Thu-Mo-Po.08-04: Building Datasets and Training Methods for ML Based Magnet Quench Detection

Maira Khan, Joe DiMarco, Steven Krave, Vittorio Marinozzi, Stoyan Stoynev, Jennifer Ngadiuba, Nhan Tran

Fermi National Accelerator Laboratory

FERMILAB-POSTER-25-0065-CSAID-TD

Introduction

Superconducting (SC) magnet diagnostics is a complex, multi-modal task that requires real-time processing of sensor data. In this work, we present data selection and augmentation techniques for training machine learning models using both acoustic and quench antenna (QA) data. We demonstrate how this training data can support the development of generalizable criteria for analyzing different magnet ramps during training. By formulating our approach as a supervised machine learning problem, we show that we can potentially enable more consistent quench detection and prediction during magnet training.

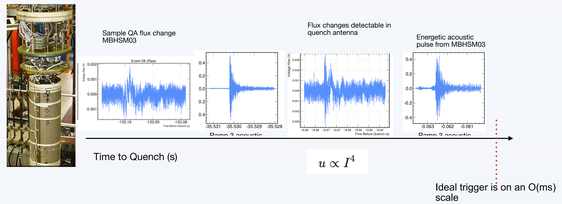


Figure 1: Pictorial representation of sensor changes we aim to capture throughout the ramp

Current and strain energy density increase as the magnet trains. This causes sensor events (seen as voltage rise) to be generated throughout the ramp. Events closer to the quench could be more "energetic", or posses different parameter distributions. Additionally, a wide variety of periodic/aperiodic disturbances might cause the quench.We would ideally like to capture all of the above in a real time detection algorithm

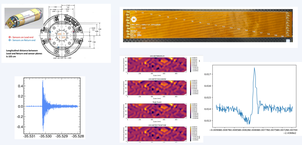


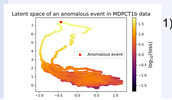
Figure 2: Diagnostic instruments used in quench detection

(Left) Acoustic sensor diagram for MQXFS1d and corresponding normalized signal pulse. Data is taking at 1MHz over 5-7 channels. See [3] for details of magnet training

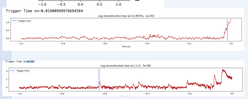
(Right) QA instrumentation with spatial localization and individual signals shown. Data is taken at 100kHz over 60 channels

Precisely how to formulate quench detection as a machine learning problem will affect how we structure our data curation and training. In general we can choose from the following approaches:

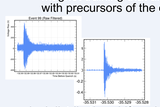
- Unsupervised learning with a neural network with some measure of statistical anomaly using a loss function such as $L_1 = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y}_i\right|$
- Unsupervised learning using clustering algorithms over some groupings of signal pulses. The include techniques like tSNE, DBSCAN on chosen features of interest from the data
- Supervised learning, where some clear objective or boundary is made. We aim to classify on the correct side of the boundary
- Weakly supervised learning where the boundary is unknown, but the model logits itself are used as a metric for finding the optimal boundary. We can take the individual class probabilities and use their ratio as a metric for the maximal separation point $p_1 = \frac{e^{z_1}}{e^{z_1}+e^{z_2}}, \quad p_2 = \frac{e^{z_2}}{e^{z_1}+e^{z_2}}$
- RL based (not shown here): where the model tracks gradual distribution shifts over time and adjusts inference in response



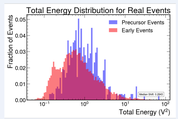
1) Do we define the quench precursor as a statistical anomaly? See [1]



2) Can we somehow classify and cluster sensor events that occur during magnet training that are associated with precursors of the quench or not?



3) Is this a gradual distribution shift problem?



4) Does it somehow depend on all of the above? Is it a generalizable criterion?

System Objectives

- **System Objectives:**
  - *Goal:* Quench Prediction before some *ms* of quench event
  - Modular and adaptable to different sensor types, sample rates, number of channels
  - Robust to changes in power supply noise
  - Prevents false triggers during training ramp (high amplitude of signal that does not correlate with quench)
  - **Software task**: Develop an algorithm that processes signals, and flags quench precursors within time and memory constraints
  - **Physics goal**: Predict quenches using metrics that capture energy loaded to the magnet, flux jumps, and current re-distribution during the training ramp to determine when a quench will occur

Template provided by PosterPresentations.com
www.PosterPresentations.com

System Objectives

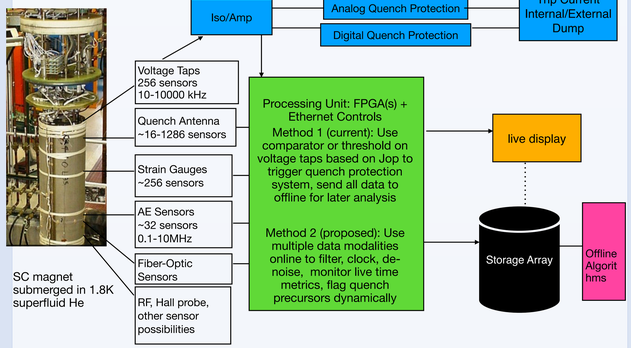


Figure 3: Schematic of aspirational DAQ system with ML integrated diagnostics

We aim to build an ML algorithm that is able to keep up with the data rate of the incoming sensors, and for this ML model to provide inference information on the likelihood that a given sensor causes a quench.

Data Selection and Augmentation

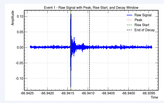


Figure 4: A signal event

To build a training dataset, we must first define some event selection parameters. We can then analyze statistics over these events. Our algorithm currently selects by threshold, and centers around the peak of the signal pulse as depicted. We can naively define two classes as a precursor vs. not based on the time *t* before the quench the event occurred.

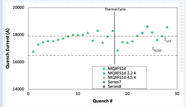


Figure 5: Maximum quench currents in a thermal cycle

We would like to ensure that our model does not simply pick up the Kaiser effect by finding a distributional shift due to the appearance of relatively less energetic events before exposure to the maximum current of the previous ramp.

So, we instead define and event e_i as between two ranges: occurring after the previous ramp's maximum current I_{n-1} or within *t* ms of the maximum current I_n within the same thermal cycle (TC) to perform our analysis

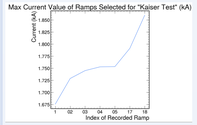


Figure 6: Chosen ramps for training and testing

For the purposes of our data training, we choose to train on five ramps, collecting all signal pulses and test on the signal pulses of two ramps which the magnet has *not* seen before, all pulse events satisfying this Kaiser condition.

We do this such that a magnet in training simply being ramped with a current is has not previously been ramped with automatically flags our algorithm as "a quench will occur". This is an essential process to magnet training as strain energy density loads.

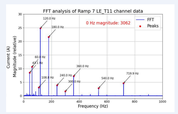


Figure 7 (left): FFT analysis of given QA channel by student Danming Peng

In order to prevent our event selection algorithm from picking up any possible sources of noise, we also apply a comb filter on all 60Hz harmonics of the signal to prevent false tagging of signal pulses

Once we have our event parameters defined, we can begin to do a statistical analysis of our two classes of events using parameters of interest such as signal energy, spectral entropy.

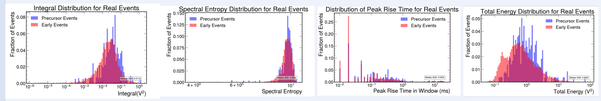


Figure 8: Distributions of early vs. precursor events
We find that there are some distinctions at the tails between these two choices of events, which show the potential for some discrimination. However, given the large overlap we find that a simple parameter based model may not be optimal.

Data Analysis Results

Given that the number of quench precursors vs. events that occur after the previous max current that are not precursors (naively in this case 200ms before the quench), we must find some way to augment our dataset. Below is the process for augmenting our data set from a given signal pulse sample.

1) Generate a function $\omega(t)$ from original precursor event $s(t)$.

   To do this we generate the FFT $X(f) = \mathrm{FFT}(s(t))$, extract a magnitude spectrum, add a random phase and perform an iFFT

2) Fit parameters A and β in decaying exponential from peak of original signal $s(t)$ assuming it is of the form $Ae^{-\beta t}(\sin\omega(t)t)$

3) Compute residual energy after frequency modulation. Add back residual energy with random amounts of log-normal and white noise.

$$E_{s_k} = \int_{t_i}^{t_j} |Ae^{-\beta t}(\sin\omega(t)t) + l(t) + w(t)|^2 = \int_{t_i}^{t_j} (|s_k(t)|)^2$$

We can evaluate the results of our event simulation algorithm by reproducing the same distributions as before.

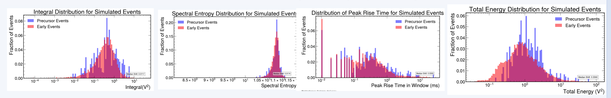


Figure 9: Distributions of simulated early vs. precursor events
We find similar distributions to our non-simulated data which indicates our simulation procedure, though not without bias is relatively accurate.

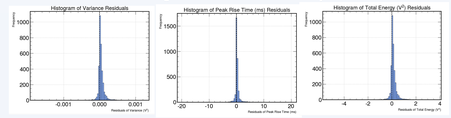


Figure 10: Residuals of simulated early vs. precursor events
We find our residuals are centered and minimally biased, indicating relative success of our simulation

Machine Learning Methods

We present the following machine learning methods and formulations for our quench detection problem.
1. Unsupervised learning using an auto-encoder
2. Supervised Learning using a CNN and SSM
3. Weakly Supervised Learning using a CNN

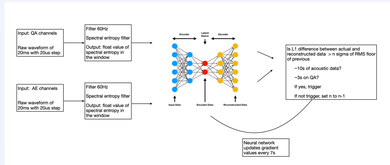


Figure 11: Schematic representation of auto-encoder and dynamic loss algorithm. The model encodes the streamed signal and depending on whether there is a relative statistical anomaly with respect to the previous RMS, and flags that a quench will occur

Thresholds are tuned based on the RMS values of the signals and L1 loss as shown in the introduction

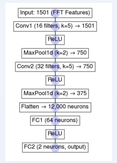


Figure 12: Schematic of CNN used in supervised model setting. FFT features are found to show the highest degree of discrimination over general statistical features which alone may include inductive bias. Note the last layer has a dimension of 2, for class output precursor or not. The identification of a precursor would result in a "'trigger" for quench prediction

Machine Learning Results

Unsupervised Machine Learning Results

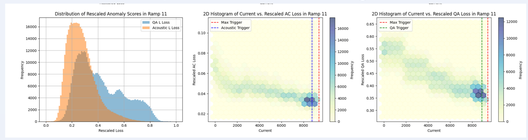


Figure 13: L1 loss scores computed on QA and Acoustic data workflows to detect relative "local anomalies". Loss accumulates and scales with respect to current, but does not achieve a measurement of some behavioral transition.

Machine Learning Results

	Spectral Entropy Reconstruction	Statistics Computation
GPU processing time of 20ms at 20us step per channel at 16bit precision	1 MHz: ~22us 100KHz:~17us	1 MHz: ~2us 100KHz: ~1us
	Dense Autoencoder	**U-Net Autoencoder**
GPU Inference time of 20ms window inputs at 16-bit precision	Acoustic Trigger: ~54us QA Trigger: ~79us	Acoustic Trigger: ~2ms QA Trigger: ~3ms
Best Total Time Trigger Efficiency within ~25s	**False Triggers: 1/29 Missed Triggers: 0/29 True Triggers: 28/29**	False Triggers: 2/29 Missed Triggers: 3/29 True Triggers: 24/29

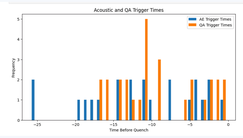


Figure 14: Histogram of trigger times
We find that the QA triggers occur more closely to the quench likely due to early disturbances in the ramp. Still the triggers occur within 25ms but this is a very loose metric for success.

Supervised Machine Learning Results

In the supervised learning case, our metrics are on an event by event basis. We are classifying each tagged event that is some type of signal as a precursor or not. Note we only use acoustic pulses in our results

	CNN with FFT features	SSM on FFT Features
AUC	0.74 (using 200ms definition as binary truth)	0.72 (using 200ms definition as binary truth
Trigger Efficiency per ramp within ~2s of the quench	False Triggers (on test set of ~3000 events): 4	False Triggers (on test set of ~3000 events): 2

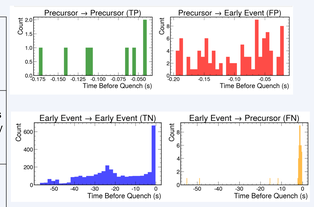


Figure 15: Histogram of triggers for CNN-FFT model
We find that most of the false triggers are concentrated within ~2s of the quench which indicates a detection of a behavioral shift

Given that our metric are not optimal in determining the exact notion of truth, i.e identifying some behavioral shift, we are now implementing weakly supervised methods to find the optimal time point for training in terms of classification. Initial studies show a behavioral shift occurring more concretely at 100ms before the quench

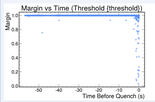


Figure 15: Calculating the difference of the two probabilities $p_1 = \frac{e^{z_1}}{e^{z_1}+e^{z_2}}, \quad p_2 = \frac{e^{z_2}}{e^{z_1}+e^{z_2}}$ allows us to establish a probability margin that shifts as a function of the time we choose to divide our class. We see high probability margins within ~100ms to the quench indicating our original choices was not optimal. We hope to expand on this soft label technique for a complete weakly supervised approach.

Discussion and Conclusions

We have shown the following data analysis milestones:
- A successful implementation of a scalable simple event tagging algorithm for building ML based training sets
- A distributional analysis that shows the degree of separation between events that could potentially cause the quench
- A simulation method for augmenting this dataset of these events

We have shown the following machine learning milestones:
- Implementation of an unsupervised auto-encoder with observation on the current scaling properties of the loss
- Successful implementation of a supervised method with fewer than 5 total false triggers in a given ramp per 30ms window
- Progress towards a weakly supervised pipeline by comparing relative probabilities of classification using different time points as the classification boundary

References

[1] Hoang, D., Boffo, C., Tran, N., Krave, S., Kazi, S., & Stoynev, S. (2021). Intelliquench: An adaptive machine learning system for detection of superconducting magnet quenches. IEEE Transactions on Applied Superconductivity, 31(5), Article 4900805. https://doi.org/10.1109/TASC.2021.3058229

[2] Stoynev, S., & DiMarco, J. (2022). Assessment and performance of flexible quench antenna array diagnostics for superconducting magnets (2LOr2B-08). In Proceedings of the 2022 Applied Superconductivity Conference. IEEE. https://doi.org/10.1109/TASC.2022.3146821

[3] Stoynev, S., Riemer, K., Zlobin, A. V., Ambrosio, G., Ferracin, P., Sabbi, G. L., & Wanderer, P. (2019). Analysis of Nb$_3$Sn accelerator magnet training (ASC2018-2LOr1B-03). IEEE Transactions on Applied Superconductivity, 29, Article 4001206. https://doi.org/10.1109/TASC.2019.2895554

Acknowledgements

This manuscript has been authored by FermiForward Discovery Group, LLC under Contract No. 89243024CSC000002 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics. MK, JN, and VM are supported by the U.S. Department of Energy (DOE), Office of Science, Office of High Energy Physics "Designing efficient edge AI with physics phenomena" Project (DEFOA-0002705)