re}$ of the test set as well as the corresponding detection results $y_{t,ls}$, $y_{t,main}$ and $y_{t,re}$ can be respectively obtained from the model of label screening, main DNN and relearning. Experiments on all datasets prove that $y_{t,main}$ has a significantly higher precision and a constant or slightly decreased recall than $y_{t,ls}$. However, $y_{t,main}$ is obtained through the label screening model which reduces the amount of anomalous data and sometimes does not represent the total dataset, thus the precision and recall if individual KPIs declined slightly. To reduce errors, $y_{t,ls}$ are preserved, and then executed and operation (&) with $y_{t,main}$. This method achieves decreased number of KPIs adversely affected, and the results of both label screening and main DNN model are preserved. The overall performance is proved better.

The anomaly points detected by the label screening model and the main DNN model where $y_{t,ls}$ & $y_{t,main} = 1$ can be revised again using the relearning result $y_{t,re}$. The relearning model focuses on finding out some false positive points from them. This procedure is described in Algorithm 3. For the normal points detected where $y_{t,ls}$ & $y_{t,main} = 0$, $y_{t,re}$ is meaningless, and it is output only for the simplicity of code implementation.

Table 2
Network parameters and experiment settings used in this paper.

Global parameter	Delay time T allowed in anomaly interval detection (see Section 4.2)	12
Label screening and main DNN structure	Sliding window length W	11
	Input dimension	12
	Hidden layer	3 layers, the numbers of nodes are 128, 128 and 64 respectively
	Activation function (Hidden layers)	ReLU
	Output dimension	2
	Learning rate	0.0001
	Epochs	100
	Batch size	50
	Optimizer	Adam (Kingma & Ba, 2014)
	Loss function	Cross entropy (Bernard, Bondarenko, & Vanduffel, 2018)
Relearning Model, RF	Sample constant C	500
	Number of input features	14
	Output	classification probabilities of 2 categories (0, 1)

Algorithm 3 Anomaly detection.

for each time point t in *test set* **do**
  if $y_{t,ls}$ & $y_{t,main} = 1$ and $y_{t,re} = 0$ **then**
    Make the final detection result $y_t = 0$
  else
    $y_t = y_{t,ls}$ & $y_{t,main}$
  end if
end for

According to the analysis above, the detection result y_t can be computed by Eq. (10):

$$y_t = y_{t,ls} \ \& \ y_{t,main} \ \& \ y_{t,re} \tag{10}$$

To draw the PR curves, a variable y_t related to thresholds (th), that is, $y_t(th)$ is required. In general situations, $y_t(th)$ is computed using Eq. (11), where the sig function is described in Eq. (12). In this case, p_{y_t} is obtained from the output of the anomaly detector and does not vary with th.

$$y_t(th) = sig(p_{y_t}, th) \tag{11}$$

$$sig(a, th) = \begin{cases} 1, & a \geq th \\ 0, & a < th \end{cases} \tag{12}$$

In this paper, however, the three models in ALSR all have their own classification probability output, and decide the final result together. So not only y_t but also p_{y_t} are varying with th, that is $y_t(th)$ and $p_{y_t}(th)$. So we give an approach described in Eqs. (13) and (14) to compute $p_{y_t}(th)$ and $y_t(th)$ in accordance with the meaning of the outputs of three models.

$$\begin{aligned} p_{y_t}(th) = & (1 - sig(p_{y_t,ls}, th)) \cdot (p_{y_t,ls}) \\ & + sig(p_{y_t,ls}, th) \cdot (1 - sig(p_{y_t,main}, th)) \cdot p_{y_t,main} \\ & + sig(p_{y_t,ls}, th) \cdot sig(p_{y_t,main}, th) \cdot p_{y_t,re} \end{aligned} \tag{13}$$

$$y_t(th) = sig(p_{y_t}(th), th) \tag{14}$$

5. Evaluation

5.1. Datasets

The datasets used in this paper are collected from the public datasets of *iops.ai* official website (Netman, 2018) (*first session of AIOps Competition: preliminary training set, final training set*). These KPI data are collected from big Internet companies and related data have been used in many most advanced papers in the field of KPI anomaly detection. In this paper, 25 KPIs that match the description of strong-seasonal KPI, stable KPI and fluctuant KPI in Section 3.1 are chosen for experiments. The number of anomaly intervals in selected KPIs should not be very small. The KPIs are divided into 70% of training set and 30% of test set.

5.2. Experiment setup

The parameters and settings of the label screening, main DNN and relearning model, as well as other global parameters are shown in Table 2.

5.3. Overall performance

Two main indicators, F-score and AUCPR, are used to measure the detection performance, and the following classifiers are compared.

Basic SVM (Support Vector Machine), LSTM (Long Short-Term Memory) and RF (Random Forest) algorithms. (To present the effect of ALSR architecture, the same 12 features are used in ALSR, SVM, LSTM and RF approaches for comparison in this section, and the effect of feature set will be later discussed in Section 5.4.) The parameters of these basic classifiers are shown in Table 3.

Opprentice (Liu et al., 2015) is a supervised learning approaches that uses 14 basic extractors to extract features and uses RF as classifier.

LogicMonitor-AI Anomaly Detection LogicMonitor is the approach proposed by the champion of the AIOps Competition, which is based on DNN, and uses sliding windows for feature extraction.

The 25 datasets used in experiments are divided into three groups, namely, strong-seasonal KPI (which have 12 KPIs), stable KPI (6), and fluctuant KPI (7). Approaches are experimented respectively in the three groups, and the PR curves are presented in Fig. 4.

The PR curves of ALSR on strong-seasonal KPI and stable KPI are closest to the top right corner, while the PR curves of LSTM and RF perform a little better on fluctuant KPI. The results indicate that ALSR achieves higher or at least nearly equal accuracy than the approaches compared, and it works more stably on different types of KPIs.

Fig. 5 shows the AUCPRs of all approaches compared, and Fig. 6 the F-scores. ALSR has greater advantage using the indicator of F-score. The F-score of 0.965 and the AUCPR of 0.978 in total that ALSR reaches outperform the results of other contrast approaches.

Table 3
Parameters in SVM, LSTM and RF classifiers.

SVM	Kernel	RBF(Radial Basis Function)
	Regularization constant	1
	Learning algorithm	SMO(Sequential Minimal Optimization)
RF	Number of Decision Trees	100
	Max features/ Max Depth/ Min Samples Leaf/ Max Leaf Nodes	Auto/ No limited
LSTM	Time Step	6
	Activation function	ReLU
	Loss	Mean Squared Error
	Optimizer	ReLU
	Batch size	5000
	Epochs	100

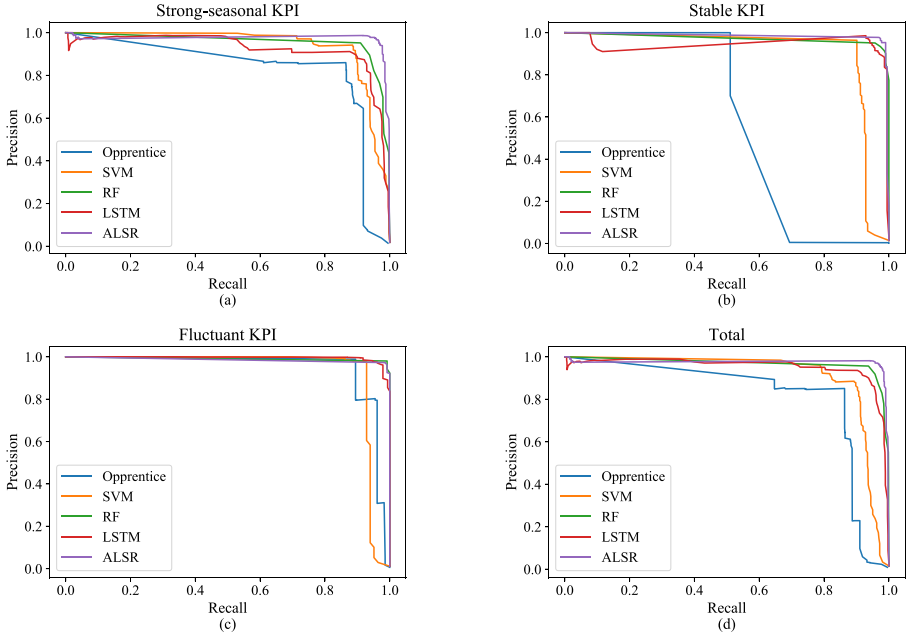

Fig. 4. The PR curves using Opprentice, SVM, LSTM, RF, LogicMonitor and ALSR separately on datasets of strong-seasonal, stable, fluctuant and total KPIs.

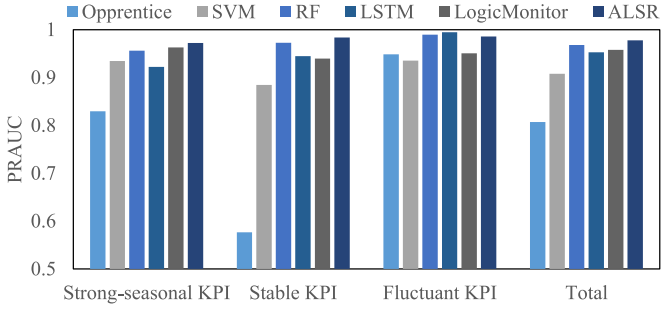

Fig. 5. The AUCPRs of Opprentice, SVM, LSTM, RF, LogicMonitor and ALSR.

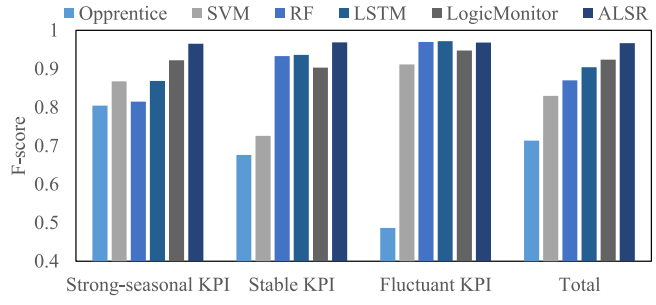

Fig. 6. The F-scores of Opprentice, SVM, LSTM, RF, LogicMonitor and ALSR.

Table 4
The F-scores of Opprentice, SVM, LSTM, RF, LogicMonitor and ALSR separately using feature set 1, 2 and 3.

	Feature set 1	Feature set 2	Feature set 3
Opprentice	0.851402	0.729836	0.873313
SVM	0.068554	0.566726	0.618743
RF	0.847268	0.760565	0.877893
LSTM	0.644807	0.659509	0.894281
LogicMonitor	0.795627	0.653428	0.923542
ALSR	0.808177	0.654177	0.965109

5.4. Feature selection

The feature set used in ALSR is shown in Table 1, and compared with other two feature sets in this Section. **Feature set 1** is the values obtained from the basic sliding windows. **Feature set 2** is used in (Shanbhag & Wolf, 2009) and contains the features extracted by HW (Holt Winter Forecasting Model), ADAP (Adaptive Threshold Algorithm), AVG (Average over Window), EWMA (Exponential Weighted Moving Average) and CUSUM (Cumulative Sum Algorithm). Features used in ALSR are regarded as **Feature set 3**.

Fig. 7 (where A,B and C respectively represent strong-seasonal KPI, stable KPI and fluctuant KPI) compares the F-scores separately using three feature sets. As it is shown, feature set 1 and 2 cannot handle some of the KPIs and get quite low F-scores, while feature set 3 works well on most datasets.

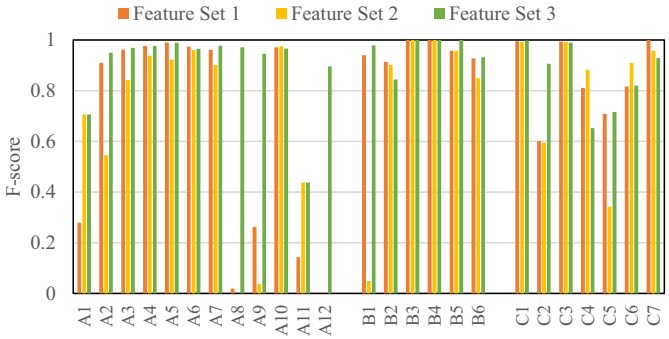

Fig. 7. The F-scores of ALSR separately using feature set 1, 2 and 3 on three types of KPIs.

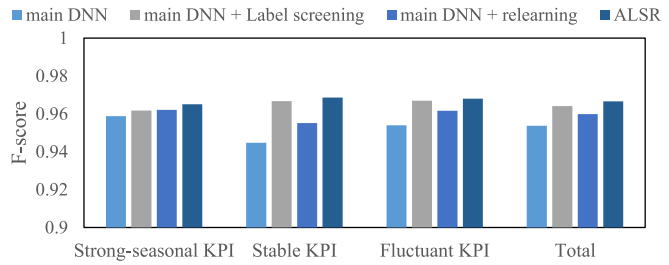

Fig. 8. The F-scores using the techniques of main DNN alone, main DNN + Label screening, main DNN + relearning, ALSR with both techniques, separately on strong-seasonal, stable, fluctuant KPIs and the total datasets.

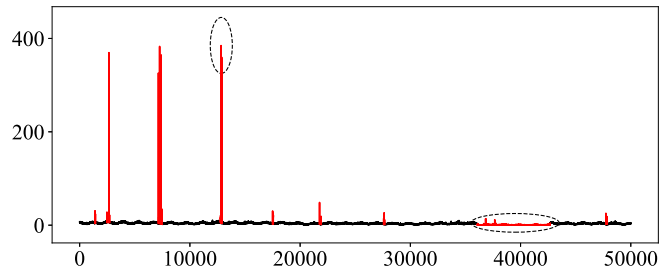

Fig. 9. Two types of anomaly intervals which are circled on a stable KPI.

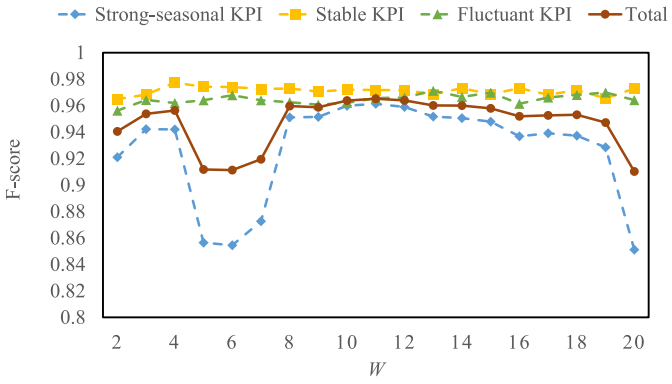

Fig. 10. The F-scores of ALSR using different Sliding Window Length *W*, separately on strong-seasonal, stable, fluctuant KPIs and the total datasets.

The effect of feature set has also been experimented in other approaches besides ALSR. Table 4 presents the F-scores using feature set 1, 2 and 3 in each of Opprentice, SVM, RF, LSTM, LogicMonitor and ALSR approaches. In all these approaches, it can be seen that feature set 3 outperforms the other two feature sets.

One phenomenon is that RF performs better than ALSR in feature set 1 and 2, that may because of the interval-oriented detection of ALSR. The criterion used in this paper (Section 4.2) allows the detection of anomaly to have some delay. This characteristic is implicitly used in feature set 3. Several features (mean, std, range, etc.) are computed using sliding windows and make it easier to detect anomaly which happened a few timestamps ago. Such features is significantly more in feature set 3 than in feature set 1 and 2. Besides, in relearning model, the permission of delay is considered in points which are both FN_{po} and TP_{io} (Section 4.5). This method is proposed to improve the accuracy of interval-oriented anomaly detection with limited delay. Feature set 1 and 2 cannot provide corresponding features for relearning, and the focusing on interval anomalies is not effective in them.

In summary, it is proved that feature set 3 takes effect in mainstream types of KPIs and different approaches. It helps ALSR to achieve better performance on total 25 KPIs.

5.5. Effect of label screening and relearning

Two main techniques have been proposed to improve the performance in this paper: adaptive label screening and relearning. Fig. 8 presents the F-scores of the main DNN model and three possible combinations of these techniques, separately on strong-seasonal, stable, fluctuant KPIs and the total datasets.

The adaptive label screening makes larger contribution, and the stable KPIs are most affected. Actually, two types of anomaly intervals in stable KPIs have been observed, one is long but smooth, while the other is short but sharp, as shown in Fig. 9. (The same phenomenon exists in seasonal KPIs and the fluctuant KPIs but much less.) Too long anomaly intervals are partly unnecessary, and only some specific points within the intervals indicate anomaly. In these cases, the label screening model naturally works to remove most unnecessary points and preserves the more important points. It is a typical example for label screening to handle long anomalies.

The relearning model contributes less than the label screening model. That may due to the smaller domain (only detected anomalies) it works in. In practice, there are sometimes even no false positive points, and the relearning model is skipped in such cases. Despite this, the lightweight relearning model works stably on most situations with false positive points and provides some fast improvement for the total system. The result shows that even though the normal/anomalous classification has passed, the detected anomalies still contains enough information for us to distinguish the false positive from the true positive. This conclusion provides inspiration for future anomaly detection approaches on finer granularity.

ALSR with both techniques performs best, and it is proved that the label screening and relearning approaches are able to be combined for greater improvement.

5.6. Impact of sliding window length W and delay time T

Sliding windows of length *W* are used to extract features from KPI data. Appropriate window length plays an important role in anomaly detection. Short sliding windows cannot derive the relationship of adjacent points, while long sliding windows are so dependent on historical information that lack sensitivity to the current value. Fig. 10 shows the variation of the detection results with the length *W* of the sliding windows. It can be seen that strong-seasonal KPI is most affected by *W*. On the contrary, stable KPI and fluctuant KPI do not vary significantly with *W* less than 20. We finally choose the appropriate sliding window length $W = 11$ in this paper.

Delay time *T* allowed in anomaly interval detection is another major parameter of ALSR. Ideally, an anomaly should be alarmed

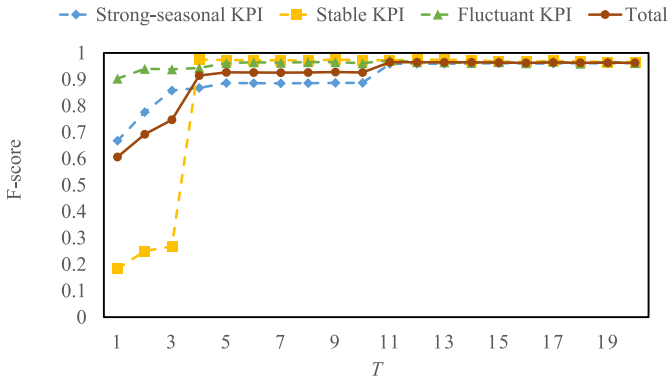

Fig. 11. The F-scores of ALSR using different Delay Time T, separately on strong-seasonal, stable, fluctuant KPIs and the total datasets.

as soon as it occurs. In practice, however, there are usually some delays while detecting. Larger delay time T allowed intuitively achieves better accuracy of anomaly detection. Fig. 11 is drawn to present how the total F-score is affected by T. The F-score rapidly raises with the increasing of T in range of (1,5). Between 5 and 12 of T, the F-score slowly increases a little, and remains constant when T varies larger than 12. That means too large T is also meaningless, because a few anomalies are actually not detected at all whichever T is chosen.

According to these observations, it's recommended to set T less than 5, and the parameter used in this paper is $T = 12$.

6. Conclusion

High accuracy detection approaches in the perspective of events are required by operators in practice, and interval-oriented anomaly detection is more likely to obtain higher accuracy than simple point-oriented anomaly detection.

The ALSR approach proposed in this paper mainly uses two techniques to make full use of anomaly points within anomaly intervals. The label screening model distinguishes important anomaly points within intervals, and removes the unnecessary points from training set. The relearning model emphasizes the differences among detected anomalies and reclassifies them in finer granularity. The false positive rate is decreased and the adverse effect can be minimized in interval-oriented anomaly detection.

The diversity of KPIs makes the anomaly detection more difficult. Although machine learning approaches have better universality compared with traditional time series models, high accuracy is usually achieved on specific type of KPI. A DNN with full connected structure is used as the foundation of ALSR, it has high accuracy and is not restricted by a specific form of KPI. The label screening and relearning model are connected to it for interval-oriented anomaly detection and better performance. The ALSR approach is proved to work stably on three mainstream types of KPIs. The total F-score on all test sets has reached 0.965 and the AUCPR has reached 0.978. The results outperform state-of-the-art machine learning approaches.

In the application scenario of AIOps, ALSR is mainly used for the detection and alarm of continuous anomaly intervals. Operators do not need to tune the parameters while using ALSR, which works on mainstream types of KPIs in an adaptive and automated way. ALSRs multistage structure can achieve a high accuracy in interval-oriented anomaly detection, and it helps operators to catch the anomaly intervals correctly while reduce the false positive rate at the same time.

Conflict of interest

The authors have declared that no conflict of interest exists.

Credit authorship contribution statement

Jingyu Wang: Conceptualization, Methodology, Resources, Writing - review & editing. **Yuhan Jing:** Conceptualization, Data curation, Formal analysis, Methodology, Investigation, Software, Writing - original draft. **Qi Qi:** Conceptualization, Formal analysis, Validation, Writing - review & editing. **Tongtong Feng:** Formal analysis, Investigation, Validation. **Jianxin Liao:** Project administration, Supervision, Funding acquisition.

Acknowledgments

This work was jointly supported by: (1)National Natural Science Foundation of China (no. 61771068, 61671079, 61471063); (2)Beijing Municipal Natural Science Foundation (no. 4182041); (3)Fundamental Research Funds for the Central Universities (no. 2018RC20).

References

Alkasassbeh, M. (2018). A novel hybrid method for network anomaly detection based on traffic prediction and change point detection. *Journal of Computer Science, 14*(2), 153–162.
Ömer Aydin, & Kurnaz, M. (2017). Wavelet-based anomaly detection on digital signals. *Signal processing & communications applications conference*.
Ayyadevara, V. K. (2018). *Random Forest*.
Bernard, C., Bondarenko, O., & Vanduffel, S. (2018). Rearrangement algorithm and maximum entropy. *Annals of Operations Research, 261*(1–2), 107–134.
Chen, Y., Mahajan, R., Sridharan, B., & Zhang, Z. L. (2013). A provider-side view of web search response time. *ACM SIGCOMM Computer Communication Review, 43*(4), 243–254.
Dao, C., Liu, X., Sim, A., Tull, C., & Wu, K. (2018). Modeling data transfers: Change point and anomaly detection. *Ieee international conference on distributed computing systems*.
Erfani, S. M., Rajasegarar, S., Karunasekera, S., & Leckie, C. (2016). High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recognition, 58*(C), 121–134.
Fontugne, R., Borgnat, P., Abry, P., & Fukuda, K. (2010). Mawilab: Combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking. *International conference*.
Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Bing, X., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative adversarial nets. *International conference on neural information processing systems*.
He, Y., Flavel, A., Ge, Z., Gerber, A., Dan, M., Papadopoulos, C., … Yates, J. (2012). Argus: End-to-end service anomaly detection and localization from an ISP's point of view. *IEEE infocom*.
Hoyle, B., Rau, M. M., Paech, K., Bonnett, C., Seitz, S., & Weller, J. (2018). Anomaly detection for machine learning redshifts applied to SDSS galaxies. *Monthly Notices of the Royal Astronomical Society, 450*(1), 305–316.
Kim, D., Yang, H., Chung, M., & Cho, S. (2017). *Squeezed convolutional variational autoencoder for unsupervised anomaly detection in edge device industrial internet of things*.
Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. *Computer Science*.
Krishnamurthy, E., Sen, S., Yin, Z., & Yan, C. (2003). Sketch-based change detection: Methods, evaluation, and applications. *Proc ACM-usenix internet measurement conference*.
Laptev, N., Amizadeh, S., & Flint, I. (2015). *Generic and scalable framework for automated time-series anomaly detection*.
Lavin, A., & Ahmad, S. (2015). *Evaluating real-time anomaly detection algorithms - the numenta anomaly benchmark*.
Liu, D., Zhao, Y., Xu, H., Sun, Y., Dan, P., Jiao, L., … Mei, F. (2015). Opprentice:towards practical and automatic anomaly detection through machine learning. *Internet measurement conference*.
LogicMonitor-AI-Team (2018). *Logicmonitor-ai anomaly detection*. http://workshop.aiops.org/files/logicmonitor2018.pdf. Accessed December 10, 2018.
Mahimkar, A. A., Han, H. S., Ge, Z., Shaikh, A., Jia, W., Yates, J., … Emmons, J. (2010). Detecting the performance impact of upgrades in large operational networks. *ACM SIGCOMM conference*.
Miao, X., Liu, Y., Zhao, H., & Li, C. (2018). Distributed online one-class support vector machine for anomaly detection over networks. *IEEE Transactions on Cybernetics, PP*(99), 1–14.
Netman (2018). *Kpi anomaly detection*. http://iops.ai/competition_detail/?competition_id=5&flag=1. Accessed December 10, 2018.

Nguyen, L. H., & Goulet, J. A. (2018). Anomaly detection with the switching Kalman filter for structural health monitoring. *Structural Control & Health Monitoring, 25*(2), e2136.

Pena, E. H. M., Assis, M. V. O. D., & Proenca, M. L. (2017). Anomaly detection using forecasting methods arima and HWDS. *International conference of the Chilean computer science society*.

Qi, J., Chu, Y., & He, L. (2018). Iterative anomaly detection algorithm based on time series analysis. *2018 ieee 15th international conference on mobile ad hoc and sensor systems (mass)*.

Sammut, C., & Harries, M. (2017). *Concept drift*.

Shanbhag, S., & Wolf, T. (2009). Accurate anomaly detection through parallelism. *Network IEEE, 23*(1), 22–28.

Singh, H., & Mehra, R. (2017). Discrete wavelet transform method for high flux n-γ discrimination with liquid scintillators. *IEEE Transactions on Nuclear Science, 64*(99). 1–1.

Tan, X., Gan, T. Y., & Shao, D. (2017). Effects of persistence and large-scale climate anomalies on trends and change points in extreme precipitation of canada. *Journal of Hydrology, 550*, 453–465.

Thi, N. N., Cao, V. L., & Le-Khac, N. A. (2018). *One-class collective anomaly detection based on long short-term memory recurrent neural networks*.

Wang, D. P., Teng, G., Computer, S. O., & University, J. N. (2018). Optimal design of relu activation function in convolutional neural networks. *Information & Communications*.

Wu, Z., Swietojanski, P., Veaux, C., Renals, S., & King, S. (2015). *A study of speaker adaptation for DNN-based speech synthesis*.

Xu, H., Chen, W., Zhao, N., Li, Z., Bu, J., Li, Z., … Yang, F. (2018). *Unsupervised anomaly detection via variational auto-encoder for seasonal KPIS in web applications*.