

Received April 29, 2020, accepted June 3, 2020, date of publication June 8, 2020, date of current version June 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3000859

Combining OC-SVMs With LSTM for Detecting Anomalies in Telemetry Data With Irregular Intervals

JUNFENG WU¹, LI YAO¹, BIN LIU¹, ZHEYUAN DING¹, AND LEI ZHANG²

¹ Science and Technology on Information System and Engineering Laboratory, National University of Defense Technology, Changsha 410073, China

² Xi'an Satellite Control Center, Xi'an 710043, China

Corresponding author: Li Yao (liyao6522@sina.com)

This work was supported in part by the Research Project under Grant GK2019A010296.

ABSTRACT To ensure the safety and stability of spacecrafts of which thousands of telemetry parameters are monitored, fast and accurate response to anomalies or potential hazards is very important and challenging. This task becomes more difficult when the obtained telemetry data are sampled at irregular intervals. Long Short-Term Memory networks (LSTM), as time series prediction models, have been applied to satellite anomaly detection and show a promising prospect. However, the anomaly detection method merely based on LSTM does not show a stable performance: when the prediction performance of LSTM is not satisfying, the performance of subsequent anomaly detection will be affected, and the impact is augmented when the telemetry data are of irregular intervals. In order to solve these problems, time intervals are introduced into the LSTM model directly. Besides that, a novel anomaly detection method, Detecting Anomalies using LSTM and Ensembled One-Class Support Vector Machines (DALEO) is proposed to further improve the performance of anomaly detection. In DALEO, multiple One-Class Support Vector Machines are used to obtain the ensemble outputs of high precision and high recall respectively. These ensemble outputs are integrated into the two stages of the anomaly detection method with LSTM in a novel way. Extensive empirical studies on real-world datasets of satellites and space shuttles demonstrate that DALEO improves the performance of anomaly detection significantly when dealing with telemetry data with irregular intervals.

INDEX TERMS Spacecrafts, irregular intervals, one-class support vector machine, long short-term memory network, anomaly detection, integration.

I. INTRODUCTION

Due to the extremely high cost of spacecrafts such as satellites and space shuttles, thousands of kinds of telemetry data are usually used to monitor their status in real time to guarantee their safety and stability during the mission. These telemetry parameters cover data from various subsystems, such as power subsystem, attitude control subsystem. It is difficult to manually design a generic method for all kinds of telemetries. Besides, the sampling frequency of telemetry data is very high, which means that a large volume of telemetry data will be obtained just in a short period of time. What makes the anomaly detection more difficult is that, limited by data transmission or other conditions, the telemetry data of equal intervals are not always available.

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

In the domain of anomaly detection, anomalies are usually divided into three categories: point, collective and contextual [1]. Point anomalies are single values that fall within low-density regions of values, while contextual anomalies are single values that do not fall within low-density regions but are anomalous with regard to their contextual values. Collective anomalies are anomalous sequences rather than single values. A myriad of methods have been proposed for anomaly detection in aerospace field, including expert systems [2], [3], nearest neighbors approaches [4], [5], clustering-based approaches [6], [7], approaches based on dimension reduction [8], [9] and etc. These methods have exhibited certain advantages in certain scenarios, but each of them has its own obvious shortcomings, such as high computational expense, poor generalizability or interpretability, and complex parameter specification. In contrast, out of limits (OOL) approaches have the advantages of low computational expense, broad

applicability, and ease of understanding [10]. However, it is difficult to find the contextual anomalies which are related to temporal information using OOL. In recent years, with the rapid development of deep learning, Recurrent Neural Networks (RNN), especially the Long Short-Term Memory networks (LSTM), have been widely used for anomaly detection of time series [10]–[13]. Compared with traditional RNN, LSTM improves the long-term memory by introducing a weighted self-loop that allows it to forget past information in addition to accumulating it [14].

Anomaly detection of time series can be realized with a LSTM based classification model with the help of sliding windows [12], [15], but it needs a lot of labeled anomaly data to train the supervised classification model, and it is difficult to obtain high-quality labeled anomaly data in the field of aerospace. Another more practical way is to use LSTM as a prediction model [10], [16]–[18] to get prediction sequence and the corresponding error sequence, based on which the anomalies are detected. These error-based methods show obvious advantages in domains where labels of anomalous samples are rare. A prediction model trained on ‘normal’ data is modeled to predict the future values using the current sequences. Therefore, the prediction model can capture and model normal behaviors of a system. When test data represent an unusual pattern, the errors between the actual values and predicted values are far larger than those of normal data, which indicate a potential anomaly. Recently, Hundman *et al.* [10] demonstrated the effectiveness of using LSTM to detect anomalies from satellite telemetry data.

Most existing anomaly detection methods for time series do not pay enough attention to the issue of irregular intervals, because various methods are available for solving the problem of irregular intervals or missing values in time series [19]. Among them, the most crude method is to directly ignore the impact of unequal intervals, and just perform analysis on the observed data, but its performance is poor when the missing rate is high [20]. The intuitive method is data imputation, that is, estimated values are added to fill the irregular data to allow the analysis to proceed in normal way. Imputation methods include linear interpolation [21], polynomial interpolation [22], matrix completion [23], matrix factorization [24], spectral analysis [25], principal component analysis [26] and so on. However, not only data imputation brings additional steps and computation costs, but also the quality of the estimated data can hardly be guaranteed [20]. Since LSTM can directly process multivariate time series and learn the complex non-linear relationship without domain knowledge, time interval, as an important feature, is combined with telemetry value as inputs of LSTM model to gain more accurate predictions.

There are some works which utilize this strategy in RNN-based method for classification [20], [27], but the strategy has never been used for prediction based anomaly detection.

Although such a strategy can reduce the impact of unequal interval for LSTM-based anomaly detection, it still cannot

solve the problem that anomaly detection based on a single model is not robust enough.

In this paper, another unsupervised method, One-Class Support Vector Machine (OC-SVM) [28], is combined with LSTM in a novel way for anomaly detection. The proposed method is named as DALEO to stand for Detecting Anomalies using LSTM and Ensembled OC-SVMs. When OC-SVM is directly used for anomaly detection of temporal data, the performance is poor due to the complex temporal relationship and high dimension of temporal data [29]. In the proposed method, a variety of features extracted from time series are fed into OC-SVM models separately to get their own results, which are aggregated to get two ensemble outputs according to different voting thresholds. One is the ensemble output with high precision, the other is the ensemble output with high recall. In the meantime, the LSTM model which takes the telemetry value and the corresponding time intervals as input is used to get the prediction values. Based on the predicted values, the error sequence and the smoothed error sequence are derived. Then the anomaly scores are calculated for each point in the sequence using the error sequence, the smoothed error sequence and the ensemble output of high precision. Subsequently, the dynamic threshold is calculated based on the derived anomaly scores and the set of candidate anomalous sequences can be obtained. Finally, two pruning strategies are applied using the ensemble output of high recall and the anomaly scores of candidate anomalous sequences sequentially to mitigate the false alarms. The retained anomalous sequences are the final result of anomaly detection.

The main contributions of this study are as follows. Firstly, the shortcomings of the existing LSTM-based anomaly detection methods are pointed out and analyzed. Secondly, a method which combines two unsupervised models in a novel way is proposed to detect anomalies in telemetry data with irregular intervals. Thirdly, the proposed integration framework for the anomaly score calculation module and pruning module provides insights to point-wise anomaly detection methods. Finally, experiments on real datasets demonstrate the effectiveness of combining time intervals with telemetry values as input of LSTM model, and verify that the two ensemble outputs of multiple OC-SVM models improve the performance of anomaly detection when they are integrated into the traditional method.

The rest of the paper is organized as follows. Section II presents the most closely related work. Section III gives the architecture of DALEO and details how it works to detect anomalies in time series sampled at irregular intervals. The experimental details and discussions are articulated in Section IV. Conclusions and future directions of our work are given in Section V.

II. RELATED WORK

The related research works are analyzed from three aspects: LSTM based anomaly detection methods, OC-SVM based anomaly detection methods, and integration methods for anomaly detection.

A. LSTM-BASED ANOMALY DETECTION

LSTM-based anomaly detection methods can be divided into two categories: classification-based methods [12], [15] and error-based methods [10], [16]–[18]. Error-based methods using LSTM have great advantages over classification-based methods in detecting anomalies where labeled samples are barely accessible. Current error-based methods detect anomalies by setting thresholds for prediction error sequences or transformed error sequences. There are two mainstream methods to determine the threshold, one is setting a fixed threshold based on the assumption of Gaussian distribution of the errors [16], [17], which is often violated, while the other is the dynamic threshold method proposed in [10]. In the dynamic threshold method, the exponential weighted moving average (EWMA) [30] is used to process the error sequence, obtaining the smoothed error sequence, based on which the dynamic threshold is determined. This method does not need any hypothesis about error distribution nor anomaly samples to determine the threshold. However, to the best of our knowledge, all the existing thresholds in LSTM-based methods are set based on merely one type of information which describe the state of the anomaly: either the error sequence [16], [17], or the transformed error sequence [10], [18]. Therefore, an anomaly score calculation method which considers a variety of anomaly information at the same time is defined in our method, based upon which a more accurate dynamic threshold can be found.

Many false alarms are often obtained when the thresholds are set, while the pruning method proposed in [10] only considers the characteristics of one point in each candidate anomaly sequence. In the proposed method, a new strategy which is based upon the overall state of the candidate anomalous sequence is applied to the pruning process.

At present, there are limited studies on using LSTM or other RNN models to deal with unequal interval time series except the works in [20], [27], which are proposed to deal with the missing data. The method in [27] introduces the time intervals to predict when the next event will happen, which has little to do with anomaly detection. The method in [20] is proposed to model the ‘informative missingness’ for multivariable time series by modifying RNN structures. Although these methods have different purposes and apply different methods to model the time series of unequal intervals, they show the value of concatenating time intervals with the original time series. In this work, experiments are conducted to verify the effectiveness of concatenating time intervals with telemetry data for anomaly detection when data are irregularly sampled.

B. OC-SVM BASED ANOMALY DETECTION

OC-SVM, as one kind of One-Class classifier, allows for the modeling of just a single class of samples and is often used for anomaly detection. Zhang et al uses the method of data transformation to get two-dimensional sequence [32], but this method does not provide additional information. In [31], [32],

the methods first use autoencoder to learn a fixed length features, then use OC-SVM to detect the anomaly according to the learnt features. The improvement on the performance shows the importance of temporal feature extraction for temporal data anomaly detection using OC-SVM. In [33], an ensemble of OC-SVMs are used to detect Fingerprint Spoof. Although this method is not related to the time series, it shows the effectiveness of ensemble of OC-SVM detectors which utilize different kinds of features.

Enlightened by these works, an ensemble of multiple OC-SVM models which take different kinds of features as input is proposed to generate high precision outputs and high recall outputs respectively for anomaly score calculation and pruning in DALEO.

C. INTEGRATION METHODS FOR ANOMALY DETECTION

The main integration methods of various models for anomaly detection in time series are ensemble-based methods, which take the outputs from base detectors and combine them in some way (max, min average) to generate a final score [34]–[36]. These methods are sensitive to inaccurate detectors, especially when the inaccurate score deviates far from those of other detectors. In [37], the method of finding the optimal weights of base detectors is proposed using Bayesian model for combining classifiers. Since this method is based on the assumption that detectors output scores fit a normal distribution, it is also restrictive. All these ensemble-based methods try to seek the weight for each base detector to get the final output, but no one shows a significantly better performance than others.

In DALEO, by introducing the concept of voting threshold, the outputs of multiple base detectors are aggregated point-wise according to different voting thresholds, and ensemble outputs of high precision and high recall are obtained. The high precision ensemble output is used to calculate the anomaly score for each point in the sequence, and the high recall ensemble output is used to mitigate the false alarms.

III. DALEO

In this section, the main framework of DALEO is firstly given. After that, the ensemble of OC-SVM models and the error calculation using LSTM model with unequal intervals telemetry data are illustrated. Finally, the novel integration method using the ensemble outputs of OC-SVMs and LSTM in the anomaly score calculation module and pruning module is detailed in subsection **D** and **E**.

A. OVERVIEW OF DALEO

The overall framework of DALEO is illustrated in Fig. 1. In DALEO, multiple kinds of features obtained from different Feature Extraction Modules (FEM) are fed into OC-SVM models separately to get their classification results, and then these results are aggregated to get the ensemble outputs according to two different voting thresholds. The high-precision ensemble output \mathbf{o}_{HP} is derived by setting

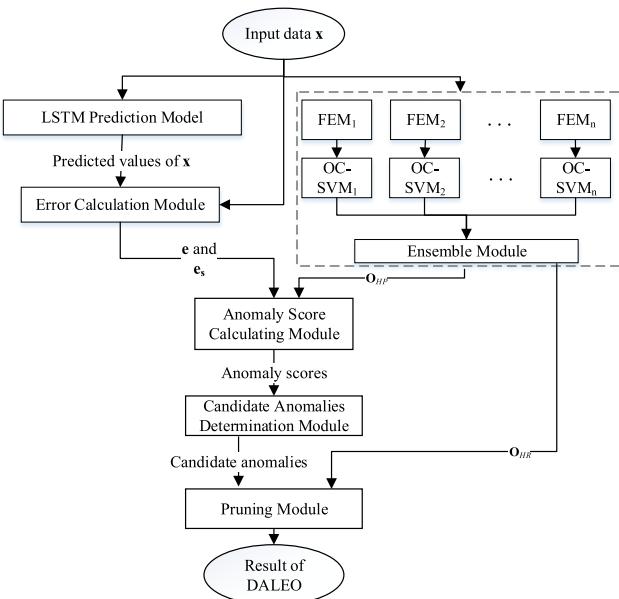


FIGURE 1. The main framework of DALEO.

a higher threshold η_1 , and the high recall ensemble output \mathbf{o}_{HR} is derived with a lower threshold η_2 . At the same time, the LSTM model is used to get the prediction values, based on which the error sequence \mathbf{e} and the smoothed error sequence \mathbf{e}_s are calculated. Then, the sequences \mathbf{e} , \mathbf{e}_s and \mathbf{o}_{HP} are used as the inputs of the anomaly score calculation module, and the anomaly score for each point in the sequence is calculated according to the proposed formula (see formula(3)). Next, the dynamic threshold is calculated based on the derived anomaly scores and the set of candidate anomalous sequences can be obtained according to the dynamic thresholds. Finally, two pruning strategies are applied using the ensemble output \mathbf{o}_{HR} and the candidate anomalous sequences to mitigate the false positives. The retained anomalies are the result of the anomaly detection of DALEO.

For ease of understanding, important concepts are presented by symbols. Table 1 shows the important symbols used in the proposed method and the corresponding meanings.

TABLE 1. Symbols and their meanings.

Symbol	Meaning
\mathbf{o}_{HP}	The ensemble output of high precision
\mathbf{o}_{HR}	The ensemble output of high recall
η_1	The voting threshold for \mathbf{o}_{HP}
η_2	The voting threshold for \mathbf{o}_{HR}
\mathbf{e}	Error sequence
\mathbf{e}_s	Smoothed error sequence
\mathbf{a}	Anomaly score sequence
$AR^{(i)}$	Confidence of the i th point belonging to an anomaly
$APR^{(j)}$	Confidence of the j th candidate anomalous sequence representing a true anomaly
L	The length of the test sequence
L_p	The input length of LSTM
L_{oc}	The input length of OC-SVM
$l_s^{(j)}$	The length of the j th candidate anomalous sequence

B. ENSEMBLE OF OC-SVMs

In this subsection, the original OC-SVM is introduced at first. Then, the analysis on \mathbf{o}_{HP} and \mathbf{o}_{HR} is given. Subsequently, how to get the ensemble outputs \mathbf{o}_{HP} and \mathbf{o}_{HR} from multiple OC-SVM models is illustrated. Finally, a concrete example of extracting different kinds of features as inputs of different OC-SVM models is given.

An OC-SVM model can judge whether a sample is abnormal or not. One-class Support Vector Machine, as one kind of one-class paradigm, learns to model patterns of a single class and distinguishes them from all other possible patterns by finding the maximal margin hyperplane which best separates the training data from other data [28]. The input of OC-SVM models for each sample can be the original data or the transformed features and the output is 1 or -1 , where 1 represents the input sample belongs to a normal sequence, while -1 represents the input sample belongs to an abnormal one. However, the performance of OC-SVM models is prone to low precision when it is used to handle time series, that is, many normal sequences are judged as abnormal. In DALEO, instead of using OC-SVM models to detect anomalies directly, the two kinds of ensemble results of multiple OC-SVM models are integrated into the anomaly score calculation module and pruning module respectively.

\mathbf{o}_{HP} as one of the inputs of anomaly score calculation, is used to locate the ‘anomalous parts’, for which the anomaly scores are augmented, while the anomaly scores of the rest of the sequence remain unchanged. Under this rule, the high precision of \mathbf{o}_{HP} is important. If the anomaly score of normal parts is augmented, it causes more false alarms eventually. On the other hand, the low recall of \mathbf{o}_{HP} is acceptable since it will just degrade to a trivial method in the worst case when no ‘anomalous parts’ is detected. \mathbf{o}_{HR} as one base of pruning, is used to mitigate the false positives from the set of candidate anomalous sequences. High recall of \mathbf{o}_{HR} needs to be guaranteed, otherwise the true positives will also be eliminated in the pruning process. Obviously, different thresholds should be applied to get \mathbf{o}_{HP} and \mathbf{o}_{HR} respectively.

The output of a OC-SVM model for each sample is 1 or -1 and the output sequence can be obtained through sliding windows for time series. Given the ensemble consists of n OC-SVM models which take different kinds of features as input, the length of test sequence is L , the input length of each OC-SVM model is L_{oc} , the stride of the sliding window is 1, the output of each OC-SVM model is $\mathbf{o}_j = \{o_j^{(1)}, o_j^{(2)}, \dots, o_j^{(i)}, \dots, o_j^{(L-L_{oc}+1)}\}$, where $o_j^{(i)} \in \{1, -1\}$ and $j = 1, 2, \dots, n$. Then the number of OC-SVM models whose output $o_j^{(i)} = -1$ is computed as

$$n_A^{(i)} = \sum_{j=1}^n \text{count}(o_j^{(i)} = -1) \quad (1)$$

for each point in the test sequence, where $0 \leq n_A^{(i)} \leq n$, and $i \in 1, 2, \dots, (L - L_{oc} + 1)$. We define

$$AR^{(i)} = n_A^{(i)} / n \quad (2)$$

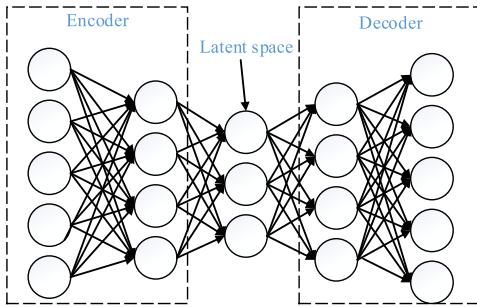


FIGURE 2. The basic structure of an autoencoder with fully-connected layer.

as the confidence of the i th point belonging to an anomaly, where $AR^{(i)} \in [0, 1]$, and the greater $AR^{(i)}$ is, the more likely the i th point belongs to an anomaly. Two voting thresholds η_1 and η_2 are set for \mathbf{o}_{HP} and \mathbf{o}_{HR} separately. $\mathbf{o}_{HP}^{(i)} = \{-1, AR^{(i)} > \eta_1\}$ and $\mathbf{o}_{HR}^{(i)} = \{-1, AR^{(i)} \geq \eta_2\}$ are obtained, where $\eta_1 > \eta_2$ and $i \in 1, 2, \dots, (L - L_{oc} + 1)$.

Diversity assures the value of ensemble. In the experiments of this paper, four Feature Extraction Modules and four OC-SVM models are specifically utilized for ensemble, the inputs of which include: the original time series \mathbf{x} , the manually extracted features $\mathbf{x}_{handcraft}$, the learnt features $\mathbf{x}_{encoded}$ through autoencoder, and $\mathbf{x}_{encoded_rff}$ which is Random Fourier features of $\mathbf{x}_{encoded}$ [31].

For the manually extracted features $\mathbf{x}_{handcraft}$, ten most widely used features are selected as Park *et al.* did in [38]:

- Mean
- Peak
- Standard deviation
- Root mean square
- Skewness
- Kurtness
- Crest factor
- Shape factor
- Pulse factor
- Margin factor

An autoencoder which consists of an encoder and a decoder is widely used as a feature extractor. The component of the encoder and the decoder can be fully-connected network, convolution network, or recurrent neural network, etc. [41]. The fully-connected network for the autoencoder is used in this case, whose basic structure is shown in Fig. 2. The encoder reads the input data and compresses the input information to generate the low dimension features as the output of hidden layer, which are fed into the decoder and used to reconstruct the original input data. Once the parameters of the autoencoder are well trained, the hidden layer outputs of the encoder $\mathbf{x}_{encoded}$ can be acquired as an effective feature representation of the original data. Then, the features $\mathbf{x}_{encoded_rff}$ are derived.

The above four kinds of inputs used in the ensemble are not necessarily the best combination, but they express the

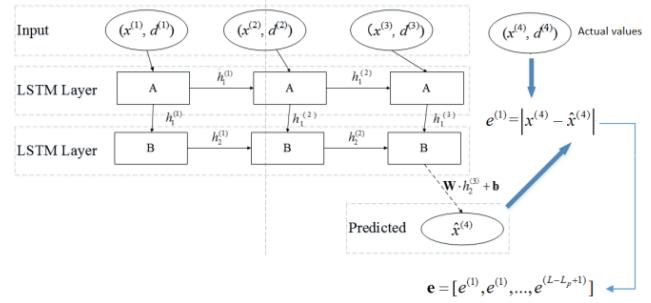


FIGURE 3. An example of using LSTM model to predict irregularly sampled time series and getting the error sequence with sliding windows. Module A and B represent the LSTM units of the first and the second layer respectively. $h_1^{(i)}$ is the output of the first layer of LSTM for $t^{(i)}$, while $h_2^{(i)}$ is the output of the second layer of LSTM for $t^{(i)}$. The weight matrix W and bias vector b are used for the dense layer to get the predicted value of x at the next time step. The error sequence can be obtained through sliding windows.

characteristics of time series from different perspectives and thus guarantee the diversity of the base models.

After feature extraction, the above different kinds of features can be fed into independent OC-SVM models to give their results respectively. Then, in Ensemble Module, the outputs are aggregated to get two ensemble outputs \mathbf{o}_{HP} and \mathbf{o}_{HR} for further steps in DALEO.

C. LSTM BASED ERROR CALCULATION FOR TIME SERIES WITH IRREGULAR INTERVALS

A single LSTM model is trained for each telemetry channel, and used to detect anomalies separately. Although LSTM can predict multiple variables simultaneously, the prediction performance degrades when the number of variables is large. Therefore, it is unrealistic to use only one or a few LSTM models to model all telemetry parameters of a spacecraft. More importantly, detecting anomalies on single telemetry channel is helpful to the rapid location of anomalies and provides convenience to aggregate the anomalies into the component level or subsystem level.

Given the unequal interval univariate telemetry sequence $\mathbf{x} = \{x^{(1)}, x^{(2)}, \dots, x^{(L)}\}$ and the corresponding time sequence $\mathbf{t} = \{t^{(1)}, t^{(2)}, \dots, t^{(L)}\}$, where the length of the test sequence is L . The time interval for each telemetry value is calculated as $d^{(i)} = t^{(i+1)} - t^{(i)}$ for each $i \in \{1, 2, \dots, L - 1\}$. Then the original time series can be represented by a two-dimensional matrix \mathbf{S} . Given the input length of LSTM is L_p , then the input for $t^{(i)}$ is

$$\mathbf{S}^{(i)} = \begin{bmatrix} x^{(i-L_p)}, & x^{(i-L_p+1)}, & \dots, & x^{(i-1)} \\ d^{(i-L_p)}, & d^{(i-L_p+1)}, & \dots, & d^{(i-1)} \end{bmatrix}$$

and the corresponding output is $\hat{x}^{(i)}$. The prediction error for each telemetry value is $e^{(i)} = |x^{(i)} - \hat{x}^{(i)}|$ and the error sequence \mathbf{e} is obtained through sliding windows. In Fig. 3, we illustrate how to get the error sequence \mathbf{e} using a LSTM prediction model and sliding windows for an irregular interval sequence with $L_p = 3$.

The error sequence \mathbf{e} is smoothed to dampen spikes that frequently occur with LSTM-based predictions [39], because abrupt changes in values are often not perfectly predicted which lead to sharp spikes in error sequence even when these changes conform to normal patterns. Exponentially-weighted average (EWMA) [30] is used to generate the smoothed error sequence \mathbf{e}_s , which is of the same length of \mathbf{e} .

D. ANOMALY SCORE BASED DYNAMIC THRESHOLD

The anomaly score of each telemetry value is calculated based on three parts: the prediction error, the smoothed error and the ensemble output \mathbf{o}_{HP} . Since \mathbf{e} and \mathbf{e}_s have the same length ($L - L_{oc} + 1$) and the length of \mathbf{o}_{HP} is ($L - L_p + 1$), the lengths of these three sequences do not equal when $L_{oc} \neq L_p$. But they can be matched into a unified time range for anomaly detection according to time indexes.

The anomaly score of each telemetry value is calculated as

$$a^{(i)} = \begin{cases} \alpha(e_s^{(i)} + \gamma^* e^{(i)}), & o_{HP}^{(i)} = -1 \\ e_s^{(i)} + \gamma^* e^{(i)}, & o_{HP}^{(i)} = 1 \end{cases} \quad (3)$$

and the anomaly score sequence \mathbf{a} is derived, where $o_{HP}^{(i)} = -1$ represents that the i th point belongs to an anomaly according to \mathbf{o}_{HP} , while $o_{HP}^{(i)} = 1$ represents the i th point belongs to an normal sequence according to \mathbf{o}_{HP} . $\alpha > 1$ is set so that the points whose $o_{HP}^{(i)} = -1$ have higher anomaly scores, and $\gamma \geq 0$ is set as the weight to balance the error value and the smoothed error value. The settings of α and γ , and how they influence the result of the anomaly detection are detailed in the next section.

The threshold for the derived anomaly scores is calculated dynamically – the parts whose anomaly scores exceed the threshold are regarded as candidate anomalies. A similar method to that used in [10] is applied to compute the dynamic thresholds, and the difference is that the smoothed errors are substituted with the proposed anomaly scores as inputs. A threshold ε is selected from the set:

$$\varepsilon = u(\mathbf{a}) + \mathbf{z}\sigma(\mathbf{a}) \quad (4)$$

such that:

$$\arg \max(\varepsilon) = \frac{\Delta u(\mathbf{a})/u(\mathbf{a}) + \Delta \sigma(\mathbf{a})/\sigma(\mathbf{a})}{|\mathbf{a}_a| + |\mathbf{A}_{seq}|^2} \quad (5)$$

where $\Delta u(\mathbf{a}) = u(\mathbf{a}) - u(\{a \in \mathbf{a} | a < \varepsilon\})$ represents the decrease in the mean of the anomaly scores, and $\Delta \sigma(\mathbf{a}) = \sigma(\mathbf{a}) - \sigma(\{a \in \mathbf{a} | a < \varepsilon\})$ represents the decrease in the standard deviation of the anomaly scores. \mathbf{a}_a represents all the points over the dynamic threshold, while each continuous sequence of $a \in \mathbf{a}_a$ makes up \mathbf{a}_{seq} , and all the sequences \mathbf{a}_{seq} make up the set of candidate anomalous sequences \mathbf{A}_{seq} .

Values evaluated for ε are determined by $z \in \mathbf{z}$ where \mathbf{z} is a set of positive values representing the number of standard deviations above $u(\mathbf{a})$. The dynamic threshold is set with the aim to cause the greatest percent decrease in the mean and standard deviation of the anomaly scores when values above are removed. The formula (3) also penalizes for having

large numbers of anomalous values ($|\mathbf{a}_a|$) and anomalous sequences ($|\mathbf{A}_{seq}|$). The optimal z depends on the distribution of anomaly scores, but experimental results suggest that the range between 0.5 to 10 always provides a proper choice for z . Once the dynamic threshold ε is determined, the resulting candidate anomalous sequences are also obtained.

E. MITIGATING FALSE POSITIVES

Anomaly score based dynamic thresholds often lead to many false positives, so it is necessary to reduce the false alarms. In anomaly detection, recall usually goes down when precision goes up. Therefore, it is necessary to eliminate false positives as much as possible with little reduction on recall. Two pruning strategies are applied sequentially according to the ensemble output \mathbf{o}_{HR} and the candidate anomalous sequences to reduce the number of the false positives.

If the value of a point in the sequence \mathbf{o}_{HR} is 1, it means that most of the OC-SVM models support that the point belongs to a normal sequence and the likelihood of the point belonging to an anomalous sequence is small. The sequence \mathbf{o}_{HR} shows how many points in each candidate anomalous sequence are judged as abnormal by \mathbf{o}_{HR} . If there are very few abnormal points in a candidate sequence, we argue that the likelihood of such a candidate anomalous sequence being a true positive is low and this sequence should be eliminated. Given the length of a candidate anomalous sequence $\mathbf{a}_{seq}^{(j)}$ is $l_s^{(j)}$, the corresponding sequence \mathbf{o}_{HR} contains $n_s^{(j)}$ points whose values equal to -1 , where $0 \leq n_s^{(j)} \leq l_s^{(j)}$. We define

$$APR^{(j)} = n_s^{(j)} / l_s^{(j)} \quad (6)$$

as the confidence of the j th candidate anomalous series being anomalous. p_1 is set as the threshold to determine whether a candidate anomalous sequence will be pruned. When $APR^{(j)} < p_1$, $\mathbf{a}_{seq}^{(j)}$ is eliminated from the set \mathbf{A}_{seq} .

The second strategy used to prune is similar to that in [10], but the smoothed errors is substituted with the anomaly scores as inputs, and the corresponding threshold is p_2 . Given $\mathbf{A}_{seq} = \{\mathbf{a}_{seq} | \text{for each } a_k \in \mathbf{a}_{seq}, s.t. a_k > \varepsilon\}$, a new set, \mathbf{a}_{max} , is created containing $\max(\mathbf{a}_{seq})$ for all \mathbf{a}_{seq} sorted in descending order. We also add the maximum anomaly score a'_m that belong to normal sequences to the end of \mathbf{a}_{max} . The consequent decrease percent is calculated as $d^{(j)} = (a_{max}^{(j)} - a_{max}^{(j+1)}) / a_{max}^{(j)}$ where $i \in \{1, 2, \dots, |\mathbf{a}_d|\}$. If $\exists j, s.t. d^{(j)} > p_2$, then all the candidate anomalous sequences whose maximum anomaly score is larger than that of $\mathbf{a}_{seq}^{(j)}$ are retained, otherwise the candidate anomalous sequences are pruned.

After the two-stage pruning, the remaining anomalous sequence is the final result of anomaly detection. In general, DALEO combines the ensemble outputs of multiple OC-SVMs with LSTM-based method in two stages for anomaly detection. DALEO is mainly designed for time series with unequal intervals, but it can also be used for trivial anomaly detection where telemetry data are regularly sampled.

IV. EXPERIMENTS

This section mainly includes the description of the datasets, the parameter setting of the base models and algorithms in section III, and the display and analysis of the experimental results.

A. DATASETS

A satellite anomaly detection dataset (SAT_AD¹) and a space shuttle anomaly detection dataset (SS_AD²), both of which are provided by NASA, are used to verify the effectiveness of DALEO. The original SAT_AD and SS_AD are datasets of equal intervals. In order to evaluate the proposed method, SAT_AD and SS_AD are randomly sampled and transformed into datasets with irregular intervals to simulate the telemetry data that might be obtained in a real scenario.

SAT_AD includes the telemetry data of SMAP (Soil Moisture Active Passive) satellite and MSL (Mars Science Laboratory) satellite as well as the corresponding annotations for anomalies. SAT_AD contains data from 82 channels, including data from 55 channels of SMAP satellite and 27 channels of MSL satellite. The training data and test data of each channel are continuous equal interval sequences. The training set only contains normal sequences, while each channel of the test set contains at least one abnormal subsequence. These abnormal sequences contain three types of anomalies: point anomalies, contextual anomalies and collective anomalies. In the annotation of this dataset, Hundman *et al.* [9] combined contextual anomalies and collective anomalies into one group and name both of them as contextual anomalies, since it is difficult to detect them by threshold or distance-based methods which do not take temporal relationship into account. Therefore, there are only two types of anomaly labels in the dataset: contextual anomalies which are related to temporal information and point anomalies which require no temporal context. Among the above 102 anomalies, 59 are point anomalies and 43 are contextual anomalies.

SS_AD contains the telemetry data about the Energize/De-Energize cycle of space shuttles. It contains three samples: TEK 14, TEK 16 and TEK 17. The telemetry values about the normal part of the dataset have strong periodicity, but the anomalies are very different and all of them belong to contextual anomalies. This dataset is relatively small, but it is also a popular anomaly detection dataset. It is mainly used to further verify the effectiveness of DALEO and the generalizability of the parameters introduced in the proposed method.

The training data and test data of the original dataset are randomly sampled on each channel according to a certain proportion, without changing the number of channels and anomalies in the original datasets. Due to the short duration of anomalies of some channels, random sampling may cause the disappearance of characteristic of anomalies in these

channels. Therefore, it is necessary to verify whether the characteristic of anomalies still exists after random sampling. If not, it is necessary to resample the channel randomly again until the characteristic of original anomalies is retained to a certain degree.

Half of the data in the SAT_AD dataset is randomly sampled, and the derived dataset is represented by SAT_AD-50. Since the anomalies in the SS_AD dataset are composed of shorter sequences, the anomaly characteristic can hardly be retained when the proportion of random sampling is low. 80% of the data in the SS_AD dataset is randomly sampled, and the derived dataset is represented by SS_AD-80. Channel D-12 and T-9 in SAT_AD-50 and SAT_AD are removed, since the training data of these two channels is insufficient after random sampling. Table 2 describes the brief information of the original datasets and the sampled datasets of SAT_AD-50 and SS_AD-80.

TABLE 2. The brief information of the datasets.

	SAT_AD	SAT_AD -50	SS_AD	SS_AD -80
Total anomaly sequences	102	102	4	4
Point anomalies	59	59	0	0
Contextual anomalies	43	43	4	4
Telemetry channels	80	80	3	3
No. of Telemetry values	487428	243714	3869	3095
Maximal time intervals	1	19	1	6

Fig. 4 shows some examples of point and contextual anomalies in SAT_AD-50, and the adjacent points have unequal time intervals, making the anomaly detection more difficult, especially for contextual anomalies.

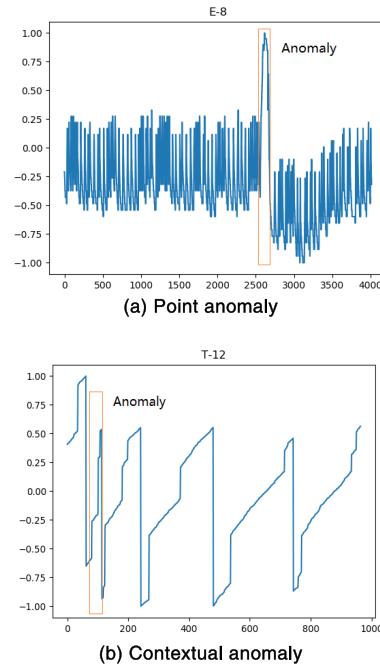


FIGURE 4. Examples of point anomalies and contextual anomalies in SAT_AD-50.

B. EXPERIMENTAL SETUP

Since DALEO involves the integration of multiple models, not only the parameters of each model need to be set properly,

¹ <https://github.com/khundman/telemanom>

² <https://github.com/chickenbestlover/RNN-Time-series-Anomaly-Detection>

but also the parameters related to the integration need to be set systematically.

The same hyper-parameters related to the LSTM model are used for SAT_AD and SAT_AD-50 as they are set in [10] to provide enough capacity to predict individual channels well. For all the datasets we use, each LSTM model is shallow with only two hidden layers and the last time step of the second layer output is used for prediction. As for SS_AD and SS_AD-80, input sequence length is increased to 400 because the data in these datasets have larger scale patterns than those in SS_AD, and thus more units in hidden layers are also used. The hyper-parameters related to the LSTM model on different datasets are shown in table 3.

TABLE 3. Parameters for the LSTM model.

	SAT_AD (SAT_AD-50)	SS_AD (SS AD-80)
units in hidden layers	[80,80]	[120,120]
input sequence length	250	400
training iterations	35	35
dropout	0.3	0.3
batch size	64	64
Optimizer	Adam	Adam

In contrast, shorter input sequence length and less units in hidden layers are set for the autoencoder model for all the above datasets to focus on local temporal features and ensure that the LSTM model and the autoencoder model extract features of different scales. The reduction of input sequence length and units in hidden layers also accelerates the training process. The main parameters of the autoencoder model are given in table 4.

TABLE 4. Parameters for the autoencoder model.

	SAT_AD (SAT_AD-50)	SS_AD (SS AD-80)
units in hidden layers	[40,20]	[80,40]
input sequence length	50	100
training iterations	20	20
batch size	32	32
optimizer	Adam	Adam

The LIBSVM package [40] is used to implement the OC-SVM algorithm, where ‘nu’ is the main tuning parameter which decides the upper bound on the fraction of training errors and a lower bound of the fraction of support vectors. ‘nu’ is set to 0.1 as a commonly used default.

In DALEO, five parameters are introduced: two weight coefficients α and γ for anomaly score calculation, two voting thresholds η_1 and η_2 , and one pruning threshold p_1 . η_1 is set to 0.6 while η_2 is set to 0.1 to make sure the high precision of \mathbf{o}_{HP} and the high recall of \mathbf{o}_{HR} . Based on multiple validation experiments, α is set to 1.3, γ is set to 0.3 and p_1 is set to 0.1. The final performance of anomaly detection is sensitive to α , γ and p_1 to some extent, and the effects of these parameters are discussed following experiment results.

In addition, the other pruning threshold p_2 is set to 0.13 as it was set in [10].

We adopt the same evaluation strategy as used in [10], [41]. A true positive is recorded if some portion of the predicted anomalous sequence of anomalies falls into a true labeled sequence. Only one true positive is recorded even if parts of multiple predicted sequences fall within one labeled sequence. A false negative represents no predicted sequences overlap with a positively labeled sequence. For a predicted anomalous sequence that does not overlap any labeled anomalous region, a false positive is recorded. The number of true positives is denoted as TP , the number of false negatives is denoted as FN , and the number of false positives is denoted as FP .

The metrics used for evaluating the final anomaly detection results include the following:

- Precision (P): $TP / (TP + FP)$;
- Recall (R): $TP / (TP + FN)$;
- F-measure (F_β): $F_\beta = (1 + \beta^2) \times P \times R / (\beta^2 P + R)$.

β is a non-negative real number to give different weights to precision and recall. When $\beta = 1$, the F-measure considers precision and recall simultaneously without any bias, and it is used to evaluate the overall performance of anomaly detection. $F_{0.5}$ which weights precision twice as much as recall is also commonly used in anomaly detection. Experiments on all datasets are repeated 5 times and the average results are reported for comparison.

C. EXPERIMENT RESULTS

First of all, we demonstrate the advantage of combining time intervals with telemetry data as inputs of LSTM model for anomaly detection with irregularly sampled data. For simplicity, such a method is denoted as Simple-LSTM. Several popular data imputation methods are selected as baselines, including two interpolation methods: Linear interpolation [19] and Cubic interpolation [20], and three other imputation methods: KNN [42], SoftImpute [43], and mean value based imputation. After data imputation, telemetry data are recovered to time series with equal intervals. For a fair comparison, all the above methods are combined with the state-of-the-art LSTM-based anomaly detection method in [9] for anomaly detection. These methods are listed as follows.

- Linear-LSTM: in this method, linear interpolation is used before the LSTM-based anomaly detection method.
- Cubic-LSTM: in this method, cubic interpolation, one of polynomial interpolation, is used before the LSTM-based anomaly detection method.
- Mean-LSTM: this method fills the missing data with mean of the observed data before using the LSTM-based method for anomaly detection.
- KNN-LSTM: this method fills the missing data with KNN algorithm based imputation before using the LSTM-based method for anomaly detection.

- Soft-LSTM: this method fills the missing data with Soft-Impute algorithm before using the LSTM-based method for anomaly detection.
- Simple-LSTM: in this method, the time intervals are calculated and combined with existing data as inputs of LSTM, and then the LSTM-based anomaly detection method is used.

The comparison results of the above methods on SAT_AD-50 are shown in table 5. For interpolation-based methods, Linear-LSTM outperforms Cubic-LSTM on both precision and recall. For other imputation-based methods, Mean-LSTM shows an obvious improvement on recall and a little higher precision than KNN-LSTM and Soft-LSTM. Simple-LSTM achieves the highest precision while the recall is slightly lower than the two interpolation-based methods. In general, interpolation-based methods seem to have advantages with respect to recall, while the performance of other imputation methods is inferior to Simple-LSTM on both precision and recall. In terms of $F_{0.5}$ and F_1 , Simple-LSTM beats all the baseline methods, which demonstrates the advantage of directly combining time intervals with telemetry data as inputs over imputation methods.

TABLE 5. Performance comparison of dealing with unequal intervals on SAT_AD-50.

	P	R	$F_{0.5}$	F_1
Linear-LSTM	77.27%	66.67%	0.7489	0.7158
Cubic-LSTM	64.08%	64.71%	0.6420	0.6439
Mean-LSTM	78.26%	52.94%	0.7143	0.6316
KNN-LSTM	70%	34.31%	0.5795	0.4605
Soft-LSTM	78.05%	31.37%	0.6015	0.4476
Simple-LSTM	85.33%	62.75%	0.796	0.7232

Table 6 describes the experiment results on SS_AD-80. For interpolation-based methods, Linear-LSTM outperforms Cubic-LSTM on precision. For other imputation-based methods, KNN-LSTM shows a better performance on precision than Mean-LSTM and Soft-LSTM. Simple-LSTM achieves the highest precision and recall. In general, Simple-LSTM shows an obvious advantage over the baselines, and further verifies the effectiveness of directly combining time intervals with telemetry data as inputs over imputation methods.

TABLE 6. Performance comparison of dealing with unequal intervals on SS_AD-80.

	P	R	$F_{0.5}$	F_1
Linear-LSTM	60.00%	75.00%	0.6250	0.6667
Cubic-LSTM	50.00%	75.00%	0.5357	0.6000
Mean-LSTM	40.00%	50.00%	0.4167	0.4444
KNN-LSTM	66.67%	50.00%	0.6250	0.5714
Soft-LSTM	50.00%	50.00%	0.5000	0.5000
Simple-LSTM	75.00%	75.00%	0.7500	0.7500

DALEO combines the OC-SVM ensemble outputs \mathbf{o}_{HP} and \mathbf{o}_{HR} with Simple-LSTM respectively, and it is compared with Simple-LSTM to see how much improvement

the combination can bring to the performance of anomaly detection. As can be seen from table 7, the performance of Simple-LSTM is badly influenced when Sat_AD is changed to Sat_AD-50 (recall decreased from 79.41% to 62.75%, and F_1 decreased from 0.8308 to 0.7232 despite the minor difference of the total anomaly sequences between these two datasets). DALEO outperforms simple-LSTM on recall significantly and shows a slight improvement of precision on Sat_AD-50. Although DALEO presents a lower precision than Simple-LSTM on Sat_AD, the overall performance (F_1) of DALEO is still better than that of Simple-LSTM due to the improvement on recall.

TABLE 7. Comparison of Simple-LSTM and DALEO on Sat_AD-50 and Sat_AD.

		Sat_AD-50	Sat_AD
<i>P</i>	Simple-LSTM	85.33%	87.10 %
	DALEO	85.88%	84.11%
<i>R</i>	Simple-LSTM	62.75%	79.41%
	DALEO	71.57%	85.71%
<i>F₁</i>	Simple-LSTM	0.7232	0.8308
	DALEO	0.7807	0.8491

Similar experiments are conducted on SS_AD-80 and SS_AD. As can be seen in table 8, DALEO outperforms simple-LSTM on recall with equal precision on SS_AD-80. On the other hand, although DALEO get a lower precision than Simple-LSTM on SS_AD, the overall performance of DALEO is still better than that of Simple-LSTM.

TABLE 8. Comparison of Simple-LSTM and DALEO on SS_AD-80 and SS_AD.

		SS_AD-80	SS_AD
<i>P</i>	Simple-LSTM	75.00%	100%
	DALEO	100%	80%
<i>R</i>	Simple-LSTM	75.00%	75.00%
	DALEO	75.00%	100%
<i>F₁</i>	Simple-LSTM	0.7500	0.8571
	DALEO	0.8571	0.8889

From table 7 and table 8, we find that DALEO shows an obvious advantage and both recall and precision may contribute to the overall performance on telemetry data with irregular intervals, though the improvement of recall might be accompanied with a small reduction of precision on datasets of equal intervals. In general, the overall performance of DALEO shows a better performance than Simple-LSTM, especially when the time series are of irregular intervals.

D. THE EFFECT OF PARAMETERS

In DALEO, three important parameters are introduced: α , γ and p_1 . In this subsection, the effects of these parameters are discussed.

We found that the dynamic thresholds and the resulting candidate anomalous sequences are relatively sensitive to the anomaly scores. Thus, both α and γ which determine the derived anomaly scores are very important parameters.

To show the trend of the performance of anomaly detection when α and γ are changed, two groups of box diagrams with respect to F_1 score on SAT_AD-50 dataset are presented. In Fig. 5(a), the trend of performance in the process of changing α from 1 to 1.6 is shown when $\gamma \in \{0, 0.05, 0.1, \dots, 0.5\}$; Fig. 5(b) depicts the trend of performance in the process of changing γ from 0 to 0.5 when $\alpha \in \{1.0, 1.05, 1.1, \dots, 1.6\}$.

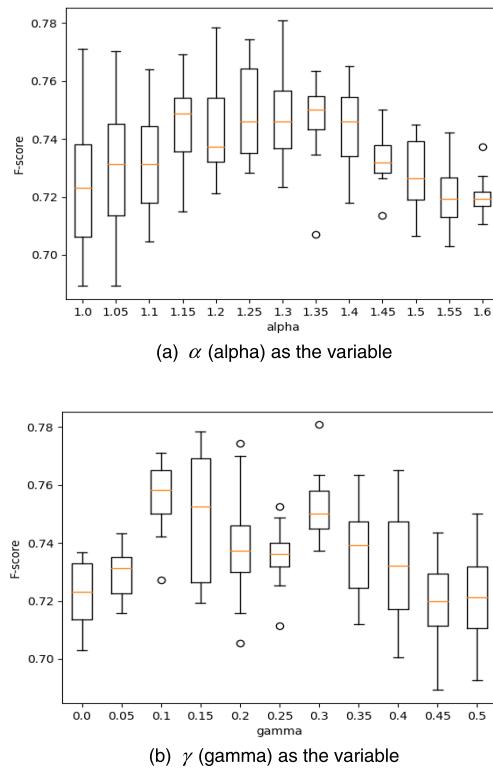


FIGURE 5. The box diagrams of F_1 score with different settings for α and γ .

As can be seen from Figure 5(a), F_1 score shows an obvious trend of rising first and then declines. When $\alpha \in [1.15, 1.4]$, F_1 score stays at a relatively high level. When $\gamma \in \{0, 0.05, 0.1, \dots, 0.5\}$ and $0 < \alpha \leq 1.6$, the corresponding F_1 score is greater than the F_1 score obtained when $\alpha = 0$, which indicates the effectiveness of integrating the high-precision ensemble output \mathbf{o}_{HP} into the proposed anomaly score calculation as long as the settings of α and γ are within a reasonable range.

As can be seen from Figure 5(b), when γ is changed from 0 to 0.5, F_1 score tends to rise first and then goes down. When $\gamma \in [0.1, 0.3]$, F_1 score fluctuates at a high level. When $\alpha \in \{1.0, 1.05, 1.1, \dots, 1.6\}$ and $0 < \gamma \leq 0.4$, the corresponding F_1 score is greater than the F_1 score obtained when $\gamma = 0$, which reveals that the integration of error sequence is effective when the setting of γ is in a reasonable range.

We also discuss how the pruning threshold p_1 , which is related to \mathbf{o}_{HR} , influences the performance of anomaly detection. Fig. 6 shows the performance change when p_1 is

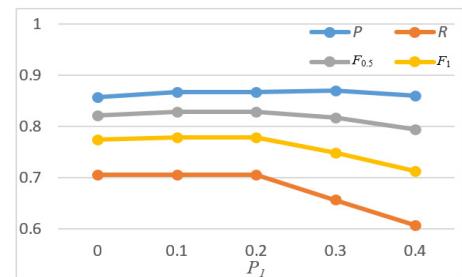


FIGURE 6. The trend of performance when increasing p_1 .

increased while other parameters remain unchanged. At first, precision presents an inclination to increase, but turn to decrease after p_1 exceeds 0.3. Obviously, increasing p_1 can only reduce recall or keep it unchanged at best. In this case, recall goes down when p_1 exceeds 0.2. Apparently, a large value for p_1 will certainly lead to performance degradation. On the contrary, when p_1 is too small, the p_1 related pruning will have little effect in reducing the number of false positives. From the above analysis, a value in the range [0.1, 0.2] seems to be appropriate for p_1 .

E. INTERNAL ANALYSIS OF DALEO

In this subsection, we discuss the effect of integrating the OC-SVM ensemble outputs into LSTM-based anomaly detection for thorough evaluation. The integration includes two stages: the first one is integrating the high precision ensemble output \mathbf{o}_{HP} into the anomaly score, and the second one is utilizing the high recall ensemble output \mathbf{o}_{HR} for pruning.

First, we compare the method with no integration (Simple-LSTM) and DALEO without the \mathbf{o}_{HR} based pruning (denoted as DALEO-WP). The comparison result of these two methods on SAT_AD-50 dataset is shown in Fig. 7.

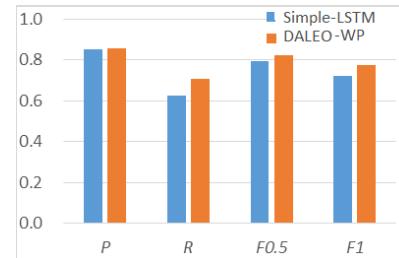


FIGURE 7. Comparison between DALEO-WP and Simple-LSTM.

From Fig. 7, it can be seen that DALEO-WP beats simple-LSTM on all the evaluation metrics, and improves the recall significantly without reducing precision. Therefore, this experiment confirms the effect of augmenting the anomaly score according to the high precision ensemble output \mathbf{o}_{HP} .

Then, the results of DALEO-WP and DALEO are compared on SAT_AD-50 to verify the effect of pruning based on \mathbf{o}_{HR} . Compared with the DALEO-WP, DALEO only has

one more step of pruning. Therefore, recall of the DALEO cannot exceed that of DALEO-WP, and can only be equal at best. On the other hand, increasing a pruning procedure also take the risk of decreasing precision, because there is a risk of reducing the true positives while eliminating the false positive ones.

As shown in Fig. 8, DALEO improves the precision without reducing recall, further improves the overall performance of anomaly detection, and verifies the effectiveness of using \mathbf{o}_{HR} for pruning.

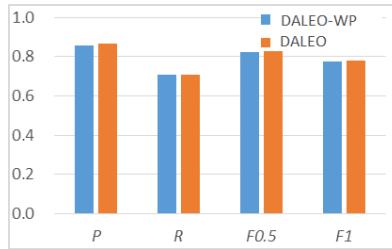


FIGURE 8. Comparison between DALEO-WP and DALEO.

V. CONCLUSION

In this paper, the impact of inaccurate prediction on the performance of anomaly detection based on single LSTM method is analyzed, and this impact is expanded when confronted with the irregular interval telemetry data. In order to improve the performance of anomaly detection, DALEO is proposed to integrate the two unsupervised anomaly detection methods, OC-SVM and LSTM. By introducing different voting thresholds, DALEO aggregate the results of multiple OC-SVM models, obtaining ensemble outputs of high precision and high recall. Then, the ensemble outputs are integrated into the anomaly score calculation module and pruning module of anomaly detection method in a novel way. This integration method is generic, which means the LSTM prediction model can be replaced by any other RNN model.

Extensive experiments have been conducted on two real-world datasets. Firstly, the effectiveness of concatenating time intervals with telemetry data for anomaly detection when data are irregularly sampled are verified. Then DALEO is compared to the baseline and the results demonstrate the obvious advantage of the proposed method. DALEO can also be used in the trivial anomaly detection problem where time series are regularly sampled. Despite the fact that DALEO improves precision at the cost of a small decrease in recall when dealing with equal interval time series on the two datasets, the overall performance of DALEO is still better than that of the baseline methods. Although the final result of anomaly detection seems to be sensitive to the introduced parameters and the optimal result may not be obtained easily, DALEO always provide an improvement as long as the parameters are within a reasonable range. Finally, the validity of the respective function of \mathbf{o}_{HP} and \mathbf{o}_{HR} are verified.

DALEO involves the integration of multiple models, but these models can be trained offline and used for anomaly detection in parallel. The computation cost of the rest parts

of DALEO almost equals to the traditional LSTM-based anomaly detection. Thus, DALEO can provide a performance improvement without much loss in efficiency.

Accurate prediction model is critical to this approach and the employed LSTM model can be substituted with any improved prediction model. Some widely used tricks of RNN, like Attention mechanism or transformation skills, have the potential to further improve the prediction capacity of the prediction model and the final anomaly detection performance. In the near future, reinforcement learning will be used so that the introduced parameters can be adjusted automatically. Then DALEO can be used to monitor important spacecrafts without human intervention continuously and reliably.

REFERENCES

- [1] M. Xie, S. Han, B. Tian, and S. Parvin, "Anomaly detection in wireless sensor networks: A survey," *J. Netw. Comput. Appl.*, vol. 34, no. 4, pp. 1302–1325, 2011.
- [2] M. Rolincik, M. Lauriente, H. C. Koons, and D. Gorney, "An expert system for diagnosing environmentally induced spacecraft anomalies," in *Proc. 5th Ann. Space Oper. Appl. Res. Symp.*, 1992, pp. 36–44.
- [3] N. Nishigori, M. Hashimoto, A. Choki, and M. Mizutani, "Fully automatic and operator-less anomaly detecting ground support system for mars probe 'NOZOMI,'" in *Proc. 6th Int. Symp. Artif. Robot. Automat. Space (I-SAIRAS)*, 2001, pp. 1–8.
- [4] D. Iverson, "Data mining applications for space mission operations system health monitoring," in *Proc. SpaceOps Conf.*, May 2008, p. 3212.
- [5] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek, "LoOP: Local outlier probabilities," in *Proc. 18th ACM Conf. Inf. Knowl. Manage.*, 2009, pp. 1649–1652.
- [6] Y. Gao, T. Yang, M. Xu, and N. Xing, "An unsupervised anomaly detection approach for spacecraft based on normal behavior clustering," in *Proc. 5th Int. Conf. Intell. Comput. Technol. Automat.*, Jan. 2012, pp. 478–481.
- [7] K. Li, Y. Wu, S. Song, Y. Sun, J. Wang, and Y. Li, "A novel method for spacecraft electrical fault detection based on FCM clustering and WPSVM classification with PCA feature extraction," *Proc. Inst. Mech. Eng., G, J. Aerosp. Eng.*, vol. 231, no. 1, pp. 98–108, Jan. 2017.
- [8] T. Yairi, Y. Kawahara, R. Fujimaki, Y. Sato, and K. Machida, "Telemetry-mining: A machine learning approach to anomaly detection and fault diagnosis for space systems," in *Proc. 2nd IEEE Int. Conf. Space Mission Challenges Inf. Technol. (SMC-IT)*, Jul. 2006, p. 8.
- [9] R. Fujimaki, T. Yairi, and K. Machida, "An approach to spacecraft anomaly detection problem using kernel feature space," in *Proc. 11th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2005, pp. 401–410.
- [10] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 387–395.
- [11] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1544–1551, Jul. 2018.
- [12] A. Zhao, L. Qi, J. Dong, and H. Yu, "Dual channel LSTM based multi-feature extraction in gait for diagnosis of neurodegenerative diseases," *Knowl.-Based Syst.*, vol. 145, pp. 91–97, Apr. 2018.
- [13] L. Bontemps, V. L. Cao, J. McDermott, and N.-A. Le-Khac, "Collective anomaly detection based on long short-term memory recurrent neural networks," in *Proc. Int. Conf. Future Data Secur. Eng.*, 2016, pp. 141–152.
- [14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.
- [15] T.-Y. Kim and S.-B. Cho, "Web traffic anomaly detection using C-LSTM neural networks," *Expert Syst. Appl.*, vol. 106, pp. 66–76, Sep. 2018.
- [16] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "LSTM-based encoder-decoder for multi-sensor anomaly detection," 2016, *arXiv:1607.00148*. [Online]. Available: <http://arxiv.org/abs/1607.00148>
- [17] A. Nanduri and L. Sherry, "Anomaly detection in aircraft data using recurrent neural networks (RNN)," in *Proc. Integr. Commun. Navigat. Surveill. (ICNS)*, Apr. 2016, pp. 5C2-1–5C2-8.

- [18] R. Zhang and Q. Zou, "Time series prediction and anomaly detection of light curve using LSTM neural network," *J. Phys., Conf. Ser.*, vol. 1061, Jul. 2018, Art. no. 012012.
- [19] J. L. Schafer and J. W. Graham, "Missing data: Our view of the state of the art," *Psychol. Methods*, vol. 7, no. 2, p. 147, Jun. 2002.
- [20] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent neural networks for multivariate time series with missing values," *Sci. Rep.*, vol. 8, no. 1, Dec. 2018, Art. no. 6085.
- [21] D. M. Kreindler and C. J. Lumsden, "The effects of the irregular sample and missing data in time series analysis," in *Nonlinear Dynamical Systems Analysis for the Behavioral Sciences Using Real Data*. Boca Raton, FL, USA: CRC Press, 2016, pp. 149–172.
- [22] C. De Boor, E.-U. Mathématicien, and C. De Boor, *A Practical Guide to Splines*, vol. 27. New York, NY, USA: Springer-Verlag, 1978.
- [23] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *J. Mach. Learn. Res.*, vol. 11, pp. 2287–2322, Jan. 2010.
- [24] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.
- [25] D. Mondal and D. B. Percival, "Wavelet variance analysis for gappy time series," *Ann. Inst. Stat. Math.*, vol. 62, no. 5, pp. 943–966, Oct. 2010.
- [26] J. Josse and F. Husson, "Handling missing values in exploratory multivariate data analysis methods," *J. Société Française Statistique*, vol. 153, no. 2, pp. 79–99, 2012.
- [27] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor AI: Predicting clinical events via recurrent neural networks," in *Proc. Mach. Learn. Healthcare Conf.*, 2016, pp. 301–318.
- [28] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 582–588.
- [29] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI," *Large-Scale Kernel Mach.*, vol. 34, no. 5, pp. 1–41, 2007.
- [30] H. J. Stuart, "The exponentially weighted moving average," *J. Qual. Technol.*, vol. 18, no. 4, pp. 203–210, 1986.
- [31] M.-N. Nguyen and N. A. Vien, "Scalable and interpretable one-class SVMs with deep learning and random Fourier features," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, 2018, pp. 157–172.
- [32] T. Ergen, A. H. Mirza, and S. S. Kozat, "Unsupervised and semi-supervised anomaly detection with LSTM neural networks," 2017, *arXiv:1710.09207*. [Online]. Available: <http://arxiv.org/abs/1710.09207>
- [33] Y. Ding and A. Ross, "An ensemble of one-class SVMs for fingerprint spoof detection across different fabrication materials," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2016, pp. 1–6.
- [34] C. C. Aggarwal and S. Sathe, "Theoretical foundations and algorithms for outlier ensembles?" *ACM SIGKDD Explor. Newslett.*, vol. 17, no. 1, pp. 24–47, Sep. 2015.
- [35] A. Chiang and Y.-R. Yeh, "Anomaly detection ensembles: In defense of the average," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol. (WI-IAT)*, Dec. 2015, pp. 207–210.
- [36] J. Gao and P.-N. Tan, "Converting output scores from outlier detection algorithms into probability estimates," in *Proc. 6th Int. Conf. Data Mining (ICDM)*, Dec. 2006, pp. 212–221.
- [37] E. Yu and P. Parekh, "A Bayesian ensemble for unsupervised anomaly detection," 2016, *arXiv:1610.07677*. [Online]. Available: <https://arxiv.org/abs/1610.07677>
- [38] D. Park, S. Kim, Y. An, and J.-Y. Jung, "LiReD: A light-weight real-time fault detection system for edge computing using LSTM recurrent neural networks," *Sensors*, vol. 18, no. 7, p. 2110, Jun. 2018.
- [39] D. T. Shipmon, J. M. Gurevitch, P. M. Piselli, and S. T. Edwards, "Time series anomaly detection; detection of anomalous drops with limited features and sparse examples in noisy highly periodic data," 2017, *arXiv:1708.03665*. [Online]. Available: <http://arxiv.org/abs/1708.03665>
- [40] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, p. 27, 2011.
- [41] H. Xu, Y. Feng, J. Chen, Z. Wang, H. Qiao, W. Chen, N. Zhao, Z. Li, J. Bu, Z. Li, Y. Liu, Y. Zhao, and D. Pei, "Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in Web applications," in *Proc. World Wide Web Conf. (WWW)*, 2018, pp. 187–196.



JUNFENG WU received the M.Sc. degree in management science and engineering from the National University of Defense Technology, Changsha, China, in 2017, where he is currently pursuing the Ph.D. degree in management science and engineering.

His research interest includes classification and anomaly detection of time series.



LI YAO received the B.Sc. degree in computer software from the School of Computer Science, Nankai University, in 1985, the M.Sc. degree in artificial intelligence from the School of Computer Science, National University of Defense Technology (NUDT), in 1990, and the Ph.D. degree in information system engineering from the School of Information System and Management, NUDT, in 1995.

She is currently a Professor with the School of System Engineering, NUDT. She is the author of more than 100 articles. Her research interests include multi-agent systems, machine learning, knowledge engineering, and intelligent decision.



BIN LIU received the bachelor's degree from Central South University, Changsha, China, in 2011, and the M.Sc. and Ph.D. degrees in management science and engineering from the National University of Defense Technology, Changsha, in 2013 and 2018, respectively.

He is currently a Lecturer with the Science and Technology on Information Systems and Engineering Laboratory, National University of Defense Technology. His research interests include knowledge engineering, machine learning, and multi-agent systems for situational awareness.



ZHEYUAN DING received the bachelor's degree from Jilin University, Changchun, China, in 2012, and the M.Sc. degree in management science and engineering from the National University of Defense Technology, Changsha, China, in 2014, where he is currently pursuing the Ph.D. degree in management science and engineering. His research interests include knowledge engineering and pattern recognition.



LEI ZHANG received the bachelor's degree from Space Engineering University, Beijing, China, in 2005, and the master's degree in engineering with the major of aerospace science and technology from Tsinghua University, Beijing, in 2017.

He is currently a Satellite Engineer with the Xi'an Satellite Control Center, Xi'an, China. His research interests include knowledge engineering, machine learning, and satellite measurement and control.