An Adaptive Approach for Anomaly Detector Selection and Fine-Tuning in Time Series

Hui Ye
Alibaba Inc
Beijing, China
yehui.yh@alibaba-inc.com

Xiaopeng Ma
Alibaba Inc
Beijing, China
xiaopeng.mxp@alibaba-inc.com

Qingfeng Pan
Alibaba Inc
Beijing, China
qingfeng.pqf@alibaba-inc.com

Huaqiang Fang
Alibaba Inc
Beijing, China
huaqiang.fhq@alibaba-inc.com

Hang Xiang
Alibaba Inc
Beijing, China
xingzhi.xh@alibaba-inc.com

Tongzhen Shao
Alibaba Inc
Beijing, China
yeqing.stz@taobao.com

ABSTRACT

The anomaly detection of time series is a hotspot of time series data mining. The own characteristics of different anomaly detectors determine the abnormal data that they are good at. There is no detector can be optimizing in all types of anomalies. Moreover, it still has difficulties in industrial production due to problems such as a single detector can't be optimized at different time windows of the same time series. This paper proposes an adaptive model based on time series characteristics and selecting appropriate detector and run-time parameters for anomaly detection, which is called ATSDLN(Adaptive Time Series Detector Learning Network). We take the time series as the input of the model, and learn the time series representation through FCN. In order to realize the adaptive selection of detectors and run-time parameters according to the input time series, the outputs of FCN are the inputs of two sub-networks: the detector selection network and the run-time parameters selection network. In addition, the way that the variable layer width design of the parameter selection sub-network and the introduction of transfer learning make the model be with more expandability. Through experiments, it is found that ATSDLN can select appropriate anomaly detector and run-time parameters, and have strong expandability, which can quickly transfer. We investigate the performance of ATSDLN in public data sets, our methods outperform other methods in most cases with higher effect and better adaptation. We also show experimental results on public data sets to demonstrate how model structure and transfer learning affect the effectiveness.

KEYWORDS

Self-adaption, Anomaly Detection, Joint Learning Network, Transfer Learning, Time Series

DLP-KDD'19, August 5, 2019, Anchorage, AK, USA

ACM Reference Format:
Hui Ye, Xiaopeng Ma, Qingfeng Pan, Huaqiang Fang, Hang Xiang, and Tongzhen Shao. 2019. An Adaptive Approach for Anomaly Detector Selection and Fine-Tuning in Time Series. In *1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data (DLP-KDD'), August 5, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 7 pages.

1 INTRODUCTION

Internet-based services have strict requirements for continuous monitoring and in-time anomaly detection, Specifically, monitoring performance ability and detecting performance anomalies are important. Such as, e-commerce platforms need to monitor income index and broadcast alert when obvious income decrease happens.

From the perspective of data science, key performance indexes are usually portrayed as time series, and potential faults in application are portrayed as anomaly. An anomaly (An outlier) in time series, is a data point or a group of data points which significantly different from the rest of the data points[8]. Due to the large amounts of performance indexes and anomalies, human monitoring of these indexes is impracticable which leads the demand for automated anomaly detection using Machine Learning and Data Mining techniques[6, 10–12]. Many fast and effective anomaly detectors were designed to localize these anomalies[2], such as outlier detector[1], change point detector[7]. Although anomaly detectors have proven effective in certain scenarios, applying them to internet-based services remains a great challenge[9]. Due to the large-scale distributed monitoring vision and complex trends of indicators, it's almost impossible to detect anomalies in all scenarios with one type of detector. In order to ensure the performance of the anomaly detection approach, expertise-based rules are required for detector selection and run-time parameters fine-tuning[9]. Furthermore, when a detector system is deployed online, the run-time parameters of anomaly detector are usually required to adjust according to real-time changes.

It's hard to propose one general approach to detect all types of anomaly, such as significant decrease or increase can be detected by static threshold directly, continuous minor changing can be detected by change point detector more quickly. State-of-the-art detectors are usually designed to detect one type of anomaly[8]. When the multi-detector detection result voting method is adopted, each detect needs to traverse all detectors and candidate run-time

parameters combinations. The effect is greatly influenced by the data set and voting rules and it is very time-consuming, which do not meet the demands of industrial real-time monitoring scenarios. Our proposed framework named ATSDLN, tackles the above challenges through an adaptive time series anomaly detector learning network.

2 METHODS

Under the background of large industrial data scale, complicated index system and an unusually large variety, on the one hand, time series data usually changes with business changes. The same time sequence may have great differences in different stages of business projects; on the other hand, influenced by commercial data and users' behaviors, there are different low ebbs of the peak flow on holidays, daytime and nights, big promotions and so on, which cause the natural differences in data. If we do not consider self-adaption when doing anomaly detection, we cannot balance between the false positive rate and the false negative rate. Therefore, choosing a universal detector to adapt to all data and scenarios is unworkable. Multi-detection algorithm fusion is a very effective method to improve the time series anomaly detection field, which is usually conducted in the two stages as follows:

`The anomaly detection stage`: it is realized by selecting the appropriate detector for the time series of the input.

`The alarm convergence stage`: it is realized by using the abnormality that is detected by each detector as the input. The alarm convergence can be achieved with the method of voting or time series feature modeling.

- `Voting method`: absolute majority vote, relative majority vote, weighted vote, etc.
- `Deep learning`: time series modeling of the detected anomalies.

Both of them are of highly expandability and support dynamic expansion of anomaly detectors. The former is self-adaption based on the original time series of the input, which is more flexible, this study takes the former. As is shown in the experimental chapter, the single detector is lower than our model in term of the accuracy, recall, and f1, and the error rate is relatively high. The starting point of this study is to set a certain sliding window size for the time series, and optimize the accuracy, recall and false positive rate of the anomaly detection through using the detector and run-time parameters for the self-adaption selection of the current sliding window time series.

Since different detectors have their own characteristics which determine the type of time series they are good at, it is natural to think about to determine which the detectors and run-time parameters are suitable for by the features of the time series. We call this way the manual rule maintenance detector and run-time parameters selection. The core work is to determine what features of time series and what threshold should be used for judgment (for instance, non-stationary time series with long-term trends can adopt dynamic thresholds). The advantage of such artificial rules is that it has strong interpretability. However, it is true that the determination of these rules relies on manual experience, which is difficult to enumerate the rules. As the data accumulation rules become more and more difficult to maintain, the abnormal coverage, correctness, versatility and expandability of the rules are also great challenges.

Fortunately, in the era of artificial intelligence, it is natural to think of using models to replace labor. Generally speaking, time series classification using traditional machine learning methods (such as KNN, DTW) can achieve better results. However, as for big data, deep learning tends to defeat traditional methods. Until recently, a paper relevant to the research was published by Fawaz H I et al. [4], which has demonstrated the feasibility of transfer learning method for different time series data. The author argues that FCN can learn time series representation well when the amount of data is sufficient, and believes that the features extracted by the deep network for the time series data are as similar and inherited as CNN in terms of time series. Moreover, one of the challenges for supervised learning is the large number of labeled data. Unfortunately, it is not readily available for the real-world labeled data problem tended to the high cost and longtime consuming. This problem in essence involves using transfer learning to obtain a solution. It can be seen that the solution based on the transfer learning becomes a better choice for the self-adaption anomaly detection problem.

A new ATSDLN model is proposed in the paper, which realized an adaptional classification of time series anomaly detectors and run-time parameters selection by combining transfer learning and dynamic adaptive joint learning. It is a pre-trained model based on public data sets for transfer learning. Figure 1 is our frame diagram. The model supports multiple channels, and can input the original time series, prediction time series or residual sequence.

From the bottom to the top, the first part is the Fully Convolutional Neural Network (FCN), which is made up of Convolution layers and Global average pooling layer. As the Figure 1 shows, transfer learning is applied to the FCN layer and fine-tuning in the FC layers, which makes the network parameters initialized better, so as to speed up the training and convergence and improve the performance of time series classification model. The main function of this part is to learn the rich time series representation by means of a large amount of training data, and then produce time series representation. This part introduces the ability of the migration learning enhancement model to extract the ability of time series representation, to deal with the problem of marking sample sparseness and model mobility.

The second part is composed of two sub-networks, both of which are supervised classification models. The left part is responsible for the classification of the detector, while the right part is responsible for the classification of the corresponding run-time parameters of the detector, the two parts can jointly study. The expression learned through the detector classification task will be used as the input of the run-time parameters selection task, which can assist the learning of the run-time parameters. Both of the sub-networks have the problems of supervised classification. From the figure 1, it can be seen that the output of p(x) determines a certain detector uniquely for the current time series, and [q(x)] is the run-time parameters that the current time series and anomaly detector choose. Because the size of the candidate run-time parameters sets of each detector is inconsistent, the last layer width of the right network follows the left as the side detector changes, that the model supports flexible addition and deletion detectors. It can be known from the above that the selection of the run-time parameters on the right side

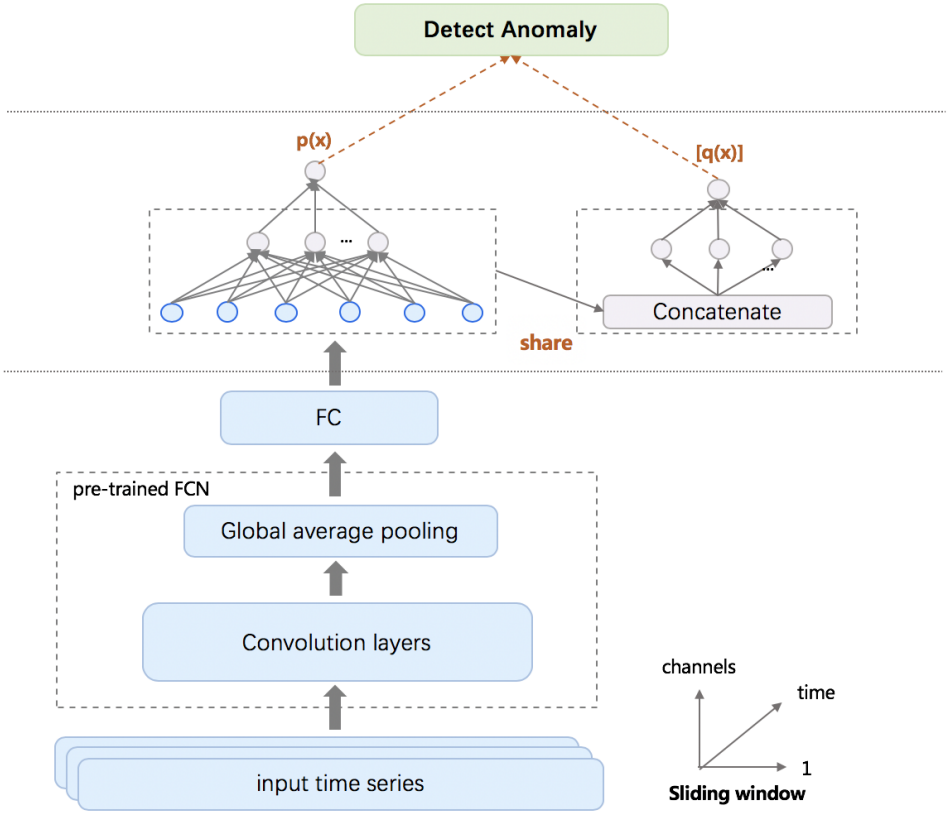

Figure 1: Whole Net Structure, left represents anomaly detectors classification task, right represents run-time parameters fine-tuning task. Blue layers are shared by the two sub-networks.

depends not only on the time series representation, but also on the detector selected by the network on the left side. So in this part, the expression learned in the left detector classification task is shared to the task of the right run-time parameters selection on the right and is taken as its input to assist in learning.

The third part, which is on the top, is the execution module for the anomaly detection. It detects the anomaly of the detector and the run-time parameters which is selected when time series use models.

3 EXPERIMENTAL SETTING

The following parts form the core components of an joint learning approach. The two sub-networks in our approach refers to anomaly detectors classification task and run-time parameters fine-tuning task, which means the network predicts optimal detector and fine-tunes the run-time parameters simultaneously without human interfering. Firstly, we collect some classical detectors, which were proposed to detect anomaly in different context. Secondly, a new evaluation criterion was proposed to evaluate the performance of these detectors in each time series data, this process also generates the label of our two sub-tasks. Thirdly, an adaptive model is trained to extract deep features of time series, which is crucial for optimal detector prediction and the run-time parameters fine-tuning tasks. Lastly, we transfer this representation learned from public data sets to other unseen data sets and evaluate the usability of transfer learning in time series anomaly detection.

3.1 Datasets

We set the different sizes of sliding window on webscope S5 data sets[1] for the experimental sample, which contains outliers and change points, and use the UCR Time series Classification Archive[2] as the source data sets for transfer learning.

Webscope S5 is a labeled anomaly detection data set. There are 367 time series in the data sets, each of which contains between 741 and 1680 data points at regular intervals. Each time series is accompanied by an indicator series with 1 if the observation was an anomaly, and 0 otherwise. **UCR** is a time series classification data sets. There are 128 data sets with different applications. The classification type of these data sets is from 2 to 60, and the the size of data sets is from 20 to 8926.

Through traversing the candidate detector and the combining operational parameters, the optimal detector and run-time parameters are selected for the time series as a training data for supervised learning. Then, by carrying out the pre-training of the transfer

[1]https://research.yahoo.com/
[2]https://www.cs.ucr.edu/

learning on the UCR time series classification data set, the data volume problem of the training data charged by the meter is solved.

3.2 Evaluation criterion

Experiments results were evaluated by comparing observed anomalies to true anomalies. In table 1, we present the evaluation measures of the model's such as precision and recall, Error which were used. FP denotes the number of false positive, FN the number of false negative, TP the number of true positive and TN the number of true negative.

Number of true positive whose proportion in anomaly detection is small, in addition without considering precision's inability to accurately express the level of false positive ratio (or false alarm ratio), especially when true positive is zero, precision is always zero. there are not very good measures for assessing anomaly detection methods. In our situation, the high false positive ratio will cause alarm fatigue of the relevant personnel, which will lead to the decrease of the attention of monitoring alarm. However, the number of true negative is large, so the false positive rate is not sensitively enough as it grows very slowly. Therefore, we propose a new metric named Error which is defined as FP/(TP+FP+FN).

3.3 Detectors for time series anomaly detection

According to the shape and context of time series anomaly, it can be summarized as outlier, mean-shift, cliff-type, deviating-trend, new-shape. See the table 2 for details. The anomaly detectors used in ATSDLN are just the same as EGADS.

In addition, the parameters fine-tuning is as important as the accuracy of selecting the most suitable detector. Detector parameters are divided into two categories: the first is the common parameters needed by all detectors, including sliding window size, sensitivity, number of historical samples. The second is the internal parameters required by each detector algorithm, such as K-Multiple variance of KSigma, eps and minPts of DBScan, confidence and drift range of ChangePoint, search radius of DTW similarity, etc.

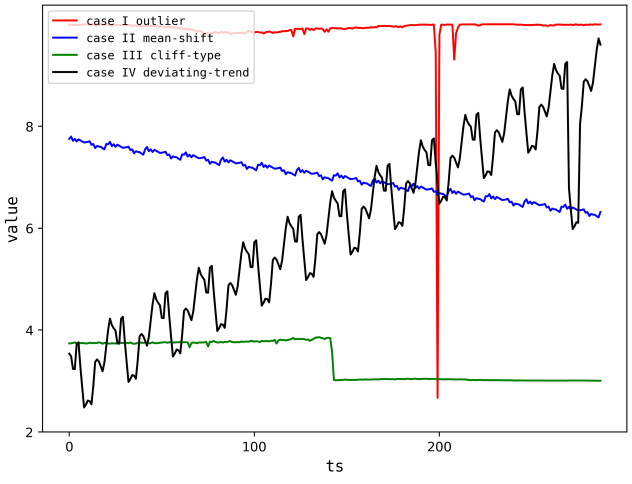

Figure 2: Example of anomaly types.

4 RESULTS AND DISCUSSIONS

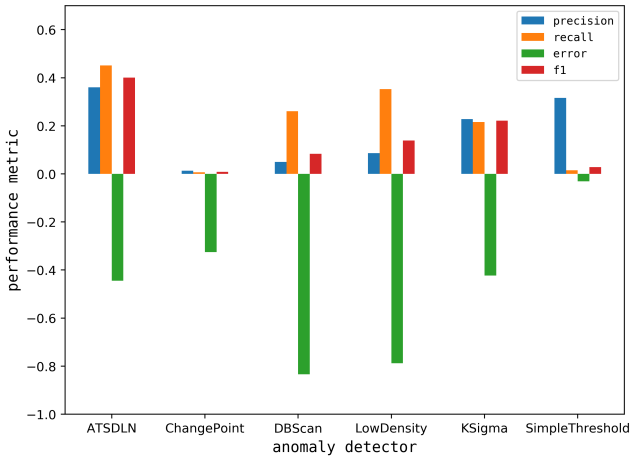

Figure 3: Anomaly model performance on different detectors(adaptively select best run-time parameters).

The second chapter mentions that the network is composed of two sub-networks, both of which are supervised classification models. The output of p(x) determines a certain detector uniquely, and q(x) is the run-time parameters corresponding to the detector. With the determined detector and run-time parameters, it is possible to judge the abnormality of the time series execution abnormality detection. The evaluation of part of the experimental effect adopts the precision, recall and error described in evaluation criterion in Chapter 3.2.

The main work of this paper is to select the appropriate detector as well as run-time parameters for a certain time series. The length of the time series is called the window size of the time series. The size of the window not only has relationship to the business attributes, but also influences the sensitivity of the detector's self-adaption selection. In theory, the smaller the window, the more sensitive the changes in the detector and parameters. The traditional voting method relies more on the accumulation of time series data and has poor adaptability. As is shown in Fig. 4, the horizontal axis is the window size of the time series, while the vertical axis if the evaluation index calculated by the abnormality detection result. It can be seen that the smaller the window, the worse the baseline effect. According to our experiments, the window size will not affect the performance of our model. The ATSDLN can better adapt to different window size. In order to compare the performance of different experiments, we choose the window size with 200 points.

4.1 Anomaly model performance analysis

In order to explore the necessity for self-adaption selection detectors and operating parameters, 29 combination parameters of five detectors are selected which described in detectors for time series anomaly detection in Chapter 3.3. The experiments are performed on the yahoo public data set. The results are shown in Figure 5, the horizontal axis shows the 29 combinations of detectors and its parameters, the vertical axis shows the performance under each

Table 1: Evaluation Metrics

Metric	Description
Precision	Precision is defined as TP/(TP + FP)
Recall	True positive rate or recall is defined as TP/(TP+FN)
False positive rate	The false positive rate is defined as FP/(FP+TN)
F1-score	F1-score is dened as 2 * precision * recall /(precision + recall)
Error	Error is defined as FP/(TP+FP+FN)

Table 2: Anomaly and Detectors

type	descriptions	detector
outlier	significantly different	KSigma/DBScan/LOF/Extreme LowDensity[1, 8]
mean-shift	sustained inapparent deviation	CUSUM changepoint[7]
cliff-type	switch to another sustained value	KernelDensity changepoint/KSigma/SimpleThreshold
deviating-trend	not in line with fitting trend	STL decomposition[3]
new-shape	un-similar with others	DTW similarity[5]

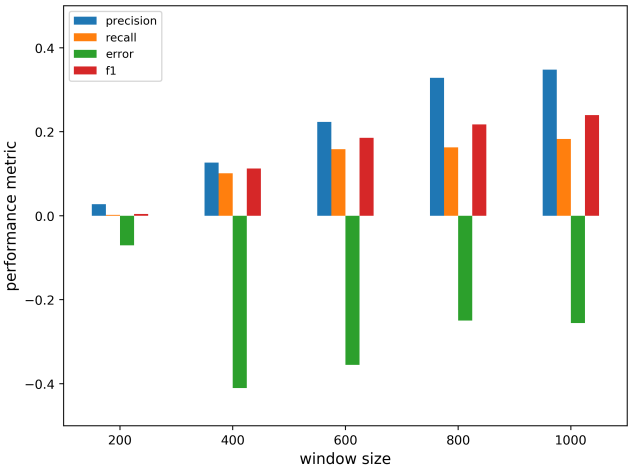

Figure 4: Baseline performance on different window size.

combination. There is no existence of fixed detector and parameters which can be optimal at the whole time series. In addition, the different effects of the run-time parameters are widely divergent under the situation when the detector is determined.

4.2 Compare with single model

Figure 3 compared the results of ATSDLN with other single detector models. It shows that the performance of our method is the best. The method proposed in the paper, regardless of accuracy, recall rate or F1, is superior to the single detector adaptive selection of optimal parameters, and the false positive rate is also reduced.

Table 3: Peformance of different network architectures

model type	Precision	Recall	Error	F1
Baseline	0.0278	0.0022	0.7035	0.0040
LSTM-DNN	0.0940	0.5195	0.8335	0.1592
FCN-LSTM-DNN	0.4419	0.0023	0.0028	0.0045
ATSDLN	0.3606	0.4512	**0.4445**	**0.4010**

4.3 Comparison of different network architectures

In this paper, we compare several network architectures to investigate our proposed model's effectiveness. The controlled models are described as follows:

- `Baseline`: The majority voting algorithm based on EGADS.
- `LSTM-DNN`: A hybrid neural network composed of Long short-term memory LSTM and DNN network.
- `FCN-LSTM-DNN`: The model adds a CNN layer to capture features based on the LSTM-DNN model.

Table 3 shows that, when the multi-detector detection result voting method is adopted, each detect needs to traverse all detectors and candidate run-time parameters combinations, it is very time-consuming, which do not meet the demands of industrial real-time monitoring scenarios. Moreover, compared to the voting algorithm, the neural network models behave higher F1 score, what's more, difference model structures can affect the evaluate metric. The CNN shows the better ability of abstract feature extraction in our task.

4.4 Influence of share layers

As previously reported, the model selects the run-time parameters for the current time series as well as the detector. The time series representation learned through the detector classification task will be used as the input of the run-time parameters selection task, which can assist the learning of the run-time parameters. This part

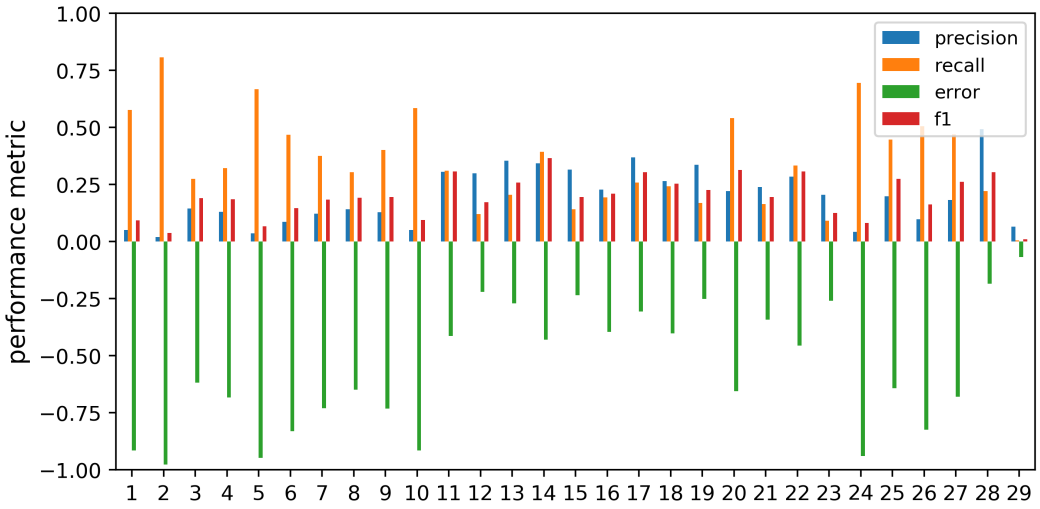

Figure 5: Anomaly model performance on different parameters.

Table 4: Effect of sharing layers

model type	Precision	Recall	Error	F1
NS-Model	0.0982	0.5146	0.8253	0.1649
SSR-Model	0.2366	0.3374	0.5212	0.2782
ATSDLN	0.3606	0.4512	**0.4445**	**0.4010**

Table 5: ATSDLN with Transfer learning

model type	Precision	Recall	Error	F1
ATSDLN	0.3606	0.4512	0.4445	0.4010
Transfer-1	0.5191	0.3623	**0.2513**	**0.4268**
Transfer-2	0.3724	0.4350	0.4230	0.4013
Transfer-3	0.3613	0.4506	0.4434	0.4011

discusses the influence of share layers. The relevant models are described as follows:

- `NS-Model`: without shared of network.
- `SSR-Model`: shared the shallow representation (the output layer of FCN).
- `ATSDLN`: shared specific representation (the layers of the detector classification task).

As is shown in Table 4, the effect of shared the shallow and specific representations of time series is optimal through the classification network (anomaly detector selection) on the left and the classification network (run-time parameters selection) on the right. This is because the run-time parameters have a strong relativity with the detector. In order to output appropriate detector categories, the expression learned by the detector classification task will be used as input of the parameter classification task to assist parameter learning.

4.5 Influence of Transfer learning

To solve the shortage of data and let the network extract temporal features and initialize the models better, we selecting some UCR sample data sets for the comparative experiments of transfer learning.

- `ATSDLN`: Training without transfer learning.
- `Transfer-1`: Transfer from FordA to our data.
- `Transfer-2`: Transfer from Earthquakes to our data.
- `Transfer-3`: Transfer from coffe to our data.

The second chapter mentions that transfer learning is applied to the FCN layer and fine-tuning in the FC layers, which makes the network parameters initialized better, so as to speed up the training and convergence and improve the performance of time series classification model. Table 5 shows that, in most of the cases, the pre-trained model can improve the performance of model.

5 CONCLUSIONS

This paper proposed a new ATSDLN model, which realized an adaptional classification of time series anomaly detectors and run-time parameters selection by combining transfer learning and dynamic adaptive joint learning. The second Chapter mentions that the network is composed of two sub-tasks: anomaly detector classification and the run-time parameters fine-tuning network, both of which are supervised classification models. Because the size of the candidate run-time parameters sets of each detector is inconsistent, the last layer width of the right network (the run-time parameters fine-tuning network) follows the left as the side detector changes, that the model supports flexible addition and deletion detectors. Furthermore, because the run-time parameters have a strong relativity with the anomaly detector, the effect of shared the shallow and specific representations of time series is optimal through anomaly detectors classification network and run-time parameters fine-tuning network. Moreover, we pre-trained FCN layers based on different data sets, the results investigated that transfer learning approach

can improve the performance of our model. Experiment results show that ATSDLN solves the problem of low precision and high false alarm ratio when the data pattern is change. ATSDLN is also applied to our industrial scenarios. In the future, we will consider extract global features of time series and alarm suppression.

REFERENCES

[1] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. 2000. LOF: identifying density-based local outliers. In *ACM sigmod record*, Vol. 29. ACM, 93–104.

[2] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *ACM computing surveys (CSUR)* 41, 3 (2009), 15.

[3] Robert B Cleveland, William S Cleveland, Jean E McRae, and Irma Terpenning. 1990. STL: A seasonal-trend decomposition. *Journal of official statistics* 6, 1 (1990), 3–73.

[4] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. 2018. Transfer learning for time series classification. In *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 1367–1376.

[5] Tak-chung Fu. 2011. A review on time series data mining. *Engineering Applications of Artificial Intelligence* 24, 1 (2011), 164–181.

[6] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. 2018. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 387–395.

[7] Yoshinobu Kawahara, Takehisa Yairi, and Kazuo Machida. 2007. Change-point detection in time-series data based on subspace identification. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*. IEEE, 559–564.

[8] Nikolay Laptev, Saeed Amizadeh, and Ian Flint. 2015. Generic and scalable framework for automated time-series anomaly detection. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1939–1947.

[9] Dapeng Liu, Youjian Zhao, Haowen Xu, Yongqian Sun, Dan Pei, Jiao Luo, Xiaowei Jing, and Mei Feng. 2015. Opprentice: towards practical and automatic anomaly detection through machine learning. In *Proceedings of the 2015 Internet Measurement Conference*. ACM, 211–224.

[10] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. 2016. LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint arXiv:1607.00148* (2016).

[11] Dominique T Shipmon, Jason M Gurevitch, Paolo M Piselli, and Stephen T Edwards. 2017. Time series anomaly detection; detection of anomalous drops with limited features and sparse examples in noisy highly periodic data. *arXiv preprint arXiv:1708.03665* (2017).

[12] Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, et al. 2018. Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 187–196.