Status of Data-Driven Beam Trajectory Anomaly Detection at European XFEL

3rd ICFA Beam Dynamics Mini-Workshop on Machine Learning Applications for Particle Accelerators

Antonin Sulc

Chicago, IL, 2 Nov 2022

IN NATURAL SCIENCES

TUHH
Technische Universität Hamburg

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

European XFEL - Overview

The European XFEL:

- Consists of superconducting cavities that boost electrons.
- The electrons are then directed through specially arranged magnets (undulators).
- Then they emit extremely short and intense X-ray flashes.
- These X-ray flashes are then distributed to three beamlines (SASE).

Data-Driven Predictive Maintenance on European XFEL

- Thousands of devices are involved in EuXFEL.
- Many components are operating in **extreme conditions** (radiation, heat...).
- At any moment, any component can **fail**.
- When a component fails, it can lead to undesired **downtime**.

Examples

Orbit Monitoring

- Analyzing electron orbits in undulator SASEs.
- Various types of problems are indicated by variations in orbits.

Orbit Monitoring

Assumption

There is a systematic pattern shown in orbits given by the physical construction of EuXFEL.

Orbit Monitoring - FODO Lattice

[Wendt(2011)]

[Holzer(2006)]

Orbit Monitoring

Orbit Monitoring

Data-Driven Orbit Monitoring

Model-Based Orbit Monitoring

- Credits to **R. Kammering**.
- Fit a sine and measure residual.
- **Pros** It is **fast** and **intuitive**.
- **Cons** No intra-bunching instabilities.

Model-Free Orbit Monitoring

- Series of BPM read-outs.
- Train a model with the anomaly loss.
- **Pros** Takes into consideration intra-bunching instabilities.

Data

- **No labels**, just logbook records.
- We rarely have an **interpretation** how a failure is propagated to orbit (if so).
- We have 84351 minutes which spans from **Feb 25 2022** to **May 19 2022**.
- Each minute record has ≈600 macropulses ordered sequentially.
- Only **first bunch** in the bunch train is considered!

BPM/position (each line is one bunch)

μ-pulse vs BPM (stack of bunches)

Model-Free Data-Driven Orbit Monitoring

- Units that are scoring a series of BPM read-outs.
- Each input is stacked in horizontal and vertical positions from all beamlines.
- We **do not** have any **faulty labels**.
- **Unsupervised one-class anomaly** loss [Ruff(2018)]:

$$L(\theta) = \|f_\theta(\mathbf{x}) - \mathbf{c}\|_2$$

- Trained features f_θ can be used for further analysis, e.g. TSNE embedding.

Models

Linear - The Simplest

Bunches **scored independently**, i. e. no correlations of adjacent bunches are taken into consideration.

Recurrent - The Classics for Sequences

Has an **”internal state”** and takes into consideration other bunches and their order

Convolutional - Translation Invariant

Does not have an internal state, but is **translation invariant**. It still considers the order.

Transformers - Permutation Invariant

Takes co-occurrences of some bunches into consideration (if positional encoding is not used) - i. e. **permutation invariant**.

Linear Units

- The simplest way to go.
- Individual bunches are independent.
- Surprisingly powerful already
- Very hard to interpret (in general)

Recurrent Neural Networks - Old Classics

- A classic way to handle sequences, literally a few lines of code to make a model.
- Slow (very inefficient on GPUs).
- Results are similar to the purely linear model.

Transformers

$$\underbrace{\underset{\text{input=}|\text{BPM}|,\ \text{output=128}}{\mathbf{h}_1 = \text{Transformer}(\mathbf{x})} \longrightarrow \underset{\text{input=128, output=16}}{\phi = \text{Linear}(\mathbf{h}_1)}}_{f_\theta(\mathbf{x})} \longrightarrow \underset{\text{input=64, output=1}}{s(\phi) := \|\phi - \mathbf{c}\|_2^2}$$

- Transformers have self-attention mechanism, puts other inputs into a **"context"**
- **Permutation** invariant, scores all bunches in correlation with other bunches.
- No position is taken into consideration, only co-occurrences.

The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .
The FBI is chasing a criminal on the run .

https://lilianweng.github.io/posts/2018-06-24-attention/

Convolutional Neural Networks

$$\underbrace{\underset{\text{input}=|\text{BPM}|}{\mathbf{h}_1 = \text{Conv2d}(\mathbf{x})} \rightarrow \mathbf{h}_2 = \text{Conv2d}(\mathbf{h}_1) \rightarrow \underset{\text{output}=16}{\phi = \text{Linear}(\mathbf{h}_2)}}_{f_\theta(\mathbf{x})} \rightarrow \underset{\text{input}=16,\ \text{output}=1}{s(\phi) := \|\phi - \mathbf{c}\|_2^2}$$

- Translation invariant.
- Unlike RNNs very fast on GPUs!
- The dynamic number of bunches was eliminated via calculating mean/maximum/sum over features along bunch dimensions.

[https://mlnotebook.github.io/post/CNN1/]

CNN + Transformers

- **Translation and permutation** invariant.
- Still fast on the GPU.
- Fixes the problem with the dynamic number of bunches, each bunch is like a token after being processed by CNN.

Example 1

The undulator server crashed after an unusual selection of colours for individual cells

Linear

RNN

Transformer

CNN+Transformers

CNN

$s_{faulty} = 0.0639$

$s_{healthly} = 0.003$

CC BY

Example 2

An issue with a cold quadrupole in A6

Linear

RNN

Transformer

CNN+Transformers

CNN

$s_{faulty} = 0.0086$

$s_{healthly} = 0.003$

CC BY

Example 3

A CPU failure on one of the control boards

Linear

RNN

Transformer

CNN+Transformers

CNN

$s_{faulty} = 0.005$

$s_{healthly} = 0.003$

CC BY

Example 4

Safety magnet at SA1/SA3

Linear

RNN

Transformer

CNN+Transformers

CNN

$s_{faulty} = 0.0003$

$s_{healthly} = 0.003$

CC BY

Thank you for your attention!

This is the joint work of R. Kammering and T. Wilksen!

https://github.com/sulcantonin/ICFA-Beam-2022

Contact

DESY. Deutsches Elektronen-Synchrotron

www.desy.de

Antonin Sulc
0000-0001-7767-778X
MCS DESY
antonin.sulc@desy.de

Bibliography I

BJ Holzer.
Lattice design in high-energy particle accelerators.
2006.

L Ruff.
Deep one-class classification.
In International conference on machine learning, pages 4393–4402. PMLR, 2018.

Manfred Wendt.
Overview of recent trends and developments for bpm systems.
Technical report, Fermi National Accelerator Lab.(FNAL), Batavia, IL (United States), 2011.