Robust anomaly detection for multivariate time series

Through temporal Graph Convolutional Network(GCN)s

and

Attention based Variational Autoencoder(VAE)

MD FIROZUR RAHMAN
firozur.rahman@fau.de

Outline

Introduction

Anomaly detection on multivariate time series (MTS) is of great importance in both data mining research and industrial applications.

Generally, it is preferred to identify entity anomalies at the entity-level directly using MTS, rather than at the metric-level using univariate time series.

To tackle the aforementioned challenges, This paper propose a novel unsupervised **MU**ltivariate Time Series **AN**omaly de**T**ection framework called MUTANT. It combines Graph Convolutional Networks (GCNs) and Variational Autoencoders(VAE) architecture.

MUTANT stands out because it understands how variables interact over time, identifies crucial variables for anomaly detection in real-time, and achieves a significant 7.18% improvement in F1-score accuracy over existing methods.

Problem Statement

Anomaly detection has been widely studied in different domains (e.g., log messages, time series, graphs, etc.), aiming at finding which instances significantly deviate from the other observations in the same dataset.

This paper addresses primary obstacles:

1. Time-Varying Correlations Between Variables.

2. learning variable importance based on temporal dependencies

3. End-to-End Optimization for Anomaly Detection in MTS

By overcoming these challenges, MUTANT aims to advance the state-of-the-art in MTS anomaly detection, offering improved accuracy and robustness in detecting anomalies that evolve dynamically over time and across multiple variables.

Proposed MUTANT

To address the aforementioned challenges, we propose a novel unsupervised Multivariate Time series Anomaly deTection framework (MUTANT) based on Graph Convolutional Network (GCN) and Variational Autoencoder (VAE).

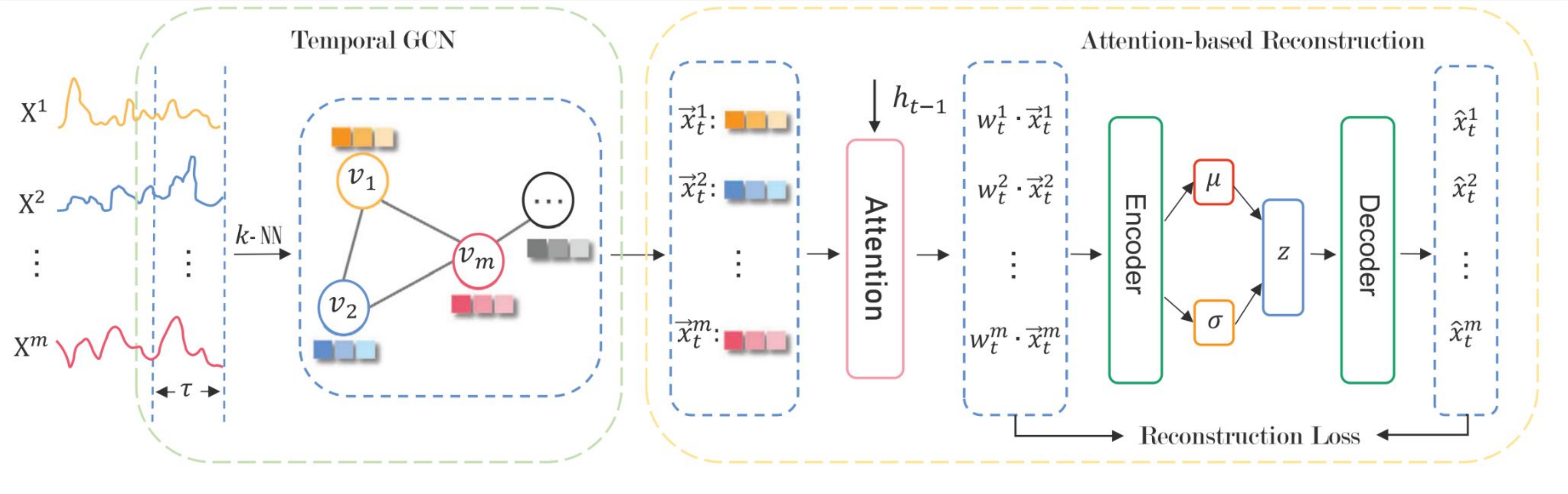
Specifically, we work in three steps:

1. We first construct a feature graph based on variable features for each time window in MTS.

2. Secondly, we design an attention-based reconstruction model.

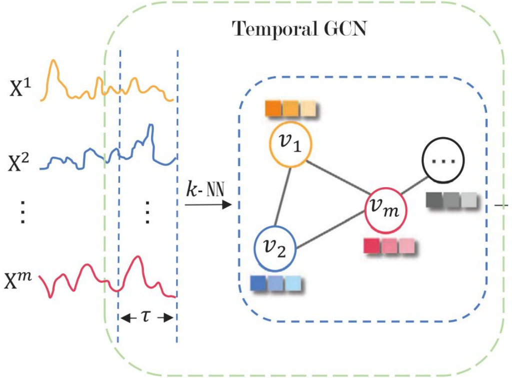
3. Finally, we use a Variational Autoencoder (VAE) module to learn the latent intrinsic representation for each observation to capture MTS patterns.

MUTANT-Framework



The overview of the proposed MUTANT.

Graph Convolutional Network



Correlation coefficient

$$\rho i, j = \frac{Cov(f_t^i, f_t^j)}{\sqrt{Var[f_t^i] \;\; Var[f_t^j]}}$$

Where

$Cov(f_t^i, f_t^j)$ is the covariance of f_t^i and f_t^j

$Var[f_t^i]$ is the variance of f_t^i

$Var[f_t^j]$ is the variance of f_t^j

$X = \{x_1, x_2 \ldots\ldots\ldots\ldots x_n\}$

τ is the length of the window.

τ = W:t = $\{x_{t-\tau+1} \ldots\ldots\ldots\ldots x_{t-1}, x_t\}$

Here perform the following layer wise propagation rule in a multi-layer convolution network.

$$H_t = ReLU(D_t^{-\frac{1}{2}} A_t D_t^{-\frac{1}{2}} H_t W_t)$$

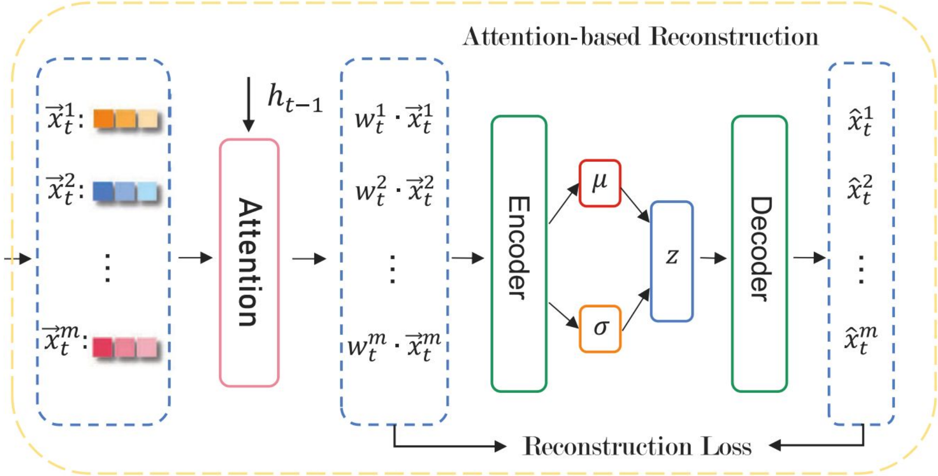
Where W_t is a layer-specific trainable weight matrix , $A_t + I$

Here

lth layer for time widow W_t

d is the embedding dimension.

Attention-based Reconstruction



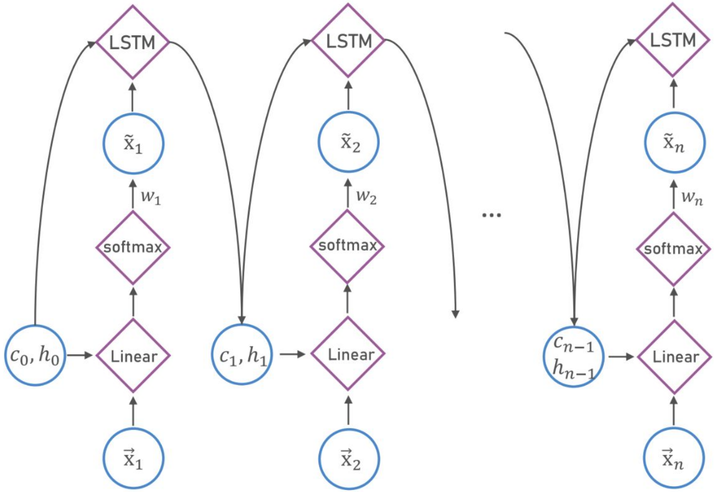
After obtaining the embedding vector of each variable, we use a reconstruction model to better obtain the essential characteristics contained in the MTS for anomaly detection.

This module mainly includes two parts:

1. LSTM-based attention
2. VAE-based reconstruction module.

VAE compresses high-dimensional input xt into low-dimensional latent representation zt by dimensionality reduction, and reconstruction xt by zt.

LSTM-based attention



The architecture of LSTM-based attention module.

LSTM unit has two transmission states:

1. Current state c_t
2. Hidden previous state h_t

The embedding vectors as the input of the LSTM unit to obtain weights of variables in the next window.

Algorithm

Spedu-code of MUTANT framework under the guidance of loss function.

Input: Multivariate time series X, parameters n ,τ, k
Output: The value of loss

1. For t = τ to n do
2. $w_t = X[t - \tau + 1]$ (*time window at time t*)
3. $\rho \leftarrow Pearson\,(w_t)$
4. $G_t \leftarrow K - NN$ (feature graph of w_t)
5. $X_t \leftarrow GCN$
6. $X_t \leftarrow LSTM\ based\ attention\ (x_t)$ (weighted embedding of w_t)
7. $X_t \leftarrow VAE(X_t)$ (reconstructor vector)
8. End for
9. Calculate Loss joint
10. Back propagation and updated parameters in MUTANT
11. Return Loss joint

Experiment

We conduct extensive experiments on four publicly available real-world datasets.

Dataset: Soil Moisture Active Passive (SMAP)
The SMAP dataset contains data for 55 entities monitored by 25 metrics.

The statistics of benchmark datasets. (%) is the ratio of anomalies in each test set.

Dataset name	Train	Test	#entities	#dimensions	Anomalies (%)
SMAP	135,183	427.617	55	25	13.13

Time complexity analysis

Time complexity analysis:
We now analyze the time complexity of our proposed MUTANT for detecting anomalies in MTS.

1. Temporal Graph convolutional network (GCN) time complexity is $O(N_e d)$

    Where

    N_e is the number in feature graph G,

    d is the dimension of embedding.

2. LSTM-based attention, computation complexity of LSTM is $O(nm^2)$

    Where

    n is the length of MTS

    m is the number of variables

3. Variational autoencoder (VAE) computational complexity is $O(n^2 m^2)$

MUTANT time complexity is $O(N_e d + nm^2 + n^2 m^2)$

Comparison and Evaluation metrics

We use Precision, Recall, and F1-score to evaluate the performance of our proposed model and baselines, which are defined

$$Precision = \frac{TP}{TP+FP} \qquad Recall = \frac{TP}{TP+FN} \qquad F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Anomaly detection accuracy in terms of precision, recall, and F1-score on SMAP datasets.

Method name	Precision	Recall	F1-Score
USAD	0.7697	0.9831	0.8634
DAEMON	0.929	0.892	0.910
AMSL	0.9431	0.9218	0.9323
LSTM-NDT	0.8455	0.9096	0.8764
MUTANT	**0.9788**	0.9719	**0.9753**

Result

```
(base) frzf7@frzf7 MUTANT % python main.py
Loading data of: SMAP
Train range: 0 None
Test range: 0 None
Data normalized
Data normalized
Train set shape: (135183, 25)
Test set shape: (427617, 25)
Test label shape: (427617,)
test_data: (277952, 25)
val_data: (149665, 25)


 Data-calculate- \-Done

********************* Final result of Test data ********************************

Dataset-Name: SMAP
Threshold-value: 99.2500000000032
True Positive: 41805
False Positive: 1855
False Negative: 613
precision: 0.9575125971237946
recall: 0.9855485876313006
f_score: 0.9713233309601832
```

Application

MUTANT provides a versatile framework designed for robust anomaly detection within multivariate time series data across a range of research and practical domains. Its primary strength lies in its proficiency at analyzing multivariate time series data, which is a prevalent data format across numerous fields.

1. Industrial Applications: MUTANT facilitates predictive maintenance in manufacturing by identifying anomalies in sensor data (such as temperature and pressure) from machinery. This capability enables proactive interventions to mitigate costly downtime.

2. Financial Sector: Within the realm of finance, MUTANT plays a crucial role in anomaly detection within transaction data. This functionality is vital for fraud prevention efforts, as it can identify suspicious activities.

3. Cybersecurity: MUTANT assists network intrusion detection systems by scrutinizing network traffic patterns, thereby enabling the detection of potential cyberattacks based on anomalous behavior.

Despite its significant advantages, implementing MUTANT may require expertise in deep learning and time series analysis techniques.

Conclusion

This proposal introduces a novel unsupervised method, MUTANT, for multivariate time series (MTS) anomaly detection. MUTANT initially constructs a feature graph based on variable features for each time window in the MTS and employs Graph Convolutional Networks (GCN) to learn embeddings for all variables.

MUTANT utilizes an attention-based module to identify variable importance and a Variational Autoencoder (VAE) to capture normal MTS patterns. End-to-end training optimizes the model's performance.

For future work, we are keenly interested in developing a more robust self-supervised learning framework based on contrastive learning for MTS anomaly detection.

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Thank you

For your time