



# Characterizing and Modeling for Proactive Disk Failure Prediction to Improve Reliability of Data Centers

Tuanjie Wang, Xinhui Liang, Quanguan Xie, Qiang Li<sup>(✉)</sup>, Hui Li, and Kai Zhang

State Key Laboratory of High-end Server and Storage Technology,  
Beijing, China

{wangtuanjie, liangxinhui, xiequanguan,  
li.qiangbj}@inspur.com

**Abstract.** In modern datacenter, hard disk drive has the highest failure rate. Current storage system has data protection feature to avoid data loss caused by disk failure. However, data reconstruction process always slows down or even suspends system services. If disk failures can be predicted accurately, data protection mechanism can be performed before disk failures really happen. Disk failure prediction dramatically improve the reliability and availability of storage system. This paper analyzes disk SMART data features in detail. According the analysis results, we design an effective feature extraction and preprocessing method. And we have optimized the XGBoost's hyperparameters. Finally, ensemble learning is applied to further improve the accuracy of prediction. The experimental results of Alibaba data set show that our system predict disk failures within 30 days. And the F-score achieves 39.98.

**Keywords:** XGBoost · Feature engineering · Hyperparameter tuning · Ensemble learning

## 1 Introduction

Large-scale data center usually has millions of hard disks. Disk failure will decrease the stability and reliability of the storage system. And it may even endanger the entire IT infrastructure, and affect the business SLA. If disk failures were predicted in advance, data can be backed up or migrated during the spare time. Disk failure prediction can greatly reduce data loss and effectively improve the reliability of the data center.

SMART (Self-Monitoring, Analysis and Reporting Technology) [1] is a monitoring data supplied by HDD, solid-state drives (SSDs), and eMMC drives. All modern HDD manufacturers support the SMART specification. Currently, it's common to predict disk failures using on SMART data and AI technology. A SMART datasets [2] was provided to contestants by PAKDD2020 Alibaba AI Ops Competition [3]. Our disk prediction model was verified on it.

There are a large amount of related work on predicting disk failures. For example, Hongzhang [4] proposed an active fault-tolerant technology based on the “acquisition-prediction-migration-feedback” mechanism. Sidi [5] proposed a method of combining disk IO, host IO and location information for fault prediction. Based on CNN and LSTM neural network algorithm, this method can extract features and train model automatically. Yong [6] proposed an online disk failures prediction method named CDEF. CDEF combine disk-level SMART signals and system-level signals. CDEF use a cost-aware rank model to select the top  $r$  disks that are most likely to have errors. Yanwen [7] proposed a disk failures prediction and interpretation method DFPE. By extracting relevant features, DFPE derives the prediction rules of the model. DFPE evaluates the importance of the features, then improves the interpretability of complex models. Ganguly [8] utilized SMART and hardware-level features such as node performance counter to predict disk failure. Ma [9] investigate the impact of disk failures on RAID storage systems and designed RAIDShield to predict RAID-level disk failures. Nicolas [10] uses SVM, RF and GBT to predict disk failures. And it reaches 67% recall. Tan [11] proposed an online anomaly prediction method to foresee impending system anomalies. They applied discrete-time Markov chains to model the evolving patterns of system features, then used tree augmented naive Bayesian to train anomaly classifier. Dean [12] proposed an Unsupervised Behavior Learning system, which leverages an unsupervised method self organizing map to predict performance anomalies. Wang [13] also proposed an unsupervised method to predict disk anomaly based on mahalanobis distance. Ceph [14] has disk fault prediction features. It needs to train SMART raw data for 12 days and use SVM [15] to predict disk failures.

However, due to the complexity of the actual production environment, noisy data, and other uncertainties, developing a disk failure prediction system that can be used in large-scale data centers is very challenging:

- The positive and negative samples are extremely imbalanced. The reason is system downtime caused by disk failure occurred infrequently. Actually, for small-scale or short-load disk storage systems, the number of failed disks is very small.
- The change of S.M.A.R.T. values is difficult to predict. According to our observation, S.M.A.R.T. values will change only when the disk is near failure, and sometimes change suddenly. In addition, when the disk is healthy, its S.M.A.R.T. value could be large and stable. Therefore, it cannot rely only on the absolute value of S.M.A.R.T.
- The generalization ability of prediction model is insufficient. There are a large number of disks of different models or even different manufacturers in the same data center. If the generalization ability of the prediction model is not strong, it is difficult to obtain high-performance prediction results.

The contributions of this article are as follows:

- Through data exploration, SMART range analysis, changepoint analysis and other methods, we found several SMART attributes that are strongly correlate to disk failure. We determine time series feature extraction method and sliding window size. We establish the principle of labeling positive and negative samples.

- In order to eliminate the differences in feature data distribution of different models, feature scaling is performed during data preprocessing. As a result, a unified model can simplify the deployment process, and improve the model generalization ability.
- During model training stage, firstly, we choose XGBoost [16] algorithm as base model, which is simple and efficient. Then we fine-tune model parameters. Finally, soft voting method is used to ensemble each sub-model, and further improving the prediction performance of the model.

The rest of this paper is organized as follows: In Sect. 2, we describes the proposed approach and details. The evaluation of our approach and experiment results are described in Sect. 3. Section 4 presents conclusion.

## 2 Solution

In this section, we present our disk failure prediction approach. Figure 1 shows the overview of the approach.

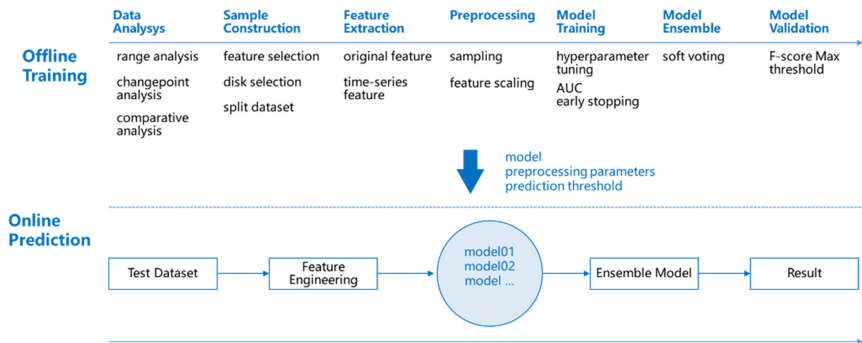


Fig. 1. Disk failure prediction overview.

Firstly, we analyze the internal distribution law of SMART data through data exploration, select representative healthy and faulty disks to construct positive and negative samples, identify fault-related SMART features and extract time series features. Feature scaling is performed during data preprocessing, and the impact of different ranges between different disk models and different SMART features can be eliminated. Secondly, based on the scaled dataset, we construct binary classification model and tune its hyperparameters. Finally, we integrate sub-models, verify integrated model using validation dataset, and take the threshold at the maximum F-score on the verification dataset as the optimal threshold.

We then import the trained model, preprocessing parameters and prediction threshold, and make online prediction on the test dataset and output the final prediction result.

## 2.1 Feature Selection

Through statistical analysis, we found that there are a large number of empty columns in the SMART dataset, and only 48 of the total 510 columns are non-empty. Then, the SMART probability density distribution and KL divergence were calculated for healthy and faulty disks, and SMART 5,187,192,193,197,198 and 199 were selected, which are related to disk failure and have big KL divergence. The KL divergence of all these selected features is positive infinity. As shown in Fig. 2, the KL divergence of SMART 198 is positive infinity, and the distribution of SMART 198 is mainly concentrated near zero for both healthy and faulty disks, and the main difference exists in the long tail on the right side. This part of data is useful for distinguishing between faulty disks and healthy disks. However, the distribution of SMART 194 has a high degree of coincidence, and the KL divergence is only 0.015, this means that it is difficult to distinguish between healthy and faulty disks through SMART 194.

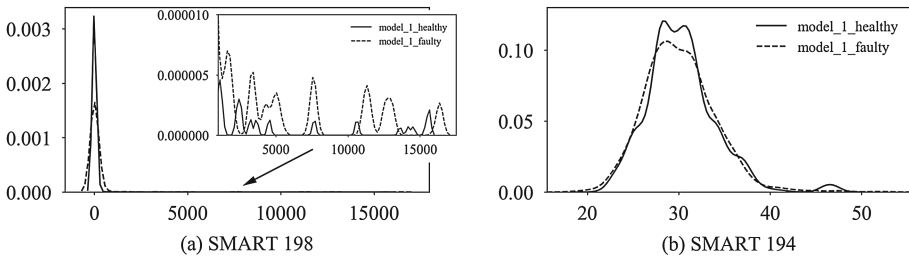


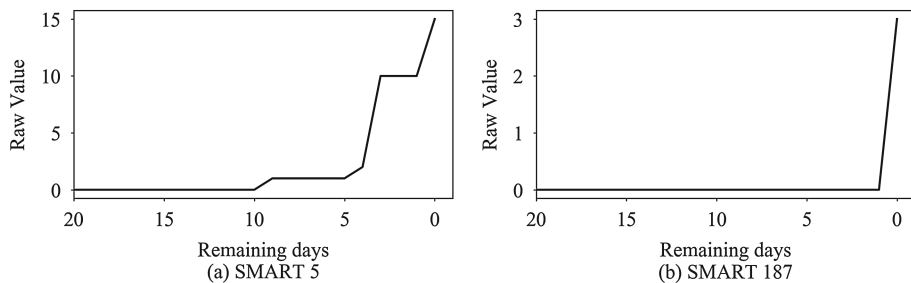
Fig. 2. SMART 198 and 194 probability density distribution.

## 2.2 Feature Analysis

For the key smart features selected by the feature selection above, further analysis is made from the following three dimensions.

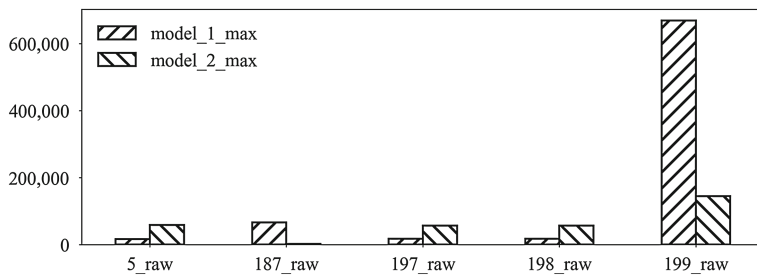
The first is range analysis. Statistics show that only around 5000 of the healthy disks exist non-zero value. Compared with all-zero disks, these disks contain more useful information. We should focus on these high-value healthy disk data when constructing model.

Secondly, changepoint analysis was performed on the SMART of faulty disks. It was found that even in the last 7 days of the faulty disks, the values of 50%–75% of these features such as SMART 5, SMART 187 are zero. And the faulty disk will not change significantly until the last 1–15 days of the life. As shown in Fig. 3, the SMART 5 of this disk did not change until the last 10 days, and did not increase significantly until the last 4 days, and SMART 187 did not change until the last 1 day. This phenomenon commonly occurs on faulty disks, that is, the closer to the end of life, the more likely sudden change will occur. Therefore, when constructing a positive sample, it is best to choose the last 0–7 days of the faulty disks, and the sliding window for extracting time series features is most suitably set between 3 and 7.



**Fig. 3.** SMART 5 (left) and 187 (right) trend graph.

Finally, by horizontal comparison and analysis of different disk models model1 and model2, it is found that the difference in the value range of each SMART feature between model 1 and model 2 is significant (As shown in Fig. 4). By scaling the SMART features of different models in the preprocessing stage, the SMART features of each model are firstly scaled to the same range, and then data is scaled again by standard scaler for training to eliminate smart distribution difference of each model. In this way, we obtain a unified model, optimize the prediction effect and improve model generalization ability successfully.



**Fig. 4.** SMART maximum comparison between model1 and model2.

### 2.3 Preprocessing

We use the dataset provided by Alibaba to complete our approach. The data from July 2017 to July 2018 is used for training, and the data of August 2018 is used for offline validation. Tianchi Alibaba uses data of September 2018 for online testing.

In the training dataset, there are a total of 184,305 disks, including 1,272 faulty disks and 183,033 healthy disks. Among all the disks, only 5,953 are not-all-empty. The judgment rule about not-all-empty is that the values of the main features (smart\_5raw, smart\_187raw, smart\_197raw, smart\_198raw, smart\_199raw) are not all 0 or empty during the entire life cycle of the disk. For the training dataset, healthy and faulty disks are down-sampled at 10: 1, and around 5,000 not-all-empty healthy disks were added as supplements.

The missing values of the original data are filled with forward padding method to ensure the continuity of the time series.

To solve the problem of sample imbalance, we only select data samples for training from the last 7 days, and the last 30th, 40th, 50th, and 60th days of each disk. Then mark the last 7 days of the faulty disks as positive and other data samples as negative.

Time series feature extraction is performed on key SMART features on every day sampled. The sliding window is 3 days, 5 days, and 7 days. The extraction method is shown in the following Table 1.

**Table 1.** Time series feature extraction.

Number	Feature extraction method	Detail
1	Change time	The number of attribute changes within a period
2	Change rate	The slope of attribute values within a period
3	Std	The standard variance of attribute values within a period

Scale the SMART data of model2 to the range of the minimum and maximum of model1. Taking the feature  $F_n$  of model2 that is scaled to model1 as an example, first calculate the scaling factor, then scale the feature  $F_n$  of model2 to  $F_{n_{scaled}}$ .

$$scale = \frac{\max(model1\_Fn) - \min(model1\_Fn)}{\max(model2\_Fn) - \min(model2\_Fn)} \quad (1)$$

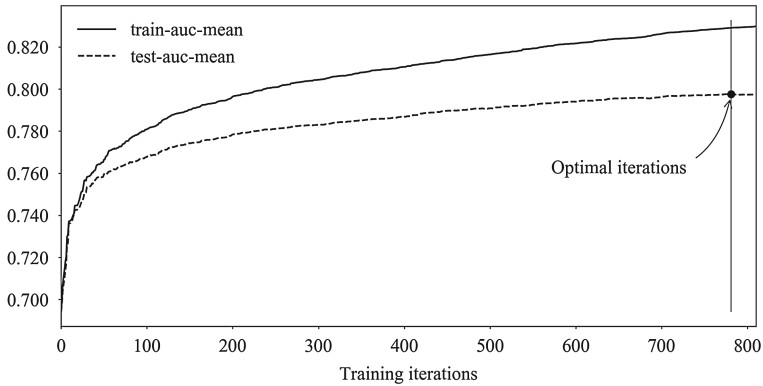
$$F_{n_{scaled}} = scale \times (Fn - \min(model2\_Fn)) + \min(model1\_Fn) \quad (2)$$

Finally, a standard method is used to scale the dataset.

## 2.4 Model Training

Our approach finally chose XGBoost [17] algorithm for model training, because the number of samples and the number of features in the data set are relatively small, and there is no need for very complicated models. At the same time, the hyperparameters of XGBoost are easy to adjust, and XGBoost is not easy to overfit. By comparing experimental results, it is found that the prediction results of XGBoost are better than Random Forest [18] and LSTM [18, 19].

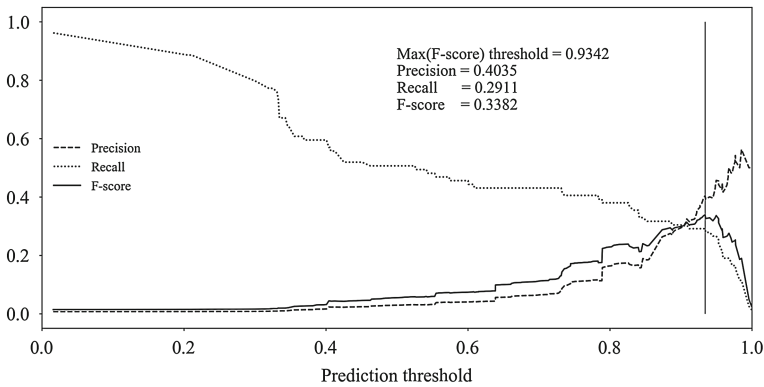
We use 3-folder cross-validation for model training, and use AUC as the evaluation function. Compared with the PRC evaluation function, AUC is not sensitive to the rate of positive and negative samples. The AUC learning curve during training is shown in the Fig. 5. When the AUC is no longer improved, the optimal number of iterations of XGBoost can be determined.



**Fig. 5.** AUC learning curve during training.

By using grid search to optimize XGBoost hyperparameters, such as `max_depth`, `scale_pos_weight` and so on, it was found that the prediction results were not improved significantly on validation dataset.

Finally, we use the validation dataset to obtain the prediction probability. As shown in the Fig. 6, the best prediction threshold is the classification threshold with maximum F-score.



**Fig. 6.** F-score, Recall and Precision change curve with prediction threshold.

### 2.5 Model Ensemble

The ensemble of sub-models can effectively improve the generalization ability of the prediction model. We finally selected 6 sub-models that perform well on the validation set. These sub-models use XGBoost as the basic algorithm. The difference between them is mainly in the preprocessing, such as different SMART features, different feature extraction methods and sliding windows. Detailed parameters of these six

sub-models are shown in the Fig. 7. The final prediction probabilities of the integrated models are obtained by averaging the prediction probabilities of these six models.

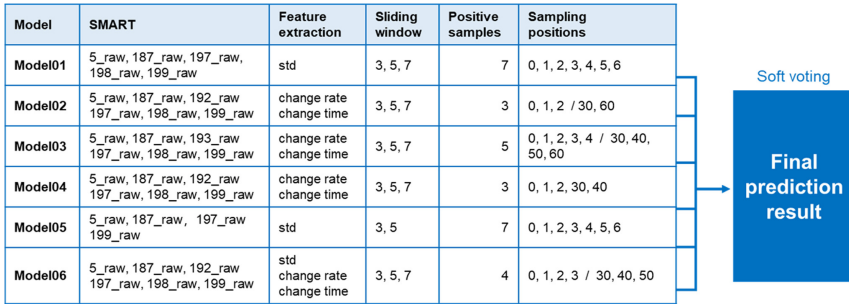


Fig. 7. Model ensemble method.

The positive samples and sampling positions in the Fig. 7 are related to the sampling process in Sect. 2.3.

### 3 Evaluation

#### 3.1 Evaluation Metric

According to the Alibaba’s requirement, prediction engine predicts the failure disks in the next 30 days. We used the precision, recall and F-score evaluation metrics redefined in the competition rules [3].

Recall reflects the proportion of positive samples correctly judged to the total positive samples, and Precision reflects the proportion of true positive samples among the positive samples decided by the classifier. The higher Recall and Precision, the better. F1-Score is the weighted average of Recall and Precision. F-score takes into account both Recall and Precision.

The metrics are defined as follows:

$$Precision = \frac{n_{pp}}{n_{pp}} \tag{3}$$

$$Recall = \frac{n_{pr}}{n_{pr}} \tag{4}$$

$$F - score = 2 \times \frac{Precision \times Recall}{Precision \times Recall} \tag{5}$$

The following Table 2 explains  $n_{pp}$ ,  $n_{pp}$ ,  $n_{pr}$  and  $n_{pr}$ .



**Table 2.** Evaluation metric detail.

Number	Metric	Detail
1	$n_{pp}$	The number of disks that are predicted to be faulty in the following 30 days
2	$n_{tp}$	The number of all the disks those truly fail among 30 days after the first predicting day
3	$n_{pr}$	The number of all the disk failures occurring in the k-day observation window
4	$n_{tpr}$	The number of truly faulty disks that are successfully predicted no more than 30 days in advance

### 3.2 Experimental Results

There are two steps in model verification stage. Firstly, we predict offline validation dataset, then select the optimal F-score value and corresponding prediction threshold. The prediction results of the offline validation dataset are shown in the Table 3. By ensembling several good sub-models together, the overall prediction performance can be improved. The best sub-model is Model\_06, with F-score 34.21, while the integrated model's F-score reached 36.36, an increase of 2.15. Secondly, we predict online test dataset for final testing. The prediction precision was 52.42, the Recall was 32.31, and the F-score was 39.98.

**Table 3.** Experimental results offline.

Model	Precision	Recall	F1
Model01	37.6	29.7	33.21
Model02	40.6	27.2	32.58
Model03	47.4	23.4	31.36
Model04	36.4	30.4	33.10
Model05	36.3	31.0	33.45
Model06	35.6	32.9	34.21
ensemble	48.4	29.1	36.36

## 4 Summary

In large-scale data centers, disk is the component with the highest failure rate. Disk failure will seriously affect the stability and reliability of IT infrastructure. Based on the SMART data set of Alibaba Data Center, this paper designs and implements an efficient disk failure prediction system. The training process of the system consists of five parts: feature extraction, preprocessing, model training, model ensemble, and model verification. XGBoost algorithm is applied. After system-level optimization, the F-score achieves 39.98. In the competition jointly held by Alibaba and PAKDD, the effectiveness and versatility of our system was approved.

There are many viable ways of extending this work, such as: Applying transfer learning algorithm to solve the problem of insufficient samples of failed hard disks. Using ranking algorithms to make further improvements. Analyzing disks that are not reported in time or reported wrongly.

**Acknowledgements.** Thanks to Alibaba and PAKDD for organizing the PAKDD2020 Alibaba Intelligent Operation and Maintenance Algorithm Contest, which give us precious training data sets. This competition also gives us the opportunity to communicate with experts. We especially thank Inspur and the leaders for their support.

## References

1. <https://en.wikipedia.org/wiki/S.M.A.R.T>
2. <https://github.com/alibaba-edu/dcbrain/tree/master/diskdata>
3. <https://tianchi.aliyun.com/competition/entrance/231775/information>
4. Hongzhang, Y., Yahui, Y., Yaofeng, T., et al.: Proactive fault tolerance based on “collection-prediction-migration-feedback” mechanism. *J. Comput. Res. Dev.* **57**(2), 306–317 (2020)
5. Sidi, L., Bing, L., Tirthak, P., et al.: Making disk failure predictions SMARTer! In: *Proceedings of the 18th USENIX Conference on File and Storage Technologies (FAST 2020)*, Santa Clara, CA, USA, pp. 151–167 (2020)
6. Yong, X., Kaixin, S., Randolph, Y., et al: Improving service availability of cloud systems by predicting disk error. In: *Proceedings of the 2018 USENIX Annual Technical Conference (USENIX ATC 2018)*, Boston, MA, USA, pp. 481–493 (2018)
7. Yanwen, X., Dan, F., Fang, W., et al: DFPE: explaining predictive models for disk failure prediction. In: *2019 35th Symposium on Mass Storage Systems and Technologies (MSST)*, Santa Clara, CA, USA (2019)
8. Ganguly, S., Consul, A., Khan, A., Bussone, B., Richards, J., Miguel, A.: A practical approach to hard disk failure prediction in cloud platforms: big data model for failure management in datacenters. In: *2016 IEEE Second International Conference on Big Data Computing Service and Applications*, pp. 105–116. IEEE (2016)
9. Ma, A., et al.: RAIDShield: Characterizing, monitoring, and proactively protecting against disk failures. *ACM Trans. Storage* **11**(4), 17:1–17:28 (2015)
10. Nicolas, A., Samuel, J., Guillaume, G., Yohan, P., Eriza, F., Sophie, C.: Predictive models of hard drive failures based on operational data. In: *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 619–625 (2017)
11. Tan, Y., Gu, X.: On predictability of system anomalies in real world. In: *2010 IEEE International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS)*, pp. 133–140. IEEE (2010)
12. Dean, D.J., Nguyen, H., Gu, X.: UBL: unsupervised behavior learning for predicting performance anomalies in virtualized cloud systems. In: *Proceedings of the 9th International Conference on Autonomic Computing*, pp. 191–200. ACM (2012)
13. Wang, Y., Miao, Q., Ma, E.W.M., Tsui, K.L., Pecht, M.G.: Online anomaly detection for hard disk drives based on mahalanobis distance. *IEEE Trans. Reliab.* **62**(1), 136–145 (2013)
14. <http://docs.ceph.com/docs/master/mgr/diskprediction>
15. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.* **2**(2), 121–167 (1998). <https://doi.org/10.1023/A:1009715923555>

16. Chen, T, Guestrin, C: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. ACM (2016)
17. Liaw, A., Wiener, M., et al.: Classification and regression by randomForest. *R News* **2**(3), 18–22 (2002)
18. dos Santos Lima, F.D., Amaral, G.M.R., de Moura Leite, L.G., Gomes, J.P.P., de Castro Machado, J.: Predicting failures in hard drives with LSTM networks. In: Proceedings of the 2017 Brazilian Conference on Intelligent Systems (BRACIS), pp. 222–227. IEEE (2017)
19. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)