Spacecraft Anomaly Detection and Relation Visualization via Masked Time Series Modeling

Hengyu Meng
School of Aeronautics and Astronautics
Shanghai Jiao Tong University
Shanghai, China
LULVHP@sjtu.edu.cn

Yuanxiang Li
School of Aeronautics and Astronatics
Shanghai Jiao Tong University
Shanghai, China
yuanxli@sjtu.edu.cn

Yuxuan Zhang
School of Aeronautics and Astronautics
Shanghai Jiao Tong University
Shanghai, China
yuxuanzhang@sjtu.edu.cn

Honghua Zhao
Maintenance Control Centre
Eastern Airlines Technic Co, Ltd
Shanghai, China
hhzhao@ceair.com

***Abstract*—Anomaly detection (AD) refers to find patterns in time series data that do not behave expectedly. Current state-of-the-art anomaly detection method, based on reconstruction error generated by LSTM sequence modeling. Recently, the remarkable improvement achieved by BERT model in language translation demonstrated that transformer is superior to LSTM models, due to its extracting relations ignoring distance. In this paper, we propose a transformer-based architecture, Masked Time Series Modeling, modeling data stream. We compared the performances of our method with state-of-the-art AD methods on challenging public NASA telemetry dataset. The experiment results demonstrated our method saves about 80% time cost because of parallel computing compared with LSTM methods and achieves 0.78 F1 point-based score. Moreover, we visualize anthropogenic anomalies through attention score matrix.**

Keywords-spacecraft; anomaly detection; transformer; mask; deep learning; attention mechanism

I. INTRODUCTION

Anomaly detection (AD) is the process of identifying non-conforming items, events, or behaviors [1, 2]. Efficient detection of anomalies can be useful in many fields. Examples include quantitative transaction, threat detection for cyber-attacks [3, 4], or safety analysis for self-driving cars [5]. Many real-world anomalies can be detected due to promotion of various industrial sensors. Especially for in-orbit spacecraft, failure to detect hazards could cause serious or even irreparable damage since spacecraft are expensive and complex system. In the absence of remedial measures, anomaly detection is important and necessary to warn operation engineer of anomalies.

With the rise of deep learning, anomaly detection based on deep learning has become the part of mainstream. The existing anomaly detection methods can be grouped into three types, which include density-based methods, clustering-based methods and reconstruction-based methods. The density-based

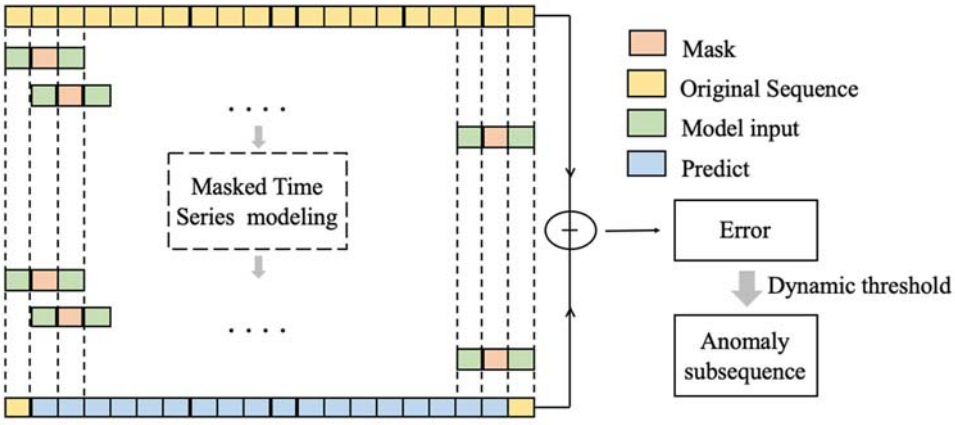

Figure 1. Framework of masked time series modeling. We intercept subsequences from the original sequence. The front and back parts of the subsequence are used as input to the model, and the middle part of the subsequence is output of reconstruct model. Detect anom anomalies based on the error between reconstructed sequence and the original.

978-1-7281-0860-5/19/$31.00 ©2019 IEEE

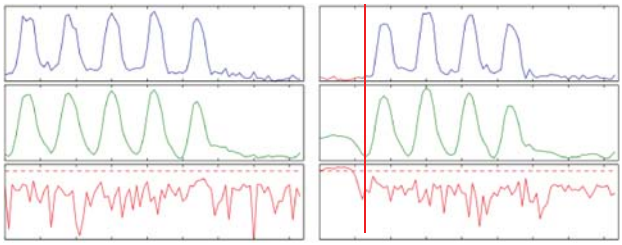

Figure 2. Examples of anomalies via reconstruction. The internal marked by the red line is the detected anomaly.

methods identify a sample with low likelihood as an outlier [6, 7]. Clustering-based methods judge an object as an outlier if it does not belong to any cluster or the cluster size is too small [8].

Reconstruction-based anomaly detection is the most popular one and has been deployed into many industrial fields [11, 15]. The main ideas of reconstruction-based anomaly detection methods are as follows: 1) What the "normal" sequence should look like, which means reconstructing sequence via RNN models trained by normal sequences; 2) Use the same model to reconstruct the sequences with anomalies and compare the reconstructed sequence with the input; 3) Set the error function and threshold. Where an abnormality occurs in the entire sequence, the reconstruction may be unideal, so classification or clustering can segment the anomalies [13, 14].

However, there exist two problems on reconstruction-based anomaly detection: 1) Long short-term memory (LSTM) depends on uni-directional sequential propagation, which makes the computation of LSTM low-effective; 2) uni-direction propagation will delay the anomaly detection time due to the sparse unexpected data in the early observed anomaly subsequences.

Contributions. In this paper, we propose a masked time series modeling method based on transformer, as shown in Fig. 1, which has two novel components: 1) the attention mechanism used for updating timestep in parallel; 2) the mask strategy used to detect the anomaly in advanced time. In this way, the reconstruction in the front of anomalies can be affected by the anomaly data, resulting in early anomaly detection. Once reconstructed time series are generated, we apply a dynamic thresholding approach for evaluating reconstruction error. Experiments show that due to the characteristic of transformer encoder, the modeling greatly reduces time consumption without significant drop in accuracy compared with LSTM reconstruction. And using bi-directional data makes model capture anomalies better for range-based precision and recall indicator. This work is tested on NASA spacecraft datasets, but can be applied to other anomaly detections tasks.

The article is organized as follows: Section 2 introduces the related work, the anomaly detection based on LSTM reconstruction and the transformer encoder generally used for NLP tasks. Section 3 presents our method, showing the inputs using contextual information, and gives the model reconstruction process. In Section 4, experiments are carried out on the NASA spacecraft dataset. We compared the relationship between model consumption time and detection accuracy, and

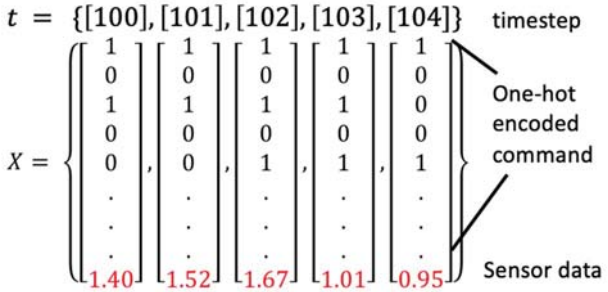

Figure 3. Each vector contains 2 parts: one-hot encoded command and continuous sensor data

compared it with LSTM methods on point-based and range-based indicators. Section 5 summarizes the full paper.

II. Related Work

Quite part of anomalies is caused by improper operations, especially in complex and stressful work of aerospace. Therefore, when considering anomaly detection, both the sensor signal and the operator's operation command should be considered. Since the one-hot coding representation (shown in Fig. 3) of the operation command in the time dimension is discrete, which is similar to the discrete word vector in natural language processing (NLP), methods proven effective in NLP such as transformer can be considered.

A. Anomaly detection based on LSTM reconstruction

Fig. 2 shows the anomaly detection based on LSTM reconstruction [9]. The researchers trained the reconstructed model using normal sequences. When a model reconstructs a sequence with anomalies (blue lines in the first column), there is a difference between the input and output (green lines in the second column) sequences. Anomalous subsequences can be segmented applying a simple threshold segmentation or probability distribution model on reconstructed error (red line in the third column).

For some anomaly detection studies, the limited size of the data set is an inevitable topic. For other reconstruction tasks, such as GAN based image generation, at least thousands of images are often required. However, some anomaly detection data sets often have only tens of hundreds of data, some specific anomalies even only have a little labelled data. And as the sampling density increases, the length of the time series data also increases, making it increasingly difficult to generate the entire time series.

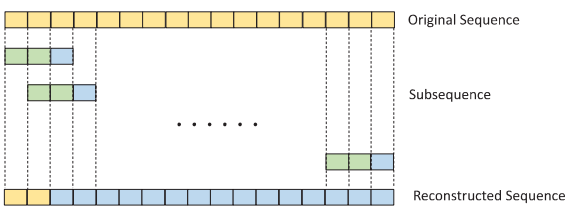

Figure 4. A fixed-length (for example 1000, the yellow squares in the figure) sequence can intercept up to 800 sub-sequences, each of which has a length of 200. Then the model can use the previous 180 time steps (the green squares) as input data and the tail 20 time steps (the blue squares) as output data.

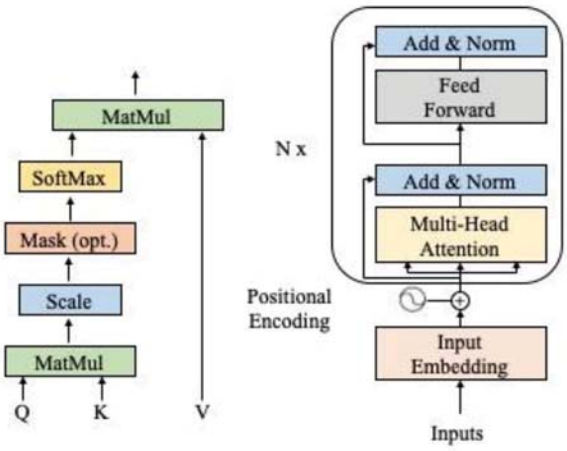

Figure 5. Dot-Product Attention(left) and Transformer Encoder(right)

Some anomaly detection researchers use local subsequences reconstruction to train models. In details, researchers intercept fixed-length subsequences from the time series data, using the front part as the input of the time series model and the tail part as the prediction object. This process is shown in Fig. 4.

Currently LSTM reconstruction error system instead of costly expert system has successfully identified several confirmed anomalies since deployed to the Soil Moisture Active Passive satellite (SMAP) and the Mars Science Laboratory rover (MSL), Curiosity.

B. *Transformer encoder*

Self-attention is a special attention mechanism. In self-attention, query is equal to key which is equal to value too.

Transformer, a self-attention mechanism that learns contextual relations between words (or sub-words) in a text, was proposed [10]. As shown in Fig.5, transformer includes two separate mechanisms—an encoder that reads the text input and a decoder that produces a prediction for the task. Since transformer-based BERT model has dominated effect in language modeling task, which shows transformer's strong ability to extract features.

As opposed to directional models, which read the text input sequentially (left-to-right or right-to-left) [11], the Transformer encoder reads the entire sequence of words at once. Therefore, it can be considered bidirectional, though it would be more accurate to say that it is non-directional. This characteristic allows the model to learn the meaning of a word based on its surroundings (left and right of the word).

For deep transformer based network, two pre-training tasks are proposed, one of which is masked language modeling. The main process of masked Bi-directional language model: randomly select 15% of the words in the corpus, and mask it, which means replacing the original word with the [Mask] mask, and then train the model to correctly predict the word that was discarded.

III. METHOD: MASKED TIME SERIES MODELING

In order to solve the shortcomings of the LSTM network, in which only the past information can be utilized, and utilize the advantage of parallel computing. Transformer encoder is applied to the spacecraft time series data modeling. Inspired by the masked language model, we propose a method using masked state prediction to reconstruct the time series. The symbols mentioned in the following is listed in Table 1.

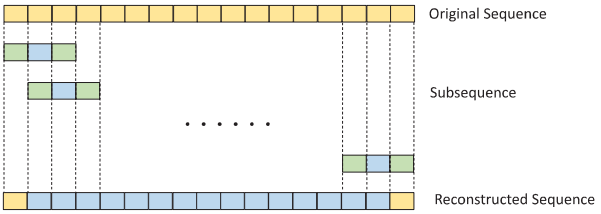

Figure. 6. Data interception process, output (the blue squares) is predicted by history and future data (the green squares in front of and behind the blue ones).

A. *Model input and output*

Since LSTM can only propagate in one direction, the reconstruction process is actually more similar to prediction: predicting current data from historical data, and not using future data. In fact, in many other time series tasks, the comprehensive use of information in both directions will greatly enhance the capabilities of the model.

The input of masked time series model is intercepted from sequences as shown in Fig. 6. The subsequence of this model input includes past and future information which is different from the input of the LSTM model. In other word, in LSTM-based reconstruction, L_{i0} is the length of whole input, and L_{i1} is 0. In our method, L_{i0} and L_{i1} are greater than 0(usually equal to each other). So there exits problems about online detection which will be discussed later.

B. *Mask operation*

Mask operation is an operation that blocks specific timesteps from the network. Specifically, in the transformer

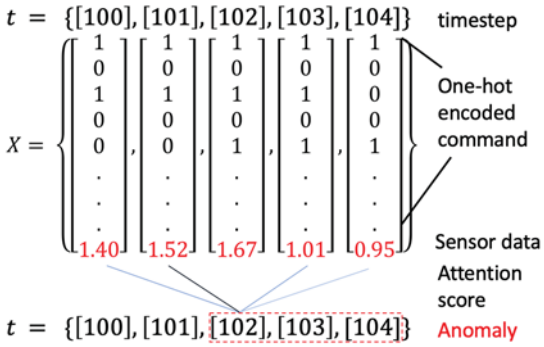

Figure 7. Visualization of attention score matrix. Assuming there exists an anomaly in timestep 102 and after, the transformer encoder can extract relations between 102 and 101,100. If attention score between 103 and 102 is abnormally high, there exists great possibility than manual command sent in 102 caused the anomaly.

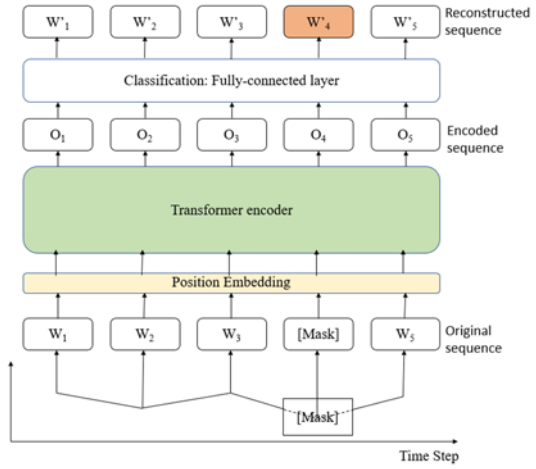

Fig 8. Process of masked time series reconstruction

encoder, the score of masked timesteps in the relation matrix is set to infinity so that it can only accept the reconstruction via the remaining time steps.

This article discusses the location and length for the mask operation. In fact, there are three variables L_{i0}, L_{i1}, L_o that determine the mask, which are the length of time series before the mask, the length of time series after the mask, and the length of the mask itself. When the post length is 0, the model degenerates to be the same as LSTM, and only uses uni-directional information for prediction. Different mask lengths also have an impact on accuracy as shown in Section IV.

C. Reconstruction model structure

The model structure of masked time series modeling is shown in Fig.7. The input to the model is the normalized raw signal w_i. After masking the middle part which will be reconstructed, the position encoding layer is added to avoid self-attention mechanism ignoring the position information. Next is the transformer encoder. After obtaining the features O_i extracted by the transformer encoder, the fully connected layer is used to obtain the final prediction of the masked state w_i'.

The Attention score is shown in formula following:

$$Attention(\boldsymbol{w_i}, \boldsymbol{w_j}) = softmax\left(\frac{\boldsymbol{w_i} \times \boldsymbol{w_j^T}}{\sqrt{\boldsymbol{d_{w_i}}}}\right)\boldsymbol{w_i} \cdots 1 \le i, j \le n \quad (1)$$

Transformer encoder Directly calculate the correlation of different time steps, no matter how far apart the two time-steps (of course in the actual experiments, maximum distance threshold is set to simplify the problem). As shown in Fig. 7 We assume that in timestep 103 and after the anomaly occurs, and our attention matrix captures that the correlation coefficient of 103 and 102 is abnormally high. We can assume that the command sent at time 102 has a considerable relationship with the abnormality at 103.

The model itself is similar to the masked language modeling. The biggest difference is that in the language modeling task, the input is the word vector after embedding, and the output is the probability distribution of the word, which is a discontinuous variable. Here in this task, the input and output contains continuous vectors. So we choose regression instead of classification in the top layers. Moreover, word vectors usually have thousands of dimensions, so multi-head mechanism is employed to map word vectors to different subspaces for parallel computation. But for the data set in our experiment, the signal has only 25 or 55 dimensions, so using too many heads makes no difference.

TABLE I. NOTATION

Notation	Description
L_{i0}, L_{i1}	Length of input subsequences in front of and behind model output
L_o	Length of reconstructed subsequences/model output
R, R_i, N_r	Set of annotated subsequences, i^{th} annotated subsequence, Total number of points in R
P, P_j, N_p	Set of predicted subsequences, i^{th} predicted subsequence, Total number of points in P
α， z	Weight of existence reward, Weight of standard deviation
$\gamma(.)$, $\omega(.)$, $\delta(.)$	overlap cardinality function, overlap size function, positional bias function
w_i, O_i, w_i'	Original timestamp, Output of transformer encoder, Reconstructed timestamp
$e, e_i, \varepsilon, \varepsilon_i$	Error sequence and its i^{th} time-step; Threshold sequence and its i^{th} time-steps

D. Anomaly detection via non-dynamic threshold [11]

After obtaining the reconstructed sequence w_i', we can calculate the reconstruction error e_i compared with the original sequence. Then we segment anomalies via dynamic threshold $\boldsymbol{\varepsilon}$ which is defined as:

$$\varepsilon = \mu(e_i) + z\sigma(e_i) \quad (2)$$

$$\varepsilon_i = argmax(\varepsilon) = \frac{\Delta\mu(e_i)/\mu(e_i) + \Delta\sigma(e_i)/\sigma(e_i)}{|e_a| + |P_j|^2} \quad (3)$$

For μ, σ, e_a above (μ is expectation and σ is standard deviation):

$$\Delta\mu(e_i) = \mu(e_i) - \mu(\{e_i \in e_i | e_i < \varepsilon\}) \quad (4)$$

$$\Delta\sigma(e_i) = \sigma(e_i) - \sigma(\{e_i \in e_i | e_i < \varepsilon\}) \quad (5)$$

$$e_a = \{e_i \in e_i | e_i < \varepsilon\} \quad P_j = sequence\ of\ e_a \quad (6)$$

TABLE II. DETAILS OF SPACECRAFT TELEMETRY DATASET

	SMAP	MSL	Total
Total anomaly sequences	69	36	105
Unique telemetry channels	55	27	82
Unique ISAs	28	19	47
Telemetry values	429735	66709	496444

TABLE III. EXPERIMENTS RESULTS COMPARISION

Dataset	Indicator	Uni-LSTM	Bi-LSTM	Uni-transformer	Bi-transformer
Spacecraft telemetry dataset	Precision	87.5%	88.4%	72.0%	85.4%
	Recall	80%	78.1%	69.0%	72.4%
	Time consumption(min)	757	1100	118	150
Multi-sensor engine	Precision	94%	95.3%	90.1%	93.4%
	Recall	12% [9]	14.1%	11.8%	12.4%
	Time consumption(min)	434	585	125	143

IV. EXPERIMENTS

This section demonstrates the process and result of experiments. First we introduce datasets we experimented on in the following: multi-sensor engine dataset and spacecraft telemetry dataset--open source datasets provided by NASA. Then we give comparison of point-based F1 score and time consumption with LSTM methods. Then, we compare in range-based indicator and visualize attention score matrix for anthropogenic anomalies analysis.

A. *datasets*

The experiments are conducted on 2 datasets:

Spacecraft telemetry dataset: real-world, expert labeled data derived from Incident Surprise, Anomaly (ISA) reports for the Mars Science Laboratory (MSL) rover, Curiosity, and the Soil Moisture Active Passive (SMAP) satellite [11]. The dataset contains 82 normalized telemetries channels of data,

which means we have to train 82 models—one model for one channel, and includes 105 anomalies. The details are shown in TABLE II.

Multi-sensor engine [16]: This dataset has readings from 12 different sensors: One of the sensors is the 'command' sent to the engine, and the rest measure dependent variables like temperature, torque, etc.

The range-based indicator which rely on semantic explanation and the visualization of attention score matrix is given based on experiments conducted on Spacecraft telemetry dataset since it records manual command more completely.

B. *Point-based precision and recal*

After obtaining the reconstruction sequence of the model output, we use the dynamic threshold method to segment the subsequence of the anomalies. The anomaly subsequence is compared with the annotated data, and the point-based precision and recall result is defined as follows:

- True positive is defined as for any R_i in R, there exists P_j that the intersection of R_i and P_j is not none. In other words, R_i overlaps with P_j . If there exist several P_j overlapping with R_i, True positive set only count once.
- False negative means when there is no P_j having an intersection with R_i, this R_i is ignored by our model so FN(false negative)+1.
- False positive means if a P_j don't overlap with any R_i, we count it as FP.

TABLE IV. HYPER-PARAMETER SETTING IN FIG. 9

Serial number	0	1	2	3	4	LSTM
Stack of transformer	6	6	8	12	12	
Length of front input	100	200	100	100	100	200
Length of post input	100	0	100	100	100	0
Length of output	5	5	5	5	1	1

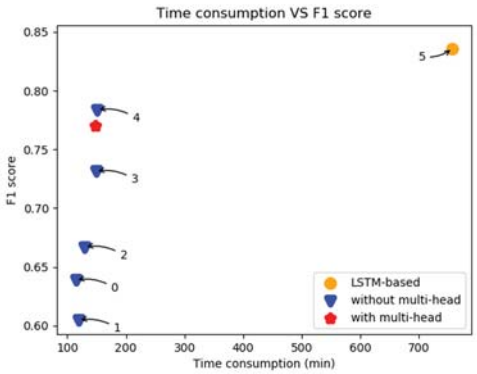

Figure. 9. Time consumption VS accurcy with different hyper-parameters. The comparsion is conducted in spacecraft telemetry datasets. The red one shows transformer with multi-head don't work better.

The experiments results are shown in TABLE III. Our method consume less time during training. At the same time, for the point-based precision and recall indicators, our method reaches the almost same precision and recall as the state of the art.

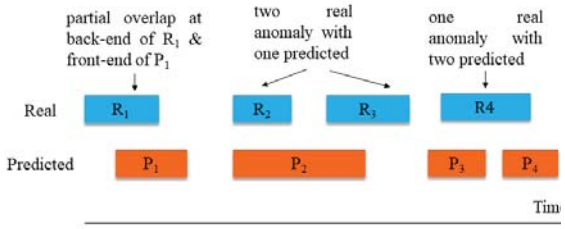

Figure 10. The three cases are all counted as true positive, however the first case detects anomalies delayed from the true value, the second case two anomalies are identified as one, and the third case counts one as two. Physical meanings are

We also explore impact of different hyper-parameters for accuracy and time performance and whether multi-head mechanism would affect results. And multi-head mechanism makes no sense in our task since the dimensions of time series is not complicated enough. Specific hyperparameter settings is shown in TABLE IV.

C. Relation extraction and range-based indicators

Nesime Tatbul [12]present a new mathematical model called range-based Precision and Recall metrics to evaluate the accuracy of time series classification algorithms. They expand the well-known Precision and Recall metrics to measure ranges. Fig.10 visualizing motivation of range-based.

$$Recall(R_i, P) = \alpha \times ExistenceReward(R_i, P) + (1-\alpha) \times OverlapReward(R_i, P) \quad (7)$$

The definition of existence reward is the approximately same as defined in point-based while overlap reward makes the differences. Overlap reward depends on three functions erange size and position bias of overlap

$$OverlapReward(R_i, P) = [\sum_{j=1}^{N_p} \omega(R_i, R_i \cap P_j, P_j)] \times CardinalityFactor(R_i, P) \quad (8)$$

$$CardinalityFactor(R_i, P) = \begin{cases} 1 & if\ R_i\ overlap\ with\ at\ most\ one\ P_j \\ \gamma(R_i, P) & otherwise \end{cases} \quad (9)$$

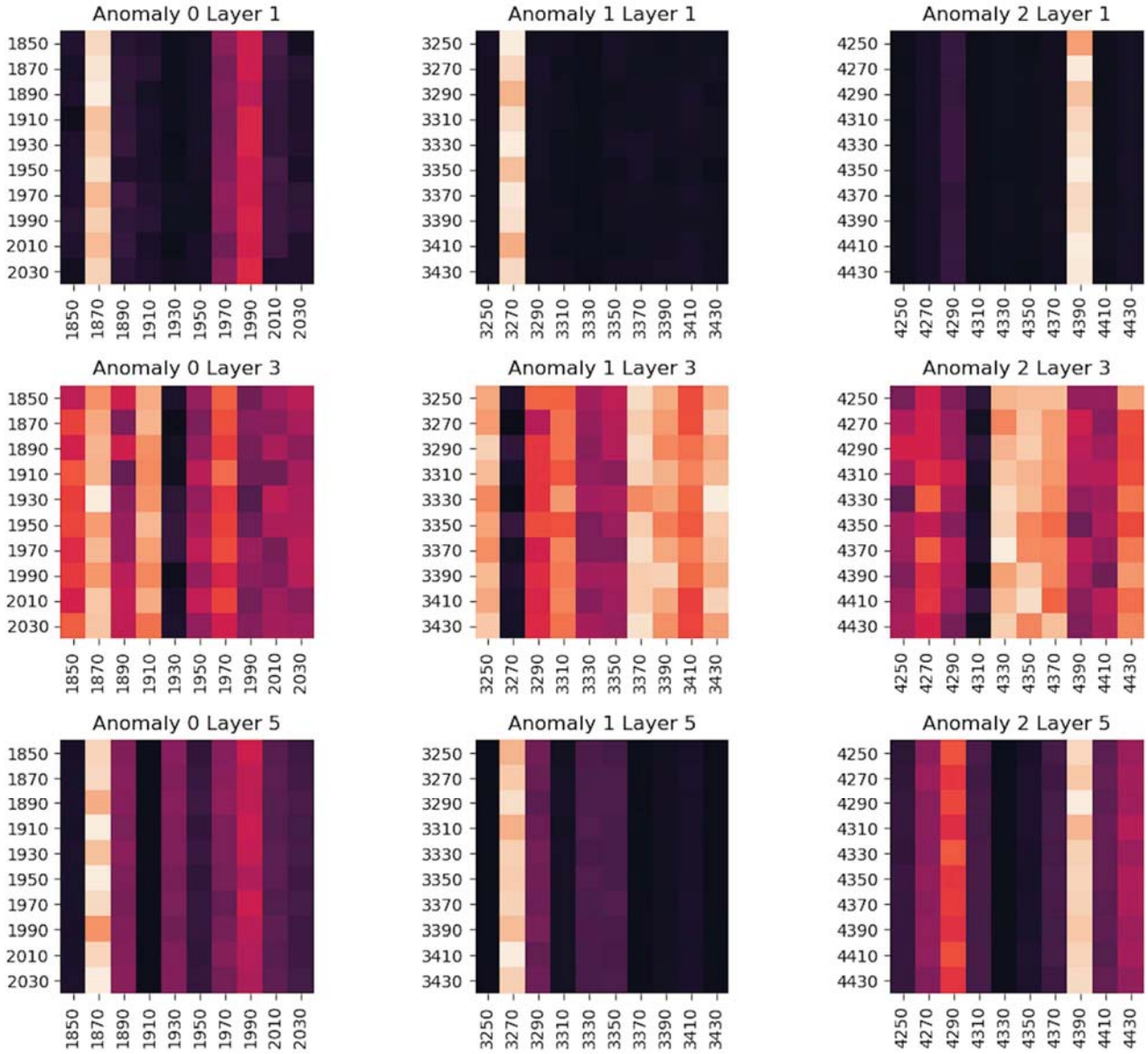

Figure. 11. Visualization of attention score matrix

TABLE V RESULTS EVALUATED ON RANGE-BASED INDICATOR

method	Our method	LSTM-based
precision	0.64	0.62
recall	0.50	0.49

In this experiment, α is set to 0.5 because we want to pay equal attention to ExistenceReward and OverlapReward. $\gamma(.)$ is set to the reciprocal of the number of overlaps. $\delta(.)$ is set to front-end bias since we want to detect anomalies as early as possible. And range-based precision only consists OverLapReward. The parameters are set to the same in range-based recall.

The experiments results conducted in **Spacecraft telemetry dataset** of LSTM-based and our method is listed in TABLE V. Even for point-range recall, our method is slightly worse than the baseline. But on range-based recall, our approach is slightly better. The probaly reason is shown in following.

D. Visualization of attention score matrix

The attention score matrix is shown in Fig. 11. We chose three subsequences with an anomaly as the observation object. The ranges of the anomalies are 1899-2099, 3289-3529 and 4286-4594 respectively. The starting and ending points of the anomaly can be seen in the low-level attention score matrix of the model. In the high-level attention score matrix of the model, there are several points with the highest attention score for each abnormal starting point which may indicates the relations between manual command and anomalies.

Summary: according to the experimental results, our method has greatly reduced the consumption time while the accuracy has not dropped significantly. At the same time, our approach has slightly improved the range-based indicator due to the use of contextual information. And the visualization of attention score matrix may reveal the relationship between human instructions and anomalies

V. CONCLUSION

LSTM-based anomaly detection method has been deployed into production now. In this paper, we focus on existing two problems on LSTM reconstruction: 1) low compute efficiency duo to uni-directional sequential propagation; 2) Delay of the anomaly detection due to the sparse distribution of anomalies.

This paper proposes a reconstruction method for time series data anomaly detection - masked time series modeling, which applies transformer encoder into time series signal modeling and prove its advantages. The application of mask help to detect the anomaly earlier. In this way, the reconstruction in the front of anomalies can be affected by the anomaly data, resulting in anomaly early detection. At the same time, the advantages of using both front and back information make it better highlight the abnormal point. This method is based on the transformer model and The experimental results verify that transformer is significantly faster than RNN.

However there are still two problems. One is the suitable architecture of transformer encoder used here. Due to the limit of computing resources and datasets, we manually design the structure of network instead of auto search. The other is about online detection. Since bi-directional information is necessary for detection, there will exist latency (the length of post time series) for online anomaly detection.

In the future, we will continue to explore relations between manual command and anomalies like visualization of attention score matrix. On this basis, the end-to-end algorithm will be developed to better detect outliers on range-based indicators.

ACKNOWLEDGMENT

Thank NASA for making datasets open-source.

REFERENCES

[1] C. C. Aggarwal, “Outlier analysis,” Berlin, GE: Springer, 2013.

[2] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: a survey,” ACM Computing Surveys, vol. 41(3), pp.15:1–15:58, 2009.

[3] A. AlEroud and G. Karabatis. “A contextual anomaly detection approach to discover zero-day attacks,” In International Conference on CyberSecurity (ICCS), pp 40–45, 2012.

[4] V. Chandola, V. Mithal, and V. Kumar, “Comparative evaluation of anomaly detection techniques for sequence data,” In IEEE International Conference on Data Mining (ICDM), pp. 743–748, 2008.

[5] O. Anava, E. Hazan, and A. Zeevi, “Online Time Series Prediction with Missing Data,” In International Conference on Machine Learning (ICML), pp. 2191–2199, 2015.

[6] F. Sylvain, P. Gilles, T. Jean-Yves, and Lotfi Chaari. “Improving spacecraft health monitoring with automatic anomaly detection techniques,” In 14th International Conference on Space Operations (SpaceOps 2016), p. 1, 2016.

[7] C. Varun, B. Arindam, and K. Vipin. “Anomaly detection: a survey,” ACM Comput. Surv. 41, 3, Article 15 (jul 2009), pp. 58, 2009.

[8] L.Qi, K.Rudy, C.Chao, G.Glenn, G.David, L.Qin, et al., “Unsupervised detection of contextual anomaly in remotely sensed data,” Remote Sensing of Environment, 2017, 202, pp. 75-87.

[9] P Malhotra, L Vig, G Shroff, and P Agarwal, “Long short term memory networks for anomaly detection in time series,” Proceedings. Presses universitaires de Louvain, 2015.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, and A. Gomez, “Attention is all you need,” In Advances in Neural Information Processing Systems, 2017, pp. 5998-6008.

[11] H. Kyle, V. Constantinou, C. Laporte, I. Colwell, and I. Soderstrom. “Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding,” pp. 387-395.

[12] N. Tatbul, T.J. Lee, S. Zdonik, M. Alam, and J. Gottschlich, “Precision and recall for time series,” neural information processing systems, 2018.

[13] K. Tae-Young and C. Sung-Bae, “Web traffic anomaly detection using C-LSTM neural networks. Expert Systems with Applications,” Volume 106, 2018, pp. 66-76.

[14] M. Zhu, K. Ye, Y. Wang, and CZ. Xu, “A deep learning approach for network anomaly detection based on AMF-LSTM,” Network and Parallel Computing. NPC 2018, Lecture Notes in Computer Science, vol 11276. Springer, Cham.2018.

[15] C. Feng, T. Li, and D. Chana, “Multi-level Anomaly Detection in Industrial Control Systems via Package Signatures and LSTM Networks,” 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Denver, CO, 2017, pp. 261-272.

[16] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, 9(8), pp. 1735–1780, 1997 .