SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Mvar-Tef: A Modern Transformer-Based Model For Intrusion Detection Over An Industrial Environment

[1]K. Arun Prasad, [2]Dr.G. Pattabirani, [3]Dr. K. Sundaramoorthy

[1] Research Scholar, Department of Information Technology, Annamalai University. E-mail: arunprasad.it@jerusalemengg.ac.in
[2]Assistant Professor, Department of Information Technology, Annamalai University. E-mail: gpattabirani@gcetj.edu.in
[3]Professor, Department of Information Technology, Jerusalem College of Engineering. E-mail: sundaramoorthy.it@Jerusalemengg.ac.in

Abstract
Determining an unobserved signal as normal or abnormal is a critical task in many real-world applications, known as time series irregularity identification. Some dense learning models, like Transformers, are typically used to study a regular representation of data to distinguish between regular and unusual signals. Though no pure Transformer-based technique exists for temporal sequence anomaly identification, some investigations have recently used Transformers to analyze time series information. This study introduces the Multi-variate Transformer encoding function (MVar-TEF), a novel Transformer model and its successful training approach for improved feature learning. In particular, we introduce two-stage techniques to enhance our model's recognition capability. The first phase is random sample initialization, in which we create a pool of random feature to cover up random portions of the samples. As a result, our model can learn how to depict typical data. Phase two is an exclusive prediction based on the variation approach: To appropriately simulate the exclusive and uncertain elements in the first stage, we suggest a novel refining procedure that provides feedback. This work offers an empirical analysis by determining the usefulness of the prediction procedure, which reliably produces more signals that resemble normality. According to extensive testing on various datasets, MVar-TEF performs considerably better in time series anomaly identification than other modern techniques.

Keywords- anomaly detection, time series, successive stage, network, prediction model

1. Introduction

In many applications, including network congestion analyses, monitoring of electrocardiograms, and identifying abnormalities in sensor information from large-scale industries, time series abnormality recognition is essential for reducing risks and avoiding financial losses [1] – [2]. Due to a shortage of labeled anomalies and extremely nonlinear time relationships, several works have concentrated on unsupervised machine learning approaches to address this challenge [3], including statistical and current DL-based methods. The research [4] suggests using the local outlier factor (LOF) for statistical procedures to give each object an outlier status. The space between each data sample and the cluster center calculates the anomaly score. The research [5] collects the features from a standard dataset to a dense cluster. The research explained that the difference between the two sub-sequences was computed to discover anomalies and that the frequency plot was used to recognize the irregular local layout. Numerous studies suggest deep learning-based techniques that use effective nonlinear modeling. Reconstruction approaches are among the most popular deep learning-based unsupervised learning techniques [6]. Typically, it learns the representation of standard information using a GAN or auto-encoder (AE), then reconstructs a signal similar to standard from any given input. The present research uses CNN, LSTM, and RNN approaches as the backbone of their structure. A reconstruction-based GAN network generates outcomes that can point out the irregular temporal ticks from the reconstructed results. The GAN approach based on LSTM is used in [7], which proposes cycle stability loss for efficient information restoration. Auto-encoders (AE) are integrated with sparsely associated recurrent units and incorporate several AE as a group. To extract temporal properties at several resolutions, [8] proposes a shape-forcing reconstruction loss and offers the recurrent AE that includes several decoders of varied decoding lengths. The work integrates the deep learning model with the statistical method. It uses an extended short-term memory algorithm to analyze and compute the dissimilarity among two contiguous sub-sequences in the anticipated region. The statistical approach identifies irregular sub-sequences. Transformers have been attempted to be used for time series identification, and the results have been impressive [9], allowing the

attention mechanism to determine the temporal relationship. Additionally, we use the Transformer framework in this study to integrate the standard temporal sequence description. We propose an adversarial structure with a pure Transformer to restructure the sensible synthetic temporal sequence information and depict the anomalous locations using the variation between genuine and actual data [10]. If the author uses a Transformer encoding technique to design a GAN, the model will learn the average time series data and produce accurate standard-like signals. On the other hand, Transformers does not utilize a reduced latent space, in contrast to the current auto-encoder structure; instead, it allows the model to discover the trivial result by simply copying and pasting an input to the output for redesigning [11]. The investigators devise a novel training strategy to teach the Transformer-based generator about sharing standard time series data. The authors use the masking technique to solve the copy-paste problem [12] – [13]. The second issue raised by this method is identifying the appropriate location to an input signal to identify anomalies efficiently. The optimal spots are irregular locations in the input signal to produce the standard output [14] – [15]. As we have no idea where the unusual components are ahead of time, it isn't easy to mask the unusual areas selectively.

This research suggests a unique Multi-variate Transformer encoding function (MVar-TEF) that employs a Transformer to build the stage-by-stage predictor model. Furthermore, we recommend a novel two-stage technique to efficiently understand the standard feature dataset. We first use random data prediction in the initial stage to train the Transformer-based MVar for feature learning of the standard data series. In Stage 1, the model efficiently discovers standard data sharing as the arbitrarily input bits are filled in. Nevertheless, our model cannot thoroughly analyze every aspect of the input signal in Stage 2, and this randomness poses a significant challenge for recognizing anomalies. As a result, the exclusive portions of Stage 1 is designed to resolve this issue. Furthermore, the areas with substantial features and probably anomalous places with much uncertainty is employed to determine the optimal feature spots. We do this by calculating entropy using the transformer block's attention maps. Stage 2 prediction technique offers feedback for enhanced learning of representations, ultimately raising anomaly identification performance. We demonstrate empirically that the suggested stage 2 prediction is necessary for MVar-TEF to tackle the anomaly identification challenge effectively. The following section describes our principal contributions:

1) We present MVar-TEF, a simple yet powerful Transformer-based network design for unsupervised data sequence anomaly recognition. Additionally, we outline our system's preliminary processing and encoding techniques to handle time series signals efficiently.

2) We provide a novel 2-stage prediction technique for encoding the standard time series data pattern. Our approach offers variational-based input to uncertain regions with the help of a recently developed multi-variational-based transformer. Using the comprehensive ablations, we experimentally confirm that our 2-stage prediction model effectively and robustly embeds the graphical representation of standard time series information into our model.

3) The stage-by-stage time-series anomaly identification datasets are proposed where MVar-TEF obtains new findings with notable improvements.

The work is drafted in the following manner: section 2 gives a deeper analysis of various ideas provided by the researchers. The proposed MVar-TEF methodology is explained in section 3. The numerical results are provided with graphical outcomes in section 4, and the conclusion is summarized in section 5.

2. Related works

This section starts by providing some background information on anomalies. Several efforts have been attempted to characterize abnormal data where statistical outliers are explained based on their nature. The data point at time steps that exhibit unusual patterns that differ noticeably from earlier time steps can be characterized in this context as the anomaly in the time-series data [16]. Using earlier literary works, we classify many anomalies associated with time-stamped data in the following ways. An outlier is a finding that differs significantly from another to another; this leads to questions like whether it was generated using a different mechanism [17]. The most popular methods for finding abnormalities are statistical abnormal identification approaches. The most popular and straightforward unsupervised global anomaly detection technique for point deviations is k-NN. The distance is applied with a distance-based approach to determine the anomaly score [18]. This method relies on the numerical value of k, is computationally costly, and could only succeed if there are enough neighbors for the average data points. The local outlier factor (LOF) technique was developed in the research [19], and it is a well-known unsupervised model for local density-based identification of abnormality.

Calculating the distances to every other instance establishes the k-nearest-neighbors set in LOF for every instance. This algorithm's fundamental premise is that the data instances' neighbors are dispersed spherically.

Nonetheless, spherical density estimation is no longer appropriate in specific application scenarios where regular values are arranged linearly associated [20]. The outlier is an improved LOF version presented by [21]; it enhances the considered linear framework. One limitation of this technique is that it may inaccurately estimate outlier scores in certain situations where clusters with varying densities are near. Low-density cluster boundaries are local outliers compared to high-density clusters in these situations. This flaw is further fixed in the algorithm Influenced Outlierness (INFLO) [22]. There are variations in the anomaly recognition methods as they are based on the semi-supervised and unsupervised SVMs. Amer et al. introduced an unsupervised variation of one class SVM. This technique doesn't demand prior training information, as it is based on [23]. It tries to determine a decision boundary that maximizes the distance between the starting point and the locations. This method can also identify irregular daily activities like walking, sitting, and sleeping. The author in [24] suggested another method for detecting time series anomalies that depend on one class support vector machine.

This method finds anomalous states by defining six meta-features on real univariate or multi-dimensional temporal series and applying SVM to the meta-representation-based data space. When there are no labels, SVM is typically responsive to the outliers. It is also applied as a novel method for identification. The author suggested a novel technique for identifying outliers using Support Vector Data Description (SVDD). Various anomaly identification algorithms are available that have been developed to identify irregularities in streaming temporal series data, allowing for the incorporation of time series properties. Robust Anomaly Detection (RAD) [25], a feature of Netflix's anomaly detection system, was publicly available in 2015. Robust Principle Component Analysis (RPCA) is the foundation of the function's anomaly detection process. A disk-aware technique is described in [26] to identify abnormal time series in a multiple data source. The statistical autoregressive-moving-average (ARMA) method and its variants are frequently utilized for time sequence recognition and abnormality detection. Based on the ARIMA model, the author in [27] introduced an anomaly recognition method for congestion management in remote sensor networks. Researchers suggested that the short-step exponential weighted average approach enhances network traffic identification and decision-making. The author in [28] presented a method for prior recognition of Denial-of-Service (DoS) intrusion in a similar domain. The actual network traffic is compared with ARIMA-produced patterns to detect anomalies.

The research engaged LSTM to categorize a time series as regular or irregular. Using a healthcare information set, they implemented their method. They showed that long- and short-term memory trained on time series performed better than MLP learned on hand-engineered attributes. CNN was utilized in the research [29] to classify multi-variate time data. They developed Multi-Channel Deep CNN (MC-DCNN), in which each channel learns the features independently by utilizing just one dimension of multiple temporal stamped information as the data source. A layer of MLP is used to classify this. According to experimental findings, MC-DCNN performs better than competitive baseline techniques, such as dynamic time wrapping and K-nearest neighbor (based on Euclidean distance). A sequence is classified as usual, and a subsequence can be classified as abnormal using any of the abovementioned DL-based time series abnormality identification approaches. A particular kind of neural network called an auto-encoder (AE) is learned to replicate its input. AEs are typically utilized to reduce dimensionality, which helps for categorization and visualization applications. Its effective unsupervised data encoding makes it an attractive option for anomaly and novelty recognition issues. The author in [30] introduced a new approach incorporating an AE to detect irregularity. Their method applies density-based clustering to the reduced information after receiving the error threshold and compressed information from DAE (deep auto-encoder) to obtain new groups with minimal density. The research uses a Deep auto-encoder to identify abnormal signals in large-scale accounting information in the intrusion recognition region.'

3. Methodology

This section initially formulates the problem and then the anticipated framework of MVar-TEF, which overcomes the shortcomings of the existing technique. Fig 1 shows an arrangement of heterogeneous characteristics.

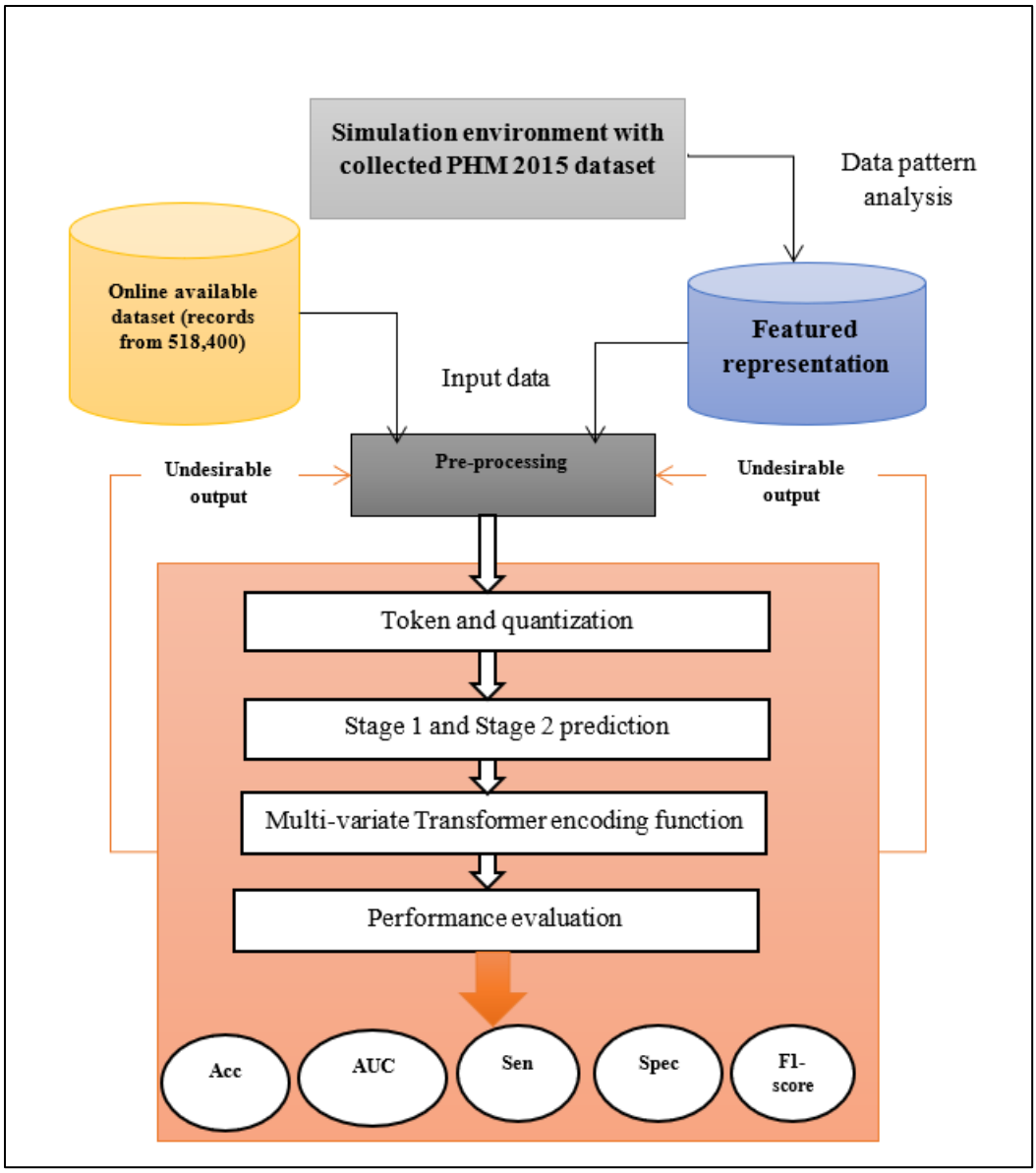

Fig 1 Flow diagram of the anticipated model

3.1. Dataset description

One huge-scale real-world data source is multi-variate time series information obtained from an automobile engine valve. The complete dataset comprises 518,400 time stamps (also known as temporal recordings) for a set of features captured between the second of December 2019 and the seventh of December 2019. Next, the study describes the feature set divided into two signals below. Despite the state of operation, sensor signals are captured every second and frequently display temporal continuity. Operation cycle signals are less likely to pursue temporal sequence because they are logged-in measures. Unlike sensor signals, these operation cycle signals occur randomly because they are solely event-driven. Because the operation is idle during both the planned short outage and the abnormal (i.e., unplanned) extended outage, we observe that the signals from sensors have equal spacing. Still, signals are irregularly placed in time.

3.2. Problem statement

Our issue is described in automatically recognizing irregular events in the industrial assembly line with the diverse temporal sequence manufacturing data source. Since the user needs to know whether an anomalous event has happened within a reasonable period, it would be useless to do point-wise detection of anomalies. Therefore, our goal is to perform range-wise identification of irregular occurrences by identifying a probable strange location, a time-stamp close to the actual abnormal event and exhibiting an abnormal pattern. Range-wise abnormality identification, given a varied multi-dimensional time series data source, aims to identify an

irregular position t such that, for a given tolerance $\delta > 0$, there exists a strange occurrence in $[t - \delta, t + \delta]$. It should be noted that the operator's criteria predict how the tolerance δ should be established in advance.''

3.3. Pre-processing

An appropriate preliminary processing phase that turns the input data signal into distinct tokens is required to handle an input data signal for a Multi-variate Transformer encoding function (MVar-TEF). To achieve this, we use the min-max scaling to stabilize every series time input data source X in the range of $(-1, 1)$. Next, we take matching integers as a token and quantize the normalized actual score within a particular range from 0 to K, where the hyper-parameter managing the quantization resolution is represented as K. Assume $\tilde{X} \in R^{T*n}$ as a signal already processed. For every experiment, we select $K = 200$ so that pre-processed signal $\tilde{X}$ relatively resembles the input value of X. We incorporate the token into the total K tokens (quantization levels) for training and testing. The expanding and compression are used successively to pre-process the input signal X so that it becomes $\tilde{X}$.

3.4. Prediction flow

Using the proposed heterogeneous multi-dimensional time-stamped data source, we present a new detection method called MVar-TEF based on a data-driven methodology and non-obvious empirical discoveries to achieve anomaly identification. The general structure of the recommended MVar-TEF structure is shown in Fig 1, where two distinct unsupervised learning designs are trained based on two different signals (i.e., sensor and operation cycle signals) before being applied to signal recognition assignments.

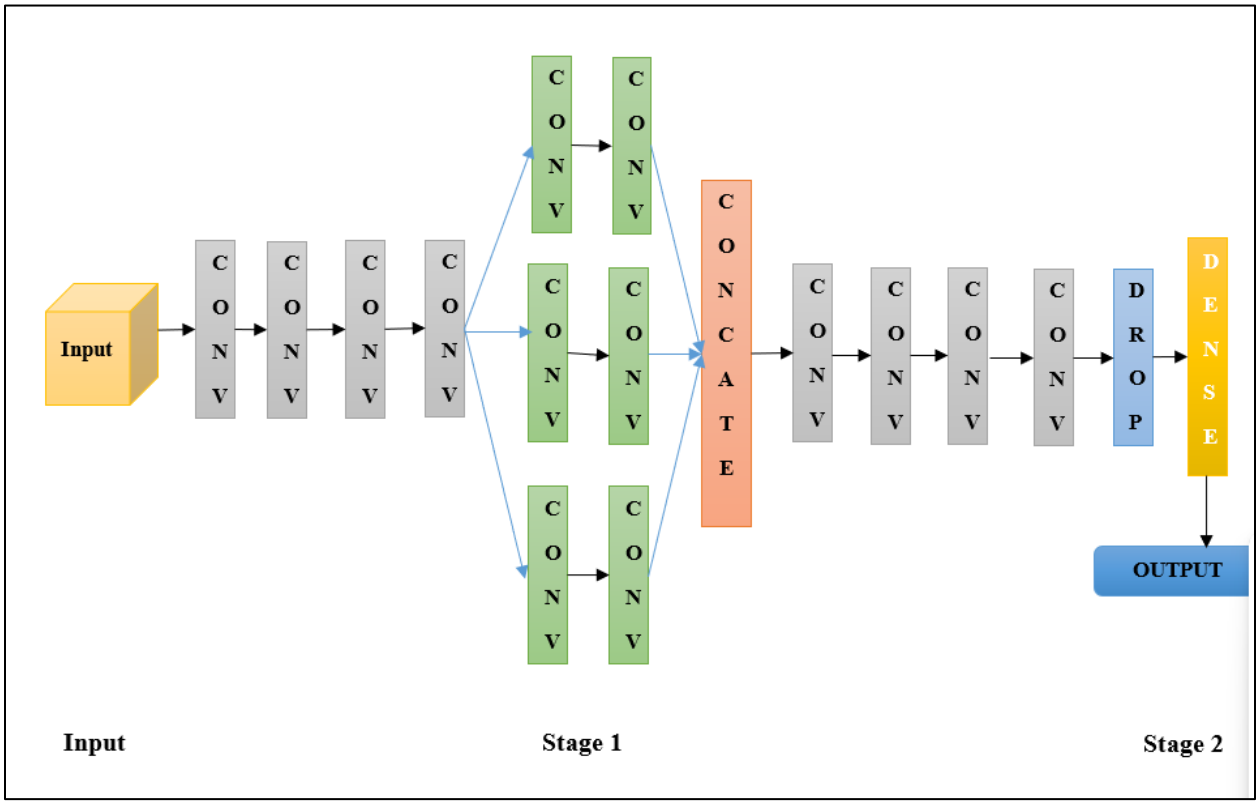

Fig 2: MVar-TEF architecture

Specifically, we first employ the initial strategy that uses multi-dimensional operation cycle signals, and the successive strategy uses multi-dimensional transducer signals for the time series dynamics; all approaches are learned without using labels in an unverified manner. This can identify a set of highly probable abnormal candidates, designated as (anomaly), where T_0 is the collection of time stamps at which operation cycle signals occur, and the anomaly score $a_t^{(o)}$ is more significant than the pre-determined threshold $\tau_1 > 0$. In contrast, we exclude extremely abnormal points among the candidates to identify the most probable anomalous points. To be more explicit, we eliminate the collection of spots (T_s), represented as $t_i \in (T_o, abnormal)$, and η is greater than 0, a range parameter that needs to be experimentally established. The highest value of the

resultant inconsistency values $a_t^{(s)}$, $\forall_t$ belongs to $[t_i - \eta, t_i + \eta]$, is less than one more threshold τ_2 greater than 0. In other words, we have finally identified the cluster of irregularities or $\hat{T}_{abnormal}$ as shown below:

$$\hat{T}_{abnormal} = T_{0,abnormal}/T_s \quad (1)$$

$$T_s = \{t\_i | forall\ t_i \in T_{0,abnormal} : \max_{t \in [t_i - \acute{\eta}, t_i + \acute{\eta}]} a_t^{(s)} < \tau_2 \quad (2)$$

3.4. Implementing the prediction process

For the multi-variate time series problem, this research suggests a framework called MVar-TEF in conjunction with a source tensor transformation technique. The suggested MVar-TEF input tensor system wraps the sliced time series information into suitable tensor visualization to handle multi-dimensional data. We then use sliding windows to obtain the feature attributes for time-stamped information and apply filters to take advantage of local connections between variables. We use two models to forecast the beginning and ending times sequentially because the work needs us to estimate the beginning and ending times for every failure. The MVar-TEF architecture is shown in Fig 1. It consists of four divisions: univariate convolution, ultimately linked stages, multi-variate convolution, and input tensor transformation. Below is a list of the four stages of introductions.

a) Stage 1- data transformation

Time series data have distinct structures. Alternatively, data can be represented as $h \times w \times 3$ 2D-tensor. As such, the convolutional layer scans the data using 2D filters. We suggest an image-like tensor method to represent multi-dimensional time series data with CNN's remarkable achievement with data classification. This will enable MVar-TEF to learn about lag-feature properties and interaction between variables. Since time-series information is a data point collection classified in time sequence, one significant characteristic is its natural temporal ordering. As an outcome, it is optimal to consider the relationships among nearby data points. We consequently apply sliding windows to time series information to handle data and categorize batches. We use the suggested input tensor conversion technique to convert the time-stamped dataset into tensors, where the sliding window's dimension corresponds to the tensors' depth, provided the signals inside a window. For instance, a sliding window with a length of 4 is shown in Fig 2, together with four sensor data. We can create $a\ 1 \times 1 \times 4$ 3D tensor for every transducer signal in the sliding window by using four different values from the time series. Subsequently, we combine all the $1 * 1 * 4$ tensors to create a tensor with dimension $2 * 2 * 4$, whose depth matches the sliding window interval.

b) Uni-variate convolution stage

After the input tensor conversion, we subject the altered tensors to linear convolution. Separating the characteristics from every sensor is the aim of the univariate convolution phase. The length of the input tensor determines the filter size used in the univariate convolution step, which scans the tensor using a filter with a plane dimension of 1×1. Initially, Uni-variate employed the 1×1 convolution in network design to deepen the network. This work adds non-linearity using 1×1 convolution to improve the local model's abstraction capacity. An illustration of univariate convolution is shown in Fig 2, where the input tensors are the outcomes obtained. Subsequently, the suggested technique employs a $1 * 1$ filter and conducts a convolution process on the input. Afterward, the outcomes are arranged on the output feature maps. We will assume that there are five filters in this instance, which explains why, following the univariate convolution stage, the output tensor has the dimensions $2 \times 2 \times 5$. Next to the 1×1 convolution processes are finished, the outputs are subjected to the rectified linear unit (ReLU) activation function, which yields nonlinear results. The output tensor's dimensions match the source tensor in the univariate convolution phase.

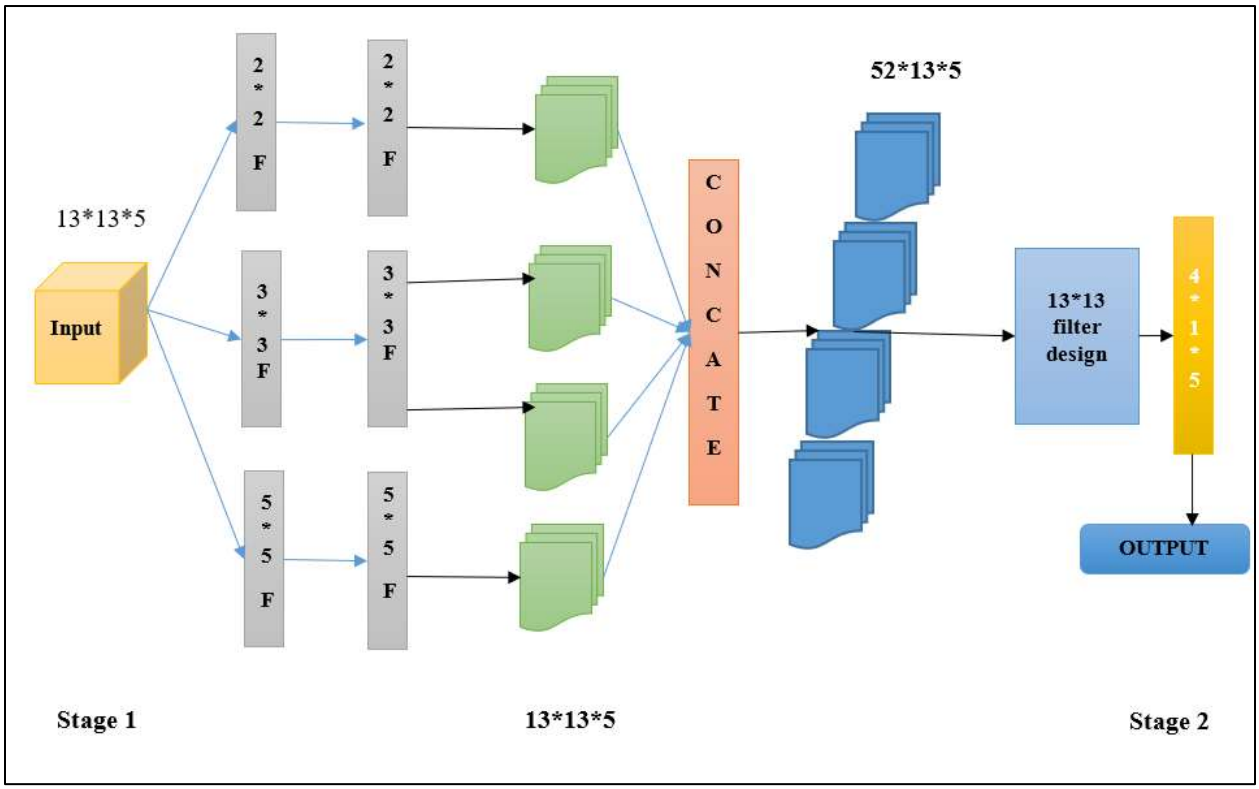

Fig 3: Stage-by-stage prediction layer

c) Multi-variate convolution stage

Based on the baseline Inception, we create a 4-way convolution, as shown in Fig 3, to determine the best local structure. Various filter scales extract distinct local information from image inputs. While small-scale filters identify specific characteristics, large-scale filters identify the symmetric attribute. Various filter scales provide a means of extracting aspects of the inter-sensor interaction from time series data. We employ three filter scales in the suggested architecture: $5 \times 5, 3 \times 3, and\ 2 \times 2$. Two layers make up each convolutional layer method. The initial convolutional layer is considered a means of determining the sensor's local correlation since it groups the nearby sensors. The subsequent convolutional layer takes the statistical data among groups. Since we apply identical padding for every convolutional layer, the tensor plane dimension and the resultant tensor of the univariate convolution phase are the same. Furthermore, we use the ReLU activation function immediately following every convolutional layer. In addition to the three filter dimensions, we explicitly employ the univariate convolution phase's output as an additional convolution method. After the procedure, we combine every tensor diverse from the Inception to load. To preserve the distinctive characteristics of each set of convolutional layers, we combine them. Subsequently, we implement an identical filter size as the input to the output obtained from concatenating the inception units. Therefore, the filter reduces every tensor from several methods into a single vector. The output tensor at this phase is $4 \times 1 \times 5$, as Fig 3 illustrates.

d) Fully Connected Layer (FCL)

Lastly, we implement FCL to the multi-variate convolution stage's output tensors. In addition, to stop the proposed MVar-TEF technique from the issue of over-fitting, we include a dropout layer preceding the last fully linked layer. The softmax function that converts the forecasts into possibilities constitutes the ultimate result. Hence, the likelihood of an abnormal event equals the MVar-TEF output. The first equation displays the cross-entropy function where the actual distribution is represented as $p^{(i)}$, and the evaluated distribution is $q(i)$.

$$H(p,q) = -\sum_{i} p(i) \log q(i) \quad (3)$$

This work uses the cross-entropy function as the loss function. The adaptive moment estimation (Adam) optimization technique with a 10^{-4} initial learning rate trains our model, which integrates momentum and Adagrad components. The developed MVar-TEF technique is employed to reduce the loss function. We experimented with two datasets to see if the suggested model worked with time series data. PHM 2015 challenge is the dataset that predicts the beginning time, completion time, and defect kind. On the other hand,

the data set is intended to be used in different environments. We employ numerous algorithms in the application to forecast the start and finish times of individual faults in a given plant in an ordered way. Furthermore, this study examines modeling the problem using binary classification, whereas an algorithm predicts each defect. There are, therefore, a maximum of 330 variants. 270 models have been deployed in the PHM 2015 challenge; however, some plants do not feature all five defect types.

e) Prediction flow

Fig 3 shows the flow of preliminary processing. Since the signal should be sampled every 15 minutes, but certain logs do not capture events due to time delays or other issues, effective pre-processing of the data is needed. Time synchronization is thus applied to data collection to address this issue. We use feature engineering and information pre-processing to add more characteristics to the framework. As a result, the prediction model incorporates extra variables like the month, hour, weekday, and aspects of every sensor signal. As previously indicated, the PHM data is of a time series type. To ensure that the modified vectors of features have the qualities of time-series information, we process the data according to the methods suggested by some existing works [25]. In the investigations, the hyper-parameter z is declared 4 for each signal $S(t)$ for a given time t. To account for signal variations before and following time t, we include lag features in $S(t - z)$ and $S(t + z)$. This results in the signals in $(2 * z) + 1$ time steps being regarded as a sliding window. Furthermore, we convert the detection issue to an issue of binary classification, suggesting that the developed approach needs to create a prediction model for every kind of failure. We remove data samples if a factory has fewer than 199 individual defects since it is essential to note that, in some instances, the specific fault count needs to be higher to train a prediction model. Next, we forecast the beginning and finish times using several models. We have an imbalanced data set issue when predicting the beginning time because most samples are regular, and only a tiny percentage is abnormal. Nevertheless, additional data pre-processing is necessary because we are interested in strange cases. The present study uses the under-sampling method to address imbalanced situations as empirical evidence demonstrated that the under-sampling algorithm may perform superior to other options.

Furthermore, the prediction algorithm in this work is based on a probabilistic approach, which produces probabilistic findings. The transformation process requires a threshold value to convert the probabilistic results into binary categorization outcomes. This research suggests a cut-point strategy to identify the threshold value for deciding if a data model is irregular. The simulated cut-point is the modified threshold score that yields the most significant value of the validation set after increasing by 0.01 and adjusting the threshold value from 0.5 to 1. The suggested simulating cut-point is a validation technique, as the procedure above is used on the validation set. The anomalous possibilities of data samples accessible with the MVar-TEF framework have been estimated. We evaluate six successive data samples, as indicated in Fig 3, to determine whether the highest probability of the information in those samples is greater than the threshold value. The time-stamp of the provided sample is the anticipated start time for the designated fault if it occurs in the sequence. For illustration purposes, assume that the threshold value is 0.7 and that stage 2, shown in Fig 3, meets the constraint above. In this case, time stamps t_3 & t_7 correspond to two anticipated instances of the given fault.

The start time in a time series data must come before the final time; the ending time forecast should be predicated on the beginning time. We predict the ending time after we have finished predicting the beginning time. Given that every time stamp following the beginning time may be probable candidates, it makes intuitive sense that there should be many comparable end times. To reduce the potential candidates, we examine the fault occurrence interval by defining an interval by determining the variation between the beginning and finish times and determining a limit that surpasses 95 percent of fault durations. Logs are collected from $_{start-train} + b\ to$ $_{start-test} + b$ as a training set by considering it as the starting time and its related boundary is obtained in the earlier process; the final test is the most likely to occur in the data sample within the interval between the start-up test and the start-up $test + b$. The beginning and end times together are considered a forecast pair.

4. Numerical results and discussion

The labels in the dataset and the methods used for data pre-processing are initially discussed in this section. Next, we present five standard anomaly recognition techniques and several performance measures. We also demonstrate experimental configurations. Our primary contributions are our extensive experimental evaluations, which confirm our MVar-TEF framework's superiority over single-phase benchmark techniques. Since the breakdown events brought on by the assembly line are not present in the dataset, we have gathered the collection of alarm points and utilized them as ground truth in our studies. Many industrial plants are equipped with an essential part for plant control and effective and safe operation, known as an alarm model.

This part informs an operator of unusual situations or process faults that could be anomaly candidates. Alarm logs in industrial systems combined with timely operator involvement can be beneficial in identifying potential malicious abnormalities (i.e., unusual events to perhaps catastrophic levels). Within the data set, an alarm location is a binary format that records the time stamp for an alarm event. In other words, an alert occurs if the values at a time stamp in a consecutive time series of binary signals are 1.

4.1. Experimental setup

This section initially explains our network model settings to provide MVar-TEF architecture. Otherwise, we use the deep algorithms utilized in stages for MVar-TEF as the standard parameters for our experiments because the discussion of unsupervised learning models produces the best results. The MVar-TEF framework is trained with an initial learning rate of 10 to 3 and a batch length 32 with the Adam optimizer algorithm. The deep techniques employ loss functions based on the mean absolute error (MAE) and mean square error (MSE). The length of both the layers is declared as 6. The first layer of the encoder adopts a $(1 - norm)$ regularizer; the conventional ReLU and tanh activation functions are employed one after another in every layer among the input and output layers. Each layer is applied with dropout with a probability of 0.2; the window length ($w = 180$) and the number of steps ($\gamma = 60$). Furthermore, we have chosen a tolerance threshold of $600s$ for δ in our trials because it may be sufficient to set off an alarm without a significant delay. Since an operation cycle typically lasts between 14 and 15 seconds, the range setting parameter η utilized in stage 2 of MVar-TEF and its value is declared as 14 (seconds). The operator's requirements and the surroundings of the assembly process can determine the appropriate values for the parameters δ and η. The training set consists of records measured between the second of December 2019 and the third of December 2019, and the test set comprises the other records, which were separated from our time series dataset in sequential order. The MVar-TEF model is trained, and the test set is employed to perform the signal-specific identification activities using the threshold setting.

Table 1: Performance evaluation in 10th iterations

Methods	**One-stage techniques**					**Two-stage techniques**
Sample	**DAE**	**GM**	**E-D**	**LSTM**	**VAE**	**MVar-TEF**
P	60	67	58	50	70	**91**
R	85	56	78	25	75	**94**
F1-score	70	61	67	30	50	**97**

Table 2: Performance evaluation in 20th iterations

Methods	**One-stage techniques**					**Two-stage techniques**
Sample	**DAE**	**GM**	**E-D**	**LSTM**	**VAE**	**MVar-TEF**
P	46	40	40	57	47	**95**
R	71	35	35	38	71	**92**
F1-score	56	60	60	46	56	**94**

Table 3: Performance evaluation in 30th iterations

Methods	**One-stage techniques**					**Two-stage techniques**
Sample	**DAE**	**GM**	**E-D**	**LSTM**	**VAE**	**MVar-TEF**

P	45	70	50	66	58	**96**
R	40	81	45	49	75	**93**
F1-score	30	71	70	55	66	**95**

Table 4: Accuracy of detection rate

DR	**DAE**	**GM**	**E-D**	**LSTM**	**VAE**	**MVar-TEF**
Testing time	50	75	55	66.3	58.8	**97.5**
Training time	45	82	45.3	49.2	75.5	**98**

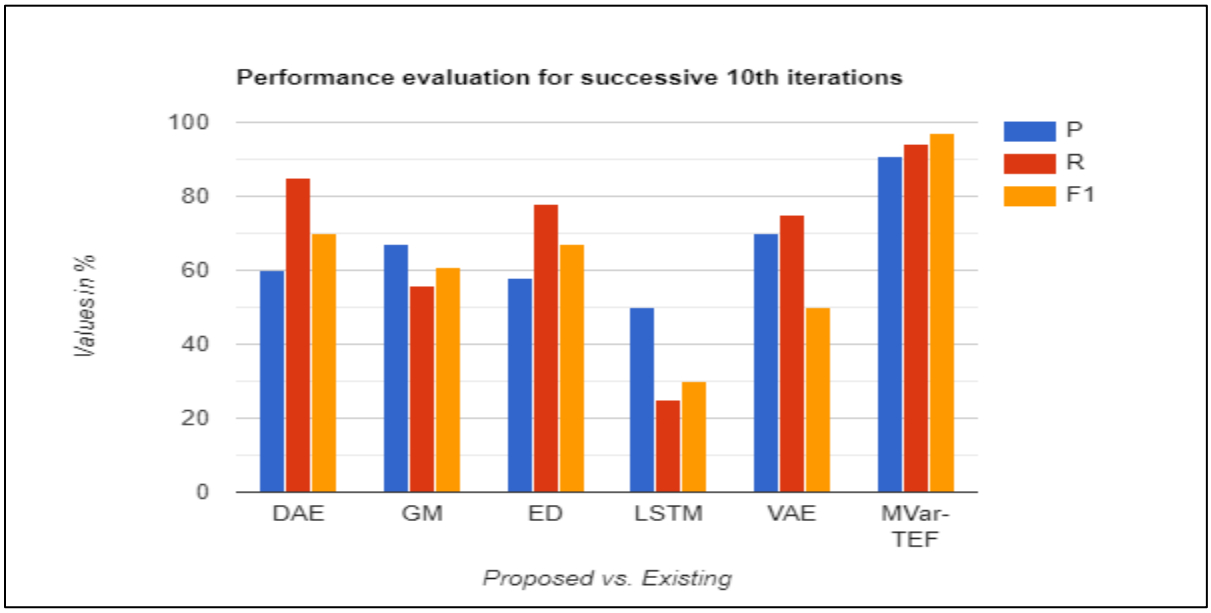

Fig 4: Performance evaluation in 10th iterations

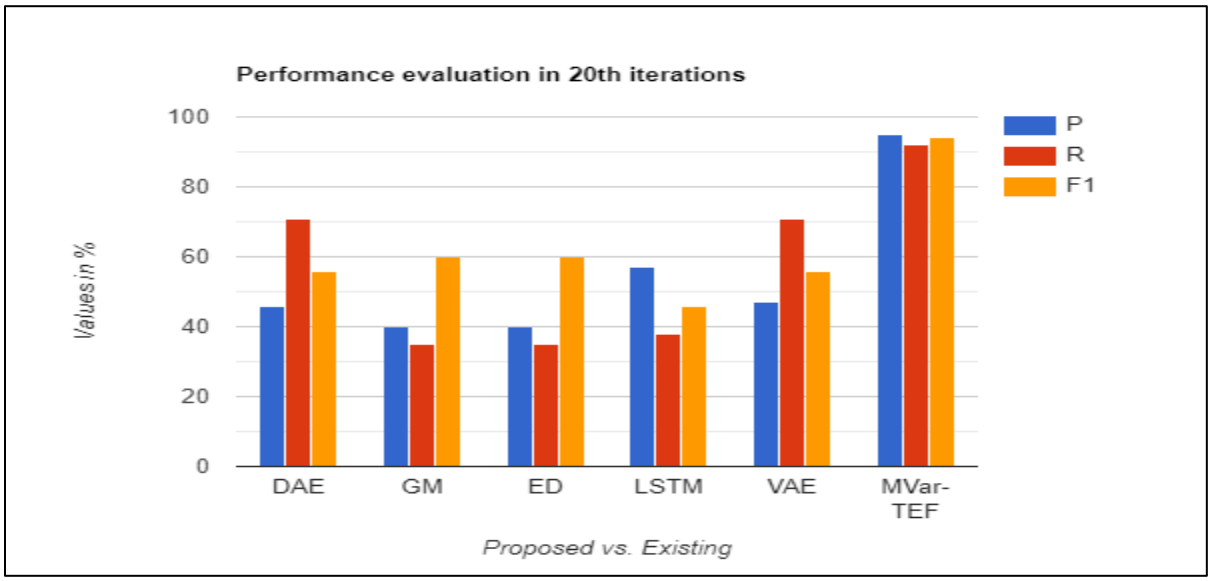

Fig 5: Performance evaluation in 20th iterations

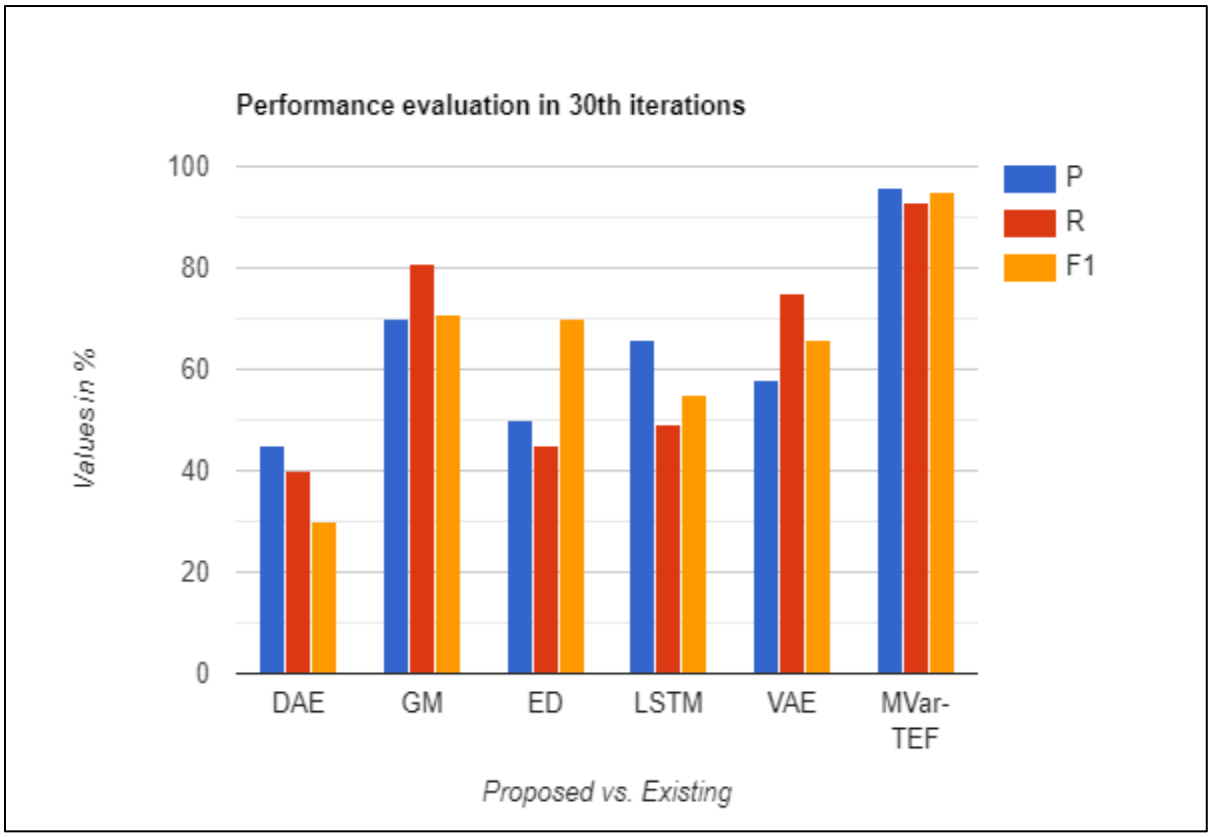

Fig 6: Performance evaluation in 30th iterations

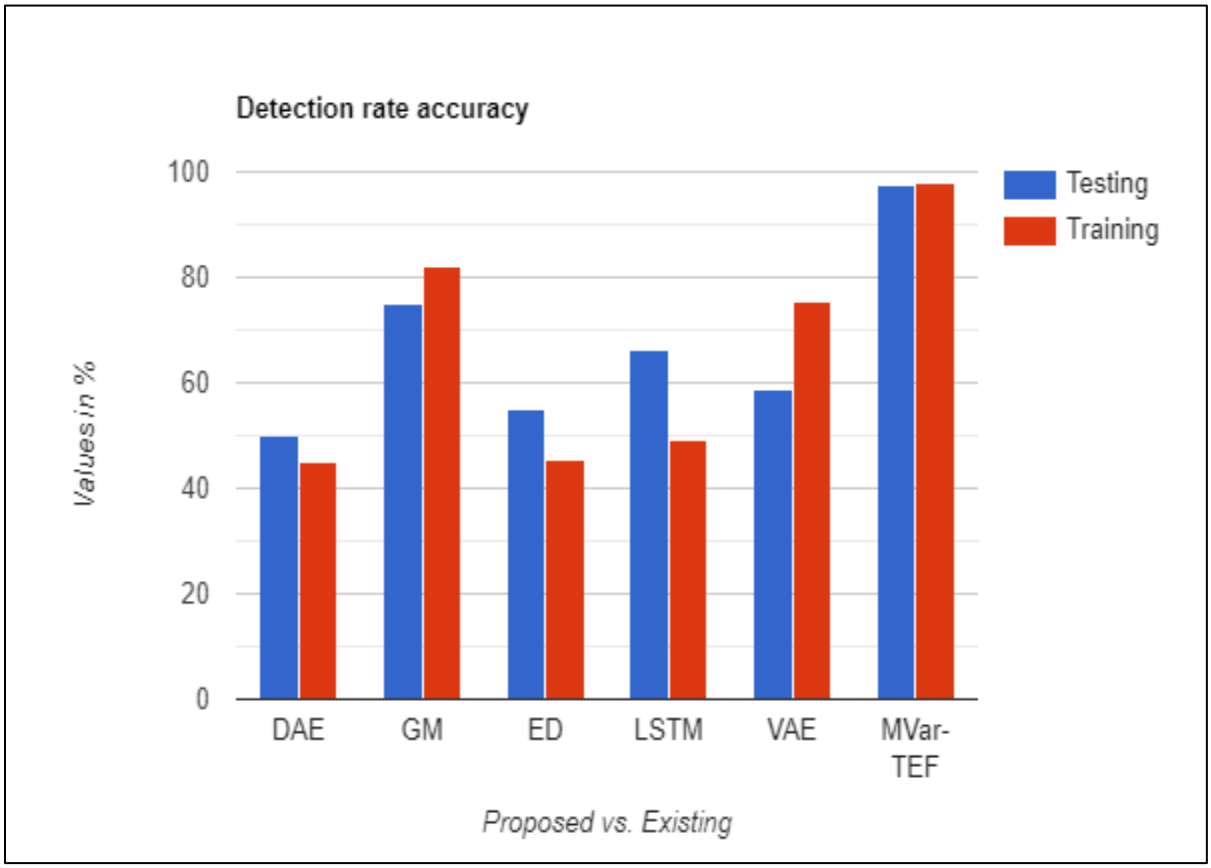

Fig 7: Detection rate accuracy

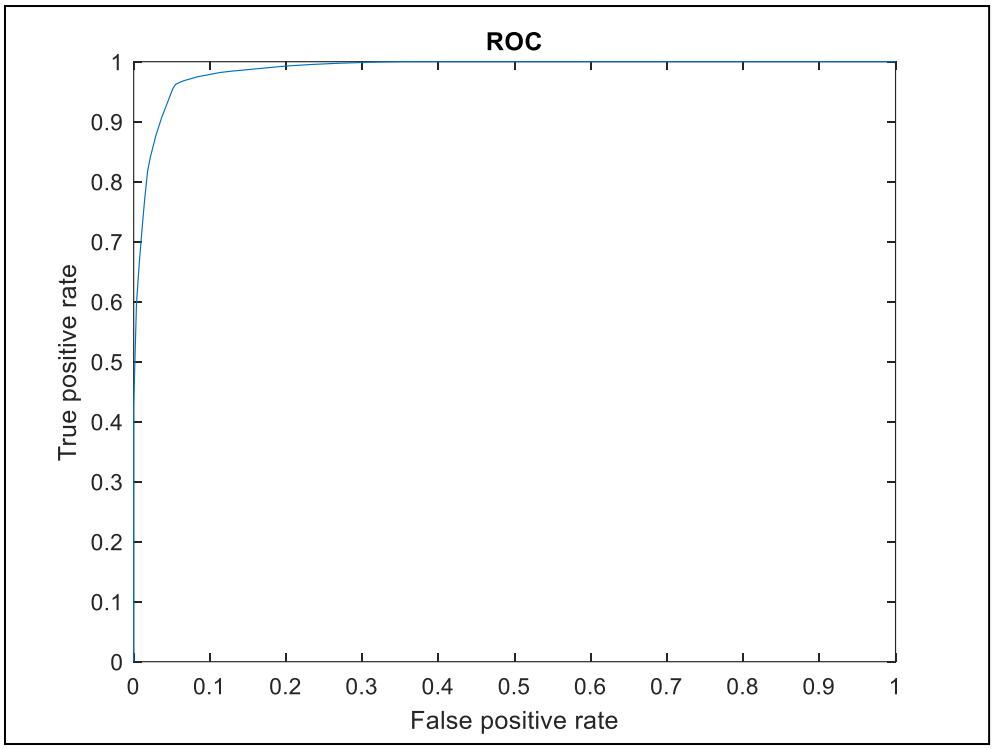

Fig 8: ROC computation

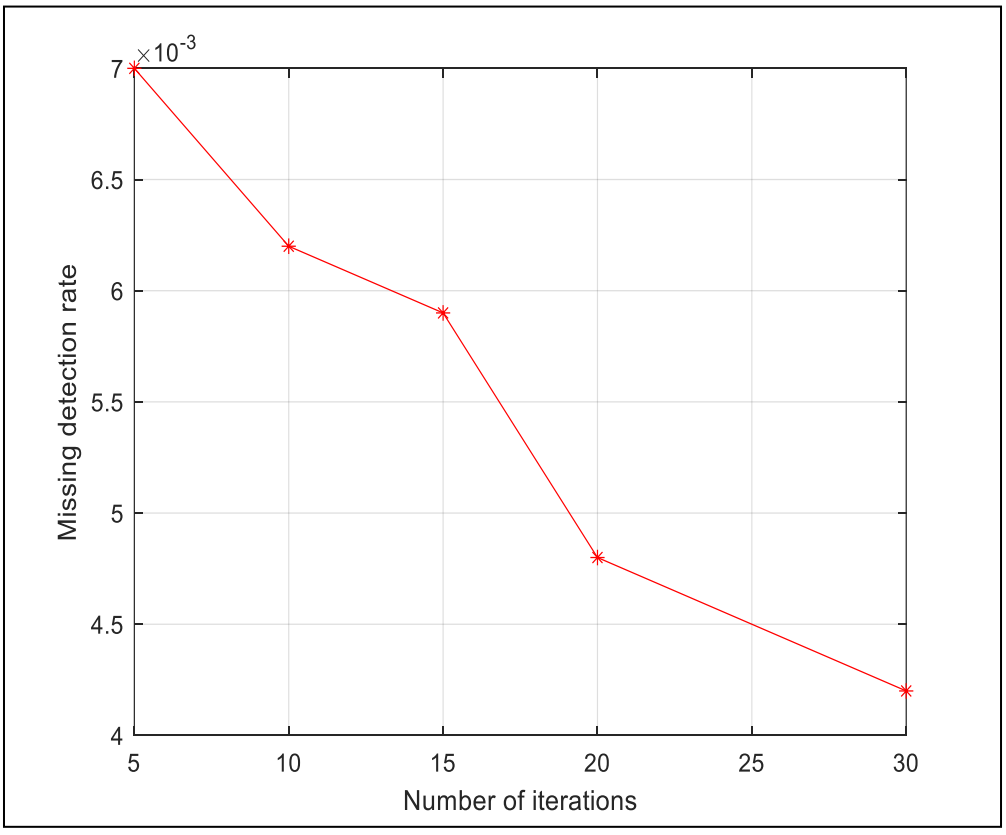

Fig 9: Detection rate

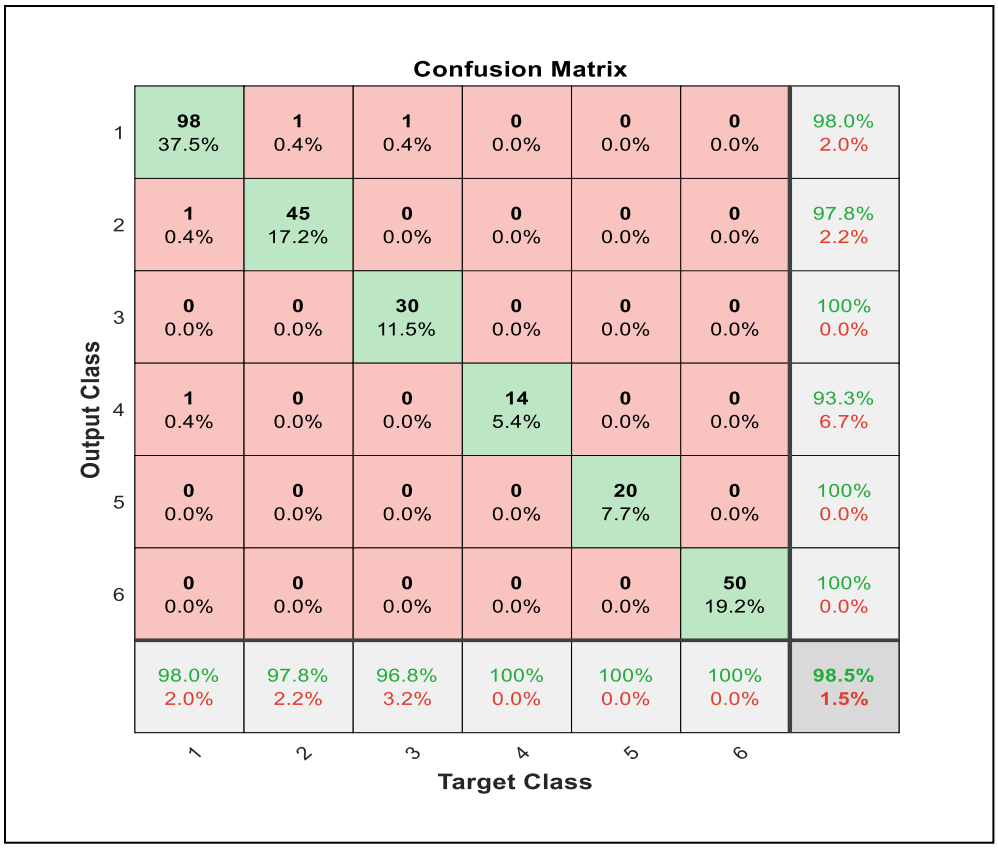

Fig 10: Confusion matrix

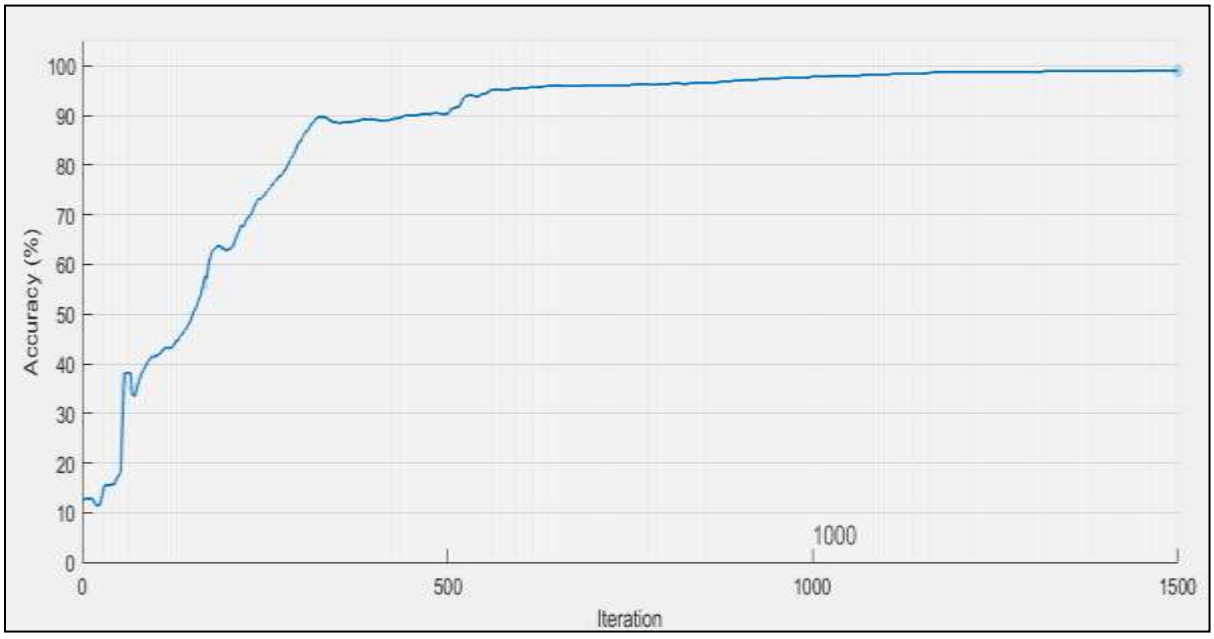

Fig 11: Accuracy based on 1500 iterations

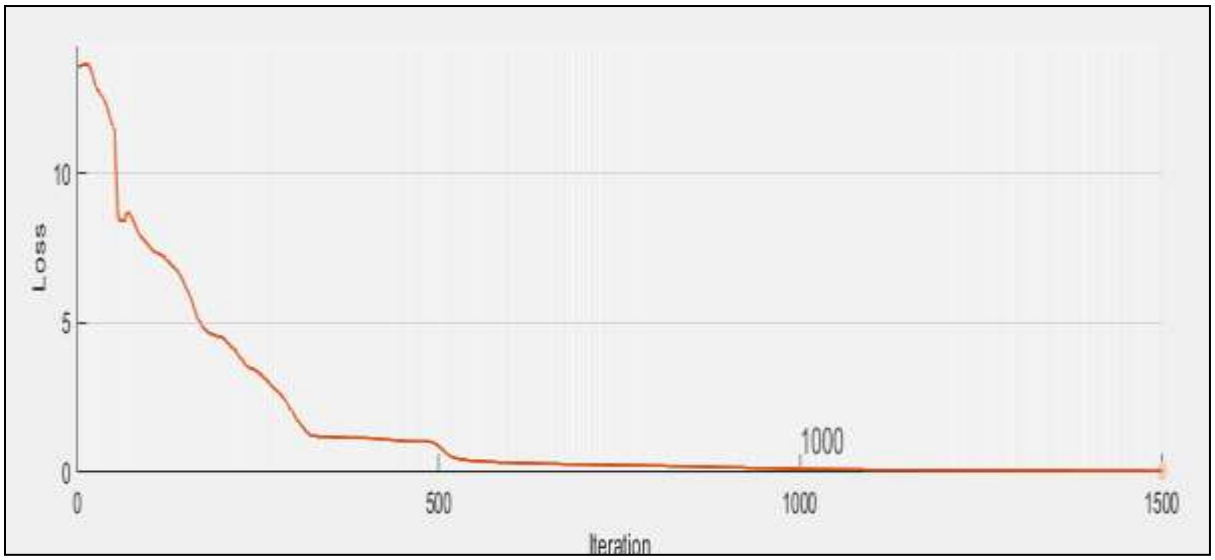

Fig 12: Loss based on 1500 iterations

4.2. Performance evaluation

Since operators typically are not concerned with point-wise metrics, we take a different approach to the widely used precision and recall to properly evaluate the range-wise alarm identification method's performance. Accuracy indicates how close an identified abnormal data can be to detecting the alarm score within the specified tolerance δ. In contrast, recall refers to identifying alarm spots by the irregular locations within the tolerance level δ. F1-score is also employed to determine the best possible achievement of the proposed technique for the given dataset. The threshold value is predicted by evaluating the precision and recall harmonic mean. The proposed MVar-TEF architecture is evaluated using two thresholds to optimize the f1-score through an extended search. This process helps to understand the applied dataset under certain fundamental boundaries of unsupervised alarm recognition, as in Fig 4 to Fig 12.

4.3. Comparison with benchmark approaches

We outline five standard anomaly identification techniques. They serve as state-of-the-art single-phase standard comparison techniques and foundation frameworks in our MVar-TEF system.

- Auto-encoder is identical to our MVar-TEF architecture's stage 1 design.
- LSTM is identical to our MVar-TEF framework's stage 2 design.
- LSTM auto-encoder combines LSTM layers with a connected variational AE to recognize irregular signals.
- GANs and trained auto-encoders design unsupervised anomaly detection.
- The Gaussian mixture model is designed by merging the auto-encoder to determine abnormalities.

4.4. Result analysis

The four main research topics that our research investigation aims to address are presented in this part.

1) To what extent does the choice of two thresholds, τ_1 *and* τ_2, affect the MVar-TEF framework?
2) To what extent does MVar-TEF performance outperform single-phase benchmark techniques regarding recognition accuracy?
3) In MVar-TEF, which combination of core models produces the most outstanding results?
4) Compared to other datasets with fewer features, how reliable is the MVar-TEF structure?

4.5. Sensitivity analysis

To examine the sensitivity of the parameters, we present the performance of the suggested MVar-TEF framework about the F1-score based on threshold values of τ_1 *and* τ_2. It can be noticed that a slight fluctuation in τ_1 results in a non-negligible variation in the F1-score, indicating that the performance is more sensitive to τ_1 values. The best F1 score is also discovered at $(\tau_1, \tau_2) = (2.550, 0.080)$, employed in the following experiments.

4.6. Comparison with benchmark approaches

To our knowledge, anomaly detection using diverse signals in practical manufacturing information has to be examined. Because of this, we show that our suggested framework uses stages 1 and 2 of our τ_1 *and* τ_2

structure, respectively, which is superior to single-phase benchmark approaches. The following four scenarios are considered as workable single-phase methods.

Lemma_1: The operation cycle signals are employed by every procedure.

Lemma_2: For each time-stamp, only sensor signals $x_t^{(s)}$ are employed by various techniques.

Lemma_3: If operation cycle signals are not recorded, we initially replace missing scores with null (zeroes). The vector concatenation of the two signals is then used as the input for each method's time stamp.

Lemma_4: All techniques employ the vector concatenation of two signals (sensor and operation cycle) as input only in cases where both signals are recorded. Tables 1 to 4 compare the MVar-TEF architecture's performance to benchmark techniques (for 4 cases mentioned above in best f1-score, Precision, and Recall. The observations are given as follows:

1) When compared to the one-phase standard techniques, it is evaluated that the proposed MVar-TEF model provides the most favorable F1 score.

2) When all four cases are employed with Lemma_1 (operation signal cycles), the results exhibit outstanding efficiency irrespective of performance measures. This depicts that operation cycle signals work well when compared to the sensor signals in terms of performance;

3) Even though the MVar-TEF recall value is similar to that of Lemma_1 with deep auto-encoder, its precision significantly increased by utilizing sensor signals at the second phase of the MVar-TEF. Specifically, sensor signals are used as a backup method to raise the highest F1 score even higher. This suggests our approach helps eliminate improbable anomalous locations (i.e., reduce FP (false positives)).

4) The vector combination of two signals reduces the detection accuracy, according to an analysis of performance between Lemma_1, Lemma_2 and Lemma_4;

5) The comparison of 5 benchmark algorithms in Lemma_1 shows that deep autoencoder performs the best in ideal F1 score, with auto-encoder and Gaussian mixture coming in second and third, respectively;

6) The LSTM and LSTM auto-encoder models exhibit poor results, suggesting that utilizing LSTM layers initially intended for temporal dependence learning will worsen detection accuracy.

4.7. Agnostic property-based analysis

By utilizing the five benchmark techniques, we differentiate the underlying models in stages 1 and 2 to demonstrate the agnostic feature of the proposed MVar-TEF architecture. Since employing either of the other methods does not perform suitably in stage 1. When operation cycle signals are used, we select deep auto-encoder, A-D, and GM as model candidates. In contrast, we select the five approaches as model candidates for stage 2, which uses sensor signals. Tables 1 to 4 compare the performance of the model combinations (•,•) in (stage 1 and stage 2) of MVar-TEF regarding f1-score, precision, and recall. Here are the results that we found. First, experimental findings demonstrate our MVar-TEF model-agnostic characteristic and demonstrate that adopting two stages improves performance compared to the single-phase benchmark approaches. Despite the models used in stage 1, Tables 2 and 3 indicate that the single-phase strategy yields the superior F1-score out of 4 cases (Lemma_1), the same as employing a fully connected network in stage 2 of MVar-TEF. Second, the tendency of sensor signals to exhibit substantial temporal dependence suggests that the model is more suitable in stage 2 compared to other standard approaches. Tables 1 to 4 conclude that the optimal model combination is in our MVar-TEF architecture.

4.8. Efficiency analysis

We detect anomalies on two additional assembly process information sets whose features are suitable for our empirical investigation in terms of both a non-negligible fraction of alarms and balanced attributes in each of the diverse signals to authenticate the reliability of the proposed MVar-TEF structure. Specifically, we observe that a relatively small amount of operation cycles or sensor signals are present in 32/39 assembly processes, indicating that our design that takes advantage of diverse time series signals may not be required in these cases (i.e., adopting a conventional single-phase recognition model instead of multiple phases techniques would result in satisfactory performance). To ensure reliable validation due to the scarce labels, we have removed any assembly process dataset (of the remaining 7) from our studies with fewer than thirty alarms. As a result, we select the following two datasets related to assembly processes:

1) After the feature selection procedure, the dataset present in the fifth assembly comprises twenty-two attributes in the operation cycle signals and seven characteristics in the sensor signals.

2) Next to the feature selection, the 12^{th} assembly process data source contains 6 and 11 features in sensor signals and operation cycle signals, respectively.

To validate the proposed MVar-TEF system on additional datasets, we show the efficiency assessment of MVar-TEF versus three one-phase standard models, wherein our MVar-TEF framework uses the LSTM and DAE algorithms in stages 1 and 2, respectively. Due to their inability to achieve the same level of performance, we

do not employ the additional two standard techniques that outperform one-phase standard methods on both assembly process datasets, indicating the robustness of our MVar-TEF architecture against different assembly process information.

5. Conclusion

In this study, we presented a new two-stage model known as MVar-TEF for mechanically identifying abnormalities employing real-world manufacturing information gathered from an automobile engine valve assembly line containing several complex issues. More specifically, we designed our MVar-TEF architecture after independently training a technique in stage 1 with signals and a different technique in stage 2 with signals. Signal-specific identification modules are executed in a data-driven fashion. Extensive experiments were conducted to show that the developed architecture is durable to fringe conditions and significantly outperforms many single-stage standard approaches. Furthermore, we discovered that our MVar-TEF detection process depends on various signals and carefully uses sensor signals as a backup tool to enhance efficiency further. This provides enhanced advantages over single-stage standard techniques. Prospective research directions encompass the resolution of a generic scenario involving increased heterogeneity and over three signal types in extremely complex real-world production. Future work will also include early identification of abnormalities for preventative maintenance.

References

[1] A. Deng and B. Hooi, ''Graph neural network-based anomaly detection in multi-variate time series,'' in Proc. AAAI Conf. Artif. Intell., 2021, vol. 35, no. 5, pp. 4027–4035

[2] Chen, D. Chen, X. Zhang, Z. Yuan, and X. Cheng, ''Learning graph structures with transformer for multi-variate time-series anomaly detection in IoT,'' IEEE Internet Things J., vol. 9, no. 12, pp. 9179–9189, Jun. 2022.

[3] Hu, X. Feng, Z. Ji, K. Yan, and S. Zhou, ''A novel computational approach for discord search with local recurrence rates in multi-variate time series,'' Inf. Sci., vol. 477, pp. 220–233, Mar. 2019.

[4] Geiger, D. Liu, S. Alnegheimish, A. Cuesta-Infante, and K. Veeramachaneni, ''TadGAN: Time series anomaly detection using generative adversarial networks,'' in Proc. IEEE Int. Conf. Big Data (Big Data), Dec. 2020, pp. 33–43.

[5] Kieu, B. Yang, C. Guo, and C. S. Jensen, ''Outlier detection for time series with recurrent autoencoder ensembles,'' in Proc. 28th Int. Joint Conf. Artif. Intell., Aug. 2019, pp. 2725–2732

[6] Xu, H. Wu, J. Wang, and M. Long, ''Anomaly transformer: Time series anomaly detection with association discrepancy,'' in Proc. Int. Conf. Learn. Represent., 2022, pp. 1–12.

[7] Ji, Y. Wang, K. Yan, X. Xie, Y. Xiang, and J. Huang, ''A space embedding strategy for anomaly detection in multi-variate time series,'' Exp. Syst. Appl., vol. 206, Nov. 2022, Art. no. 117892.

[8] Tuli, G. Casale, and N. R. Jennings, ''TranAD: Deep transformer networks for anomaly detection in multi-variate time series data,'' Proc. VLDB Endowment, vol. 15, no. 6, pp. 1201–1214, Feb. 2022.

[9] Li, X. Peng, J. Zhang, Z. Li, and M. Wen, ''DCT-GAN: Dilated convolutional transformer-based GAN for time series anomaly detection,'' IEEE Trans. Knowl. Data Eng., vol. 35, no. 4, pp. 3632–3644, Apr. 2023.

[10] Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, ''Deep one-class classification,'' in Proc. Int. Conf. Mach. Learn., 2018, pp. 4393–4402.

[11] Shin, S. Lee, S. Tariq, M. S. Lee, O. Jung, D. Chung, and S. S. Woo, ''ITAD: Integrative tensor-based anomaly detection system for reducing false positives of satellite systems,'' in Proc. 29th ACM Int. Conf. Inf. Knowl. Manage., Oct. 2020, pp. 2733–2740.

[12] Kim, H. Kang, and P. Kang, ''Time-series anomaly detection with stacked transformer representations and 1D convolutional network,'' Eng. Appl. Artif. Intell., vol. 120, Apr. 2023, Art. no. 105964.

[13] Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, ''Robust anomaly detection for multi-variate time series through stochastic recurrent neural network,'' in Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Jul. 2019, pp. 2828–2837.

[14] Li, Y. Zhao, J. Han, Y. Su, R. Jiao, X. Wen, and D. Pei, ''Multi-variate time series anomaly detection and interpretation using hierarchical intermetallic and temporal embedding,'' in Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining, Aug. 2021, pp. 3220–3230

[15] Qin, Y. Luo, and G. Tao, ''Memory-augmented U-transformer for multi-variate time series anomaly detection,'' in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Jun. 2023, pp. 1–5.

[16] Doshi, S. Abudalou, and Y. Yilmaz, ''Reward once, penalize once: Rectifying time series anomaly detection,'' in Proc. Int. Joint Conf. Neural Netw. (IJCNN), Jul. 2022, pp. 1–8.

[17] Bozic, P. Palafox, J. Thies, A. Dai, and M. Nießner, ‘‘TransformerFusion: Monocular RGB scene reconstruction using transformers,’’ in Proc. Adv. Neural Inf. Process. Syst., vol. 34, 2021, pp. 1403–1414.
[18] Jiang, S. Chang, and Z. Wang, ‘‘TransGAN: Two pure transformers can make one strong GAN, and that can scale up,’’ in Proc. Adv. Neural Inf. Process. Syst., 2021
[19] Zhang, J. Xie, N. Barnes, and P. Li, ‘‘Learning generative vision transformer with energy-based latent space for saliency prediction,’’ in Proc. NIPS, vol. 34, 2021, pp. 15448–15463.
[20] Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, ‘‘Decision transformer: Reinforcement learning via sequence modeling,’’ in Proc. Adv. Neural Inf. Process. Syst., vol. 34, 2021, pp. 15084–15097.
[21] Xu, ‘‘Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting,’’ in Proc. Adv. Neural Inf. Process. Syst., vol. 34, 2021, pp. 22419–22430.
[22] Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, ‘‘SegFormer: Simple and efficient design for semantic segmentation with transformers,’’ in Proc. Adv. Neural Inf. Process. Syst., vol. 34, 2021, pp. 12077–12090.
[23] Liu, J. Li, and M. Zhu, ‘‘Improving text generation with dynamic masking and recovering,’’ in Proc. 13th Int. Joint Conf. Artif. Intell., Aug. 2021, pp. 3878–3884.
[24] Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ‘‘BERT: Pre-training of deep bidirectional transformers for language understanding,'' in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol. Minneapolis, MN, USA: Association for Computational Linguistics, vol. 1, Jun. 2019, pp. 4171–4186.
[25] Cheng, X. Chen, D. Zhang, and L. Lin, ‘‘Motion-transformer: Selfsupervised pre-training for skeleton-based action recognition,’’ in Proc. 2nd ACM Int. Conf. Multimedia Asia, Mar. 2021, pp. 1–6.
[26] Bao, L. Dong, S. Piao, and F. Wei, ‘‘BEit: BERT pre-training of image transformers,’’ in Proc. Int. Conf. Learn. Represent., 2022, pp. 1–18.
[27] Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, ''A transformer-based framework for multi-variate time series representation learning,'' in Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining, Aug. 2021, pp. 2114–2124.
[28] Sherubha, "Graph-Based Event Measurement for Analyzing Distributed Anomalies in Sensor Networks," Sådhanå(Springer), 45:212, https://doi.org/10.1007/s12046-020-01451-w
[29] Sherubha, "An Efficient Network Threat Detection and Classification Method using ANP-MVPS Algorithm in Wireless Sensor Networks," International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075, Volume-8 Issue-11, September 2019
[30] Sherubha, "An Efficient Intrusion Detection and Authentication Mechanism for Detecting Clone Attack in Wireless Sensor Networks," Journal of Advanced Research in Dynamical and Control Systems (JARDCS), Volume 11, issue 5, Pg No. 55-68