Anomaly detection in time series for TBM

A literary review of the useful AD-method for Tunnuling Machines

Mathieu Giamberini, 2024

Abstract

Introduction

By default, TBM (Tunnel Boring Machines) are complicated technical objects and involved a lot of technologies working together. Any defect in the chain could stop the entire machine. Therefore, detecting those before they become a real issue can save a lot of time and resources. Fortunately, the number of sensors on the machines keeps raising and thanks to platforms like HK-Connect, their data can be accessed anywhere by any authorized user. These data must carry a lot of information regarding the state of the machine, which begs the question of how to extract it to detect anomalies. At the time of writing, the task is mostly done by experienced users of the TBM, e.g. a weird noise in the hydraulic pack, an odd torque increase on one of the main drive motors, and so on. This method is clearly limited and uses only a small portion of the available data stream. Despite this, the problem of anomaly detection is quite old, and with the rise of machine learning technic, it keeps getting better.
Therefore, in this paper, we will, have a first look of at the literary landscape of Anomaly detection method (ADM). First, let's define exactly what we mean by that, given a multivariate time series:

$$\mathcal{X} = (\mathbf{x}_1, \dots \mathbf{x}_t, \dots \mathbf{x}_T) \in \mathbb{R}^{N \times T}$$

with N the number of sensors and T the number of temporal indices. The method $\mathcal{M}$ will return a binary category for each data point telling where and when there is anomalies, ie:

$$\mathcal{M} : \mathbb{R}^{N \times T} \to \{0, 1\}^{N \times T}$$

We need to address some looses ends with this definition. First, in most cases, there is no already labeled data to train the method on, and labeling it by hand is not feasible. So the method must be unsupervised, which raises the need for a systemic definition of anomalies. Most papers (ref) consider the training data to be normal and define some anomaly score, which tells how far away the new data is. Here, the methods will, be categorized based on the way this score is defined. In our context, the training data is previous tunnel bore, which certainly contains anomalies. Therefore, we will also discuss some papers which tried to address this issue by modifying the methods.

1 Secondary sources

2 Overview

Anomaly detection method can be categorize in many way. In this paper its divided in two parts, general method and useful ticks to improve theses. The former represent general ways to solve the AD problem. Its compose of tree categorizes : forecasting, clustering and index monitoring. The latter are so called add-on. The papers in this category present a AD method with some tricks which, in our mind can be applied for any method to improve it.

2.1 Forecasting Methods

When working all day long with some equipment, you know it so well that if asked you could almost predict the sound it will make in some close future. But if suddenly sound doesn't match your expectation you know there is some things wrong. That is the core idea for the forecasting method, predict the near future using some generative method and if the error go over some threshold the data point is labeled as an anomaly. Example of those technic may be LSTM [ref claim LSTM is the best] or using time convolution (TCN) [Ref TCN].

2.2 Clustering method

3 Description

- Cluster/Density (probability)
  - idea, Training data is normal data,
  -
- Anomaly score
- Forecasting
- Graph
- Training with data Contamination

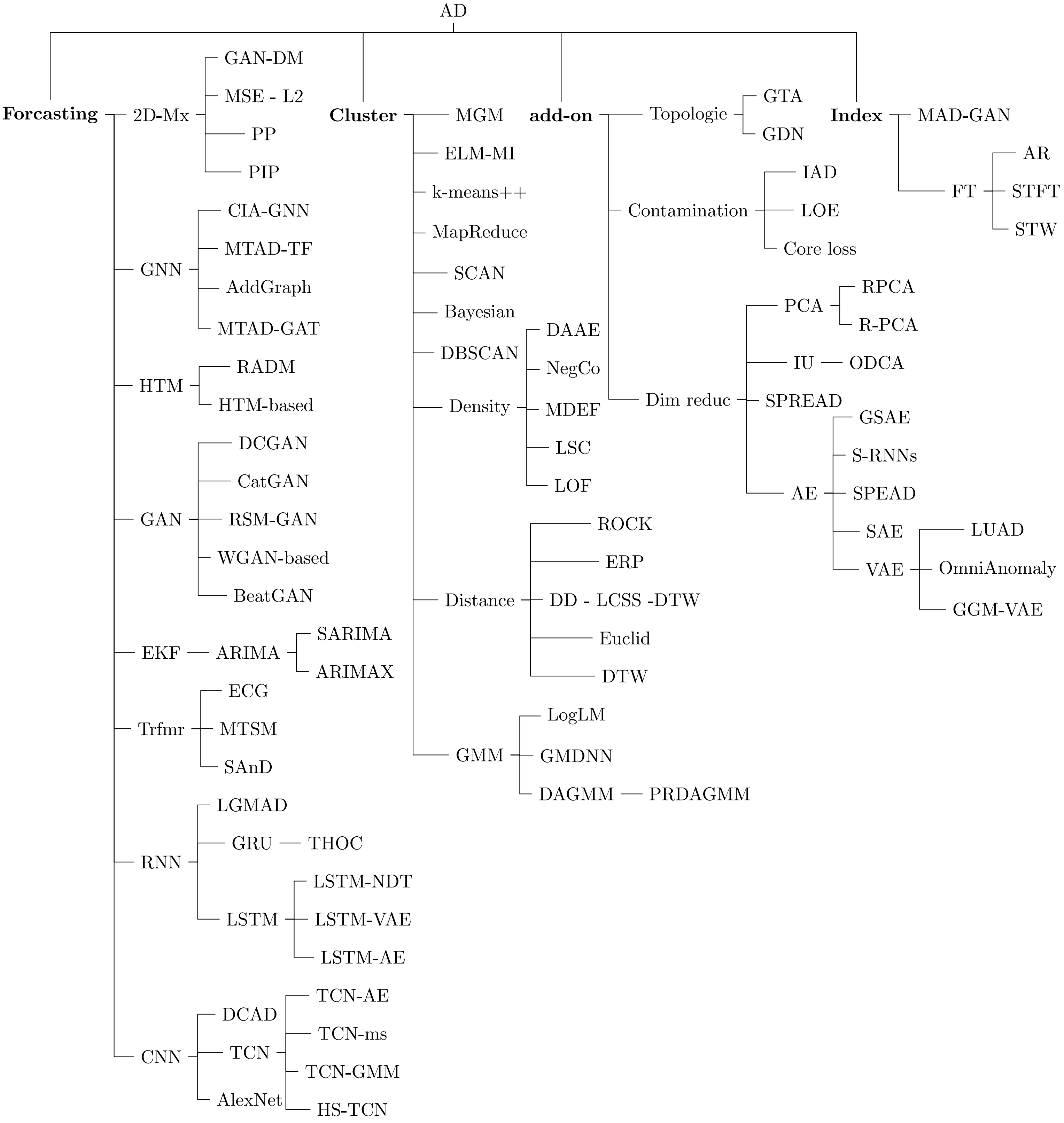

Figure 1: Anomaly detection method categorization

3.1 Graph Deviation Network (GDN)

When using "distance to prediction" for AD in a multivariate settings, it's sensible to assume that related sensors have redundant information (e.g. a pressure and a extension sensors for a hydraulic cylinder). So if there measurement was put together to predict the next ones it could improve their prediction and so reduce the detection threshold. This is a base idea of the GDN architecture in [**?**]. Mark the related sensor in a directed Graph and apply an attention layer on the past measurement of the neighborhoods to predict the next ones. Here is some short assertion and nice propriety of this approach :

- Even if every sensor of the TBM are in the graph, the AD is still localize to each sensor.
- The training data is assume to be free of anomaly.
- Prior knowledge of relation between sensor can be embedded in the graph by restricted some relationship. This can lower the complexity and the time of training.

To test their method the authors used dataset of simulated attack on water treatment physical test-bed systems, the Secure Water Treatment (SWaT) and Water Distribution (WADI) dataset. As a base line they use other method which we discus in other Section.

Conclusion