Continuous Convolutional Neural Networks for Disruption Prediction in Nuclear Fusion Plasmas

William F. Arnold
Kim Jaechul School of AI
KAIST
will@mli.kaist.ac.kr

Lucas Spangher
Plasma Science and Fusion Center
Massachussets Institute of Technology
spangher@psfc.mit.edu

Cristina Rea
Plasma Science and Fusion Center
Massachussets Institute of Technology

Abstract

Grid decarbonization for climate change requires dispatchable carbon-free energy like nuclear fusion. The tokamak concept offers a promising path for fusion, but one of the foremost challenges in implementation is the occurrence of energetic plasma disruptions. In this study, we delve into Machine Learning approaches to predict plasma state outcomes. Our contributions are twofold: (1) We present a novel application of Continuous Convolutional Neural Networks for disruption prediction and (2) We examine the advantages and disadvantages of continuous models over discrete models for disruption prediction by comparing our model with the previous, discrete state of the art, and show that continuous models offer significantly better performance (Area Under the Receiver Operating Characteristic Curve = 0.974 v.s. 0.799) with fewer parameters.

1 Introduction

Effectively combating climate change requires significant decarbonization of the grid. Nuclear fusion has long been considered a "holy grail" for producing on-demand energy without significant land demands or waste, and many have quantified beneficial carbon impacts [3, 12]. Fusion was considered far away until recently, when renewed public-private enterprise in fusion has ushered in a wave of investment, interest, and supporting industries.

One major approach to generating fusion in laboratory plasmas is *magnetic confinement*, which leverages the tendency of plasmas to remain confined perpendicular to magnetic fields. Currently, the highest performing magnetic fusion concept is the *tokamak*, a toroidal vacuum chamber with external magnetic coils that help generate a large plasma current.

One of the largest challenges for conducting tokamak research and future commercialization are plasma *disruptions*, or the loss of plasma stability that may deposit large amounts of temperature and current on the tokamak's walls [8]. Quenches can inflict severe damage to the tokamak, requiring costly repairs and delaying future experiments. For more background, please see the Appendix.

Disruptions are difficult to model using physical or "first-principles" simulations due to the numerous causes of instability and unobserved factors. In response, several Machine Learning strategies have been previously suggested to estimate *disruptivity*, i.e. the probability of a disruption outcome, allowing timely activation of mitigation systems [9]. For instance, DIII-D, the U.S.'s largest tokamak, employs a random forest (RF) for plasma state monitoring [10]. Current state of the art models employ classical neural techniques, such as "Hybrid Deep Learner" (HDL), a convolutional Long-Short Term

Tackling Climate Change with Machine Learning: workshop at NeurIPS 2023.

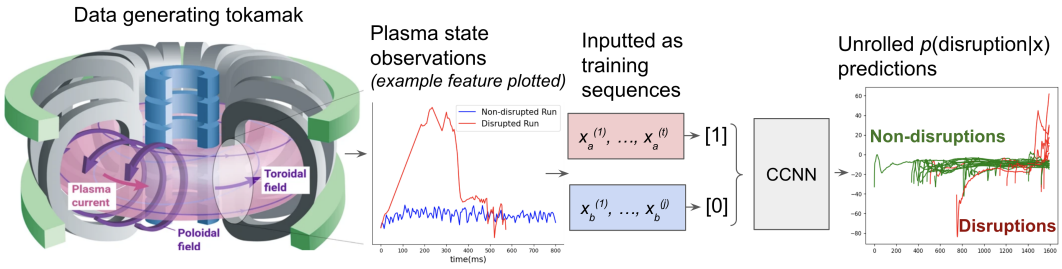


Figure 1: Disruption prediction framework. Tokamak picture taken from [2].

Memory network [14]. These models are still behind what is necessary for commercially viable fusion, where recall of at least 95% is required [8]. Moreover, they might not be tailored to a plasma dataset's unique structure. The RF and HDL treat the problem as a discrete time series, where the observation frequency is fundamental to the underlying system (i.e. like characters or words, which are fundamental units of measure in a sentence.) A tokamak's features' sampling rates are by no means fundamental, and vary depending on diagnostic instrument.

The model that we present here, the Continuous CNN [6], addresses these shortcomings. Instead of learning discrete filters directly, this model learns real, continuous functions that are that sampled. In our case, this is a mapping $\varphi : \mathbb{R} \to \mathbb{R}$ that maps time to filter weight. Here we use the Multiplicative Anisotropic Gabor Network (MAGNet) [11], that uses a combination of 1D convolutions and Gabor functions, (a product of a normal PDF and $\sin$). Learning a MAGNet instead of a discrete filter constrains the kernel to learn smooth features, even though it is later reduced to a discrete series of weights when passing through the sample's resolution.

We endeavor to create a model that surpasses the HDL in the Area Under the Receiver Operating Characteristic Curve (AUC) metric, and also investigate whether a continuous approach is superior.

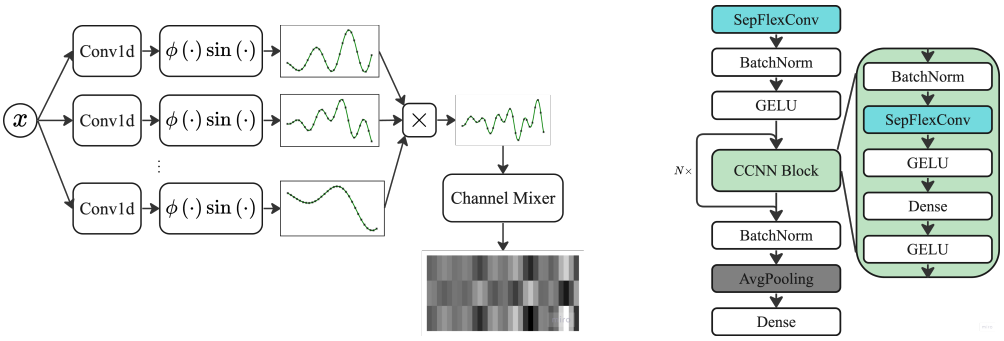
2 Model and Data

2.1 Dataset composition

Our dataset is composed of 4418 plasma shots from the Alcator C-Mod tokamak from MIT (C-Mod) [5]. Of these, 20% culminate in a disruption. The data is normalized to a sample rate of 200Hz (every 5ms). Shots with duration less than 125ms are excluded. Each shot is truncated 40ms prior to its end, as this is the minimum time needed for the activation of disruption mitigation systems. We did not train the model across multiple reactors, as detailed in [14], due to irregularities noticed in the data other than C-Mod, and because C-Mod is most similar to a new machines of interest [1]. Our main objective is to understand how this model performs on a single tokamak, and determine how predictable disruptions actually are when using continuous filters. We set up the training and test task as a sequence to label prediction. Each shot is a training example with a binary disruption label. Non-disruptive shots are augmented by randomly clipping them at shorter lengths. For an explanation of the features we use, please see table 2 in the Appendix.

2.2 CCNN Architecture

We instantiated the same architecture as described in [6]. We use separable `FlexConvs` (`SepFlexConvs`) from [11] with MAGNet kernels. A diagram of the MAGNet kernel network is detailed in Fig 2a. Discretized coordinate values x go through a 1D convolution, and are then passed into the Gabor function $\phi \sin$, where ϕ is a normal PDF with learned variance and mean. This is multiplied with the previous filter and passed through another 1D convolution (omitted in the diagram). This repeated N_f times and then passed through a Channel Mixer (another 1D convolution). Another portion of the network computes a continuous mask that truncates the filter at a given threshold. See [11] for details. Here, separability refers to the channel mixer inserted at the end of the MAGNet, as show in Fig 2a. Non-separable `FlexConvs` were also explored extensively and no improvement in performance was observed.



(a) A representation of how the MAGNet kernel is constructed. Coordinates x are transformed into sampled representations of continuous functions, which are then turned into a discrete convolution.

(b) The architecture of a CCNN. Filters are stacked with recurrent connections between dense layers, norms, and nonlinearities.

Figure 2: Summary of the CCNN architecture.

Although the model is trained for sequence-to-label prediction, it can also function as a sequence-to-sequence model. We replace the `AvgPooling` layer in the CCNN block with a moving average. This approach yielded better results compared to other label representation methods, such as using a vector of 1s or τ-windowing (0 then 1 for τ timesteps before the end). Modeling problems as sequence-to-label with a `AvgPooling` layer is observed in multiple high performing global convolution models such as [4, 6], and is key to performance.

Our network size is far smaller than any of those present in [6]. Our highest performing model has only 2,664 parameters, and increasing model size past did not appear to increase performance. Other than our the final dense layer, nearly our whole model is visualized in Fig 5. Smaller models are also reasonable performers: a model as small as 994 parameters was found with only a 0.005 drop in AUC. Since the convolution length does not depend on the number of parameters, high performance over long range dependencies is achievable without many variables. We also found that [6] uses large initialization values for some filters, resulting in sampling artifacts. This may be advantageous common benchmarks like Long Range Arena [13], but were very harmful to performance in our model. We lowered the initialization of ω_0 by a factor of 250x and observed increased performance.

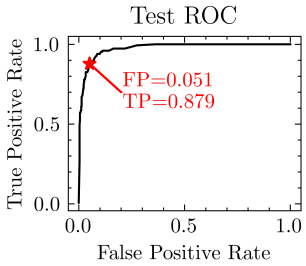
3 Results and Discussion

Performance is *exceptionally* higher than [14] with an AUC of **0.974** on C-Mod compared to 0.799. The ROC curve is shown in Fig 3a. As indicated, 87.9% of disruptions can be caught 40ms before disruption with only a 5.1% false positive rate. As shown in 1, we perform significantly better than [14] when more disruption data is present. Poor performance on case 2 is likely due to difficulty in dealing with the high class imbalance ($\tilde{1}76$ non-disruptions for each disruptive example).

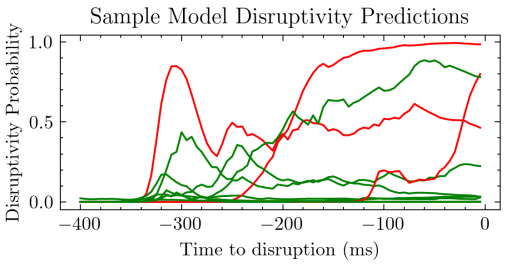
	Training Set Composition		AUCs	
Case no.	Non-disruptions	Disruptions	Baseline [14]	Ours
1	All	All	.799	**.974**
2	All	20	**.642**	.567
3	33%	All	.776	**.915**
Mean			.739	**.818**

Table 1: A table containing different training cases and output metrics

The last 400ms of shots are shown in Fig 3b. Here we see a variety of failure cases that our model encounters. Due to the sequence-to-label nature of our model, it can output positive disruption predictions ($p > 0.5$) early in the sequence and reduce this later, as seen in the red bump around $T = -300$ms. In a real-world use, outputting these weights would stop the current shot, and no further data would be observed. Modeling this characteristic of disruption prediction is difficult. Our data augmentation strategy of randomly clipping non-disruptive shots is intended to counteract this,



(a) The Test ROC Curve. A recall rate of 87.9% can be achieved with a false positive rate of 5.1%



(b) Sample Disruptivity plots over time. $T = 0$ is the time of disruption. Red lines denote disruptions, green lines denote non-disruptions.

Figure 3

but it is not always successful, especially as we do not augment disruptive examples. We qualitatively observe that most often, the model is able to predict disruptions 100-400ms before they occur.

3.1 Barriers to real-world use

While our model is efficient enough to be used in live disruption prediction (<3ms on CPU for a 1s shot without any optimizations), there remain are barriers to deployment. When generating the dataset, a number of features is obtained by running an equilibrium reconstruction code, i.e. EFIT [7]. EFIT features are available at runtime, but are more prone to generate missing data (represented as white noise) as they rely on more input diagnostics, and we have not yet robustly trained the model on data augmented by the random periods of white noise. Also, there are many data sources that we did not include, including photos and two-dimensional EFIT profiles. These are readily available during experiments, but require significant data-wrangling effort to collect and aggregate. Thus we have set a lower bound the model performance possible in disruption prediction, further motivating the collection of larger, more informative datasets.

3.2 Continuous vs. Discrete

Creating filters of arbitrary length with a very small number of parameters is extremely powerful for the task of disruption prediction. Our data size is inherently limited, and training a model with many discrete convolutions is unlikely to generalize well. We learn interpretable, continuous filters (see Fig 4), which are significantly longer range than what would be feasible using a discrete models, at 190ms, 655ms, 510ms, and 1360ms long (see Fig 5). Hence we believe continuous models show a distinct advantage for disruptivity prediction.

3.3 Architecture limitations

Many of the samples trained on were quite short (less than a second). Using a moving average before the dense layer at the end of the model for exceedingly long sequences may eventually limit the models ability to signal useful information due to an excessively large denominator. Solutions such as a windowed average, exponential average, or even a simple sum will be explored in the future.

4 Conclusion

Overall, this contribution could support the fight against climate change by bringing fusion energy closer to commercial viability. We present a new state-of-the-art model for disruption prediction based on the CCNN. It achieves greater performance metrics than a baseline, and motivates the development of future models, built with more data, which may greatly reduce harm of disruptions.

Acknowledgments and Disclosure of Funding

We thank PSFC for providing ample compute resources, William Brandon for advice on model setup, Andrew Maris for advice on physics, and Matteo Bonotto, Tommasso Gallingani, Daniele Bigone, and Francesco Cannarile for consistent technical collaboration, advice, ideas and feedback. This work was supported by Eni S.p.A. through the MIT Energy Initiative and by Commonwealth Fusion Systems under SPARC RPP021 funding. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2023-0-00075, Artificial Intelligence Graduate School Program(KAIST)).

References

[1] AJ Creely et al. "Overview of the SPARC tokamak". In: *Journal of Plasma Physics* 86.5 (2020), p. 865860502.

[2] Ionuţ-Gabriel Farcaş, Gabriele Merlo, and Frank Jenko. "A general framework for quantifying uncertainty at scale". In: *Communications Engineering* 1.1 (2022), p. 43.

[3] U. Giuliani et al. "Nuclear Fusion Impact on the Requirements of Power Infrastructure Assets in a Decarbonized Electricity System". In: *Fusion Engineering and Design* 192 (2023), p. 113554. ISSN: 0920-3796. DOI: 10.1016/j.fusengdes.2023.113554.

[4] Albert Gu, Karan Goel, and Christopher Ré. *Efficiently Modeling Long Sequences with Structured State Spaces*. Aug. 2022. DOI: 10.48550/arXiv.2111.00396. arXiv: 2111.00396 [cs]. (Visited on 09/28/2023).

[5] I.H. Hutchinson et al. "First Results from Alcator-C-MOD". In: *Physics of Plasmas* 1.5 (1994), pp. 1511–1518. ISSN: 1070-664X. DOI: 10.1063/1.870701.

[6] David M. Knigge et al. "Modelling Long Range Dependencies in ND: From Task-Specific to a General Purpose CNN". In: (2023). DOI: 10.48550/ARXIV.2301.10540. (Visited on 09/23/2023).

[7] L. L. Lao et al. "Reconstruction of Current Profile Parameters and Plasma Shapes in Tokamaks". In: *Nuclear Fusion* 25.11 (Nov. 1985), p. 1611. ISSN: 0029-5515. DOI: 10.1088/0029-5515/25/11/007. (Visited on 09/29/2023).

[8] Andrew D Maris et al. "The Impact of Disruptions on the Economics of a Tokamak Power Plant". In: *Fusion Science and Technology* (2023), pp. 1–17.

[9] A Murari and JET EFDA Contributors. "Automatic disruption classification based on manifold learning for real-time applications on JET". In: *Nuclear Fusion* 53.9 (2013), p. 093023.

[10] C. Rea et al. "A Real-Time Machine Learning-Based Disruption Predictor in DIII-D". In: *Nuclear Fusion* 59.9 (July 2019), p. 096016. ISSN: 0029-5515. DOI: 10.1088/1741-4326/ab28bf. URL: https://dx.doi.org/10.1088/1741-4326/ab28bf (visited on 09/22/2023).

[11] David W. Romero et al. *FlexConv: Continuous Kernel Convolutions with Differentiable Kernel Sizes*. Mar. 2022. DOI: 10.48550/arXiv.2110.08059. arXiv: 2110.08059 [cs]. (Visited on 09/29/2023).

[12] Lucas Spangher, J Scott Vitter, and Ryan Umstattd. "Characterizing fusion market entry via an agent-based power plant fleet model". In: *Energy Strategy Reviews* 26 (2019), p. 100404.

[13] Yi Tay et al. "Long Range Arena: A Benchmark for Efficient Transformers". In: *ArXiv* (Nov. 2020). (Visited on 09/28/2023).

[14] J. X. Zhu et al. "Hybrid Deep-Learning Architecture for General Disruption Prediction across Multiple Tokamaks". In: *Nuclear Fusion* 61.2 (Dec. 2020), p. 026007. ISSN: 0029-5515. DOI: 10.1088/1741-4326/abc664. (Visited on 09/22/2023).

Appendix

4.1 Background on Nuclear Fusion

There are two main techniques to confine fusion plasmas, both of which are currently in experimental / laboratory development. The first, is *inertial confinement*, using highly energetic lasers to compress and heat a solid target of fuel, or fast moving projectiles. and *magnetic confinement*, which leverages magnetic fields to confine and heat the ionized gas of fuel. While various designs are under exploration for both methods, our group focuses on magnetic confinement using tokamaks.

Tokamaks are designed to leverage axisymmetric magnetic-fields, and so they maintain stability under specific conditions. Stability is defined as the plasma's maintaining shape, current, and magnetic fields, and most importantly, manageability with respect to external controls. However, tokamak plasmas may become instable due to a heterogenous range of causes. For instance, if most electrons lag in toroidal rotations compared to poloidal ones, it can induce magnetic forces that unevenly pressurize the plasma. If unchecked, such imbalances can escalate, leading to plasma confinement loss, which leads to large temperature and current quenches deposited on the devices' walls. To name a few other types of instabilities, the plasma may experience vertical displacement events, unexpected material residue deposited on the devices' plasma facing components, or kinks in the toroidal oscillations. Thus, a disruption predictor must be general enough to cover a range of behavior. These predictors may be essential in proving commercial viability of the devices[12].

4.2 Data summary

Feature	Definition	Relevant instab.
Locked mode indicator	Locked mode mag. field normalized to toroidal field	MHD
Rotating mode indicator	Std. dev. of Mirnov array normalized by toroidal field	MHD
β_p	Plasma pressure normalized by poloidal magnetic field	MHD
ℓ_i	Normalized plasma internal inductance	MHD
q_{95}	Safety factor at 95th flux surface	MHD
n/n_G	Electron density normalized by Greenwald density	Dens. limit
Δz_{center}	Vertical position error of plasma current centroid	Vert. Stab.
Δz_{lower}	Gap between plasma and lower divertor	Shaping
κ	Plasma elongation	Shaping
$P_{\text{rad}}/P_{\text{input}}$	Radiated power normalized by input power	Impurities
$I_{\text{p,error}}/I_{\text{p,prog}}$	Plasma current error normalized by programmed current	Impurities
V_{loop}	Toroidal "loop" voltage	Impurities

Table 2: The input features of the model, their definitions, and a categorization of the type of instability the signal indicates.

4.3 Learned Filters

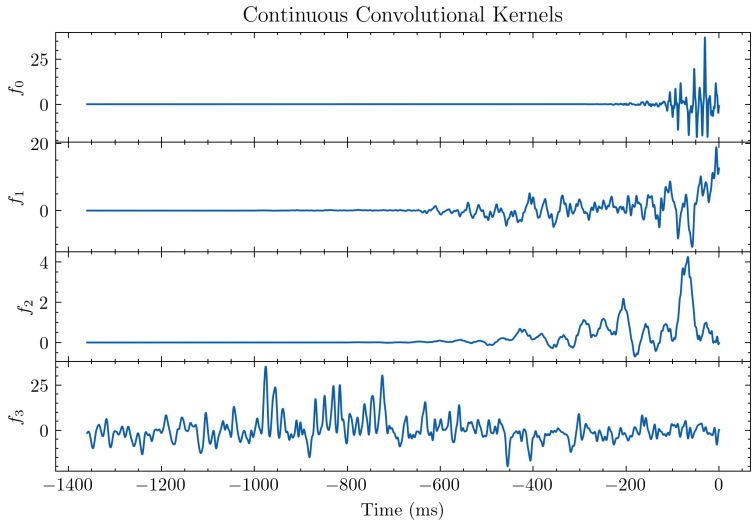


Figure 4: The continuous kernels learned in the order they appear in the model. Notice that f_0 is a short, jagged kernel that appears to mimic a derivative. We hypothesize thus that feature f_0 refers to the loop voltage feature, which becomes increasingly chaotic near disruptions but is stable for non-disruptions. The higher filters are more abstract combinations of the previous, so they do not map cleanly onto input features. We have generally seen similar behavior from f_0 throughout

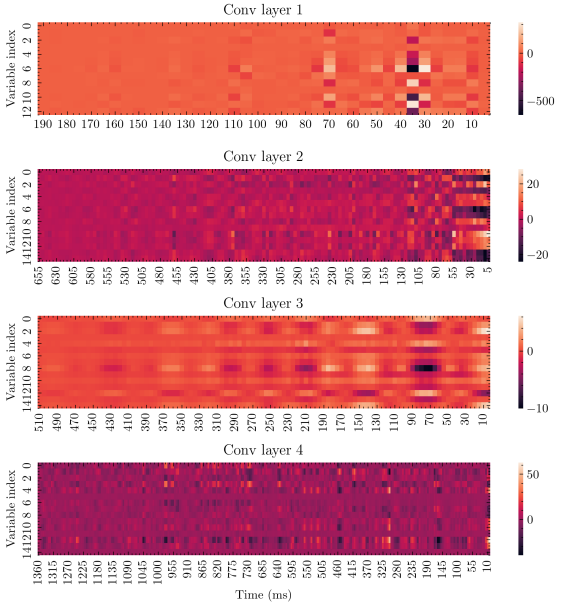


Figure 5: The discretized forms of the kernels. They are quite long: all over 190ms, and most over 500. The first filter takes the form of a derivative with it's main feature being a single large bump across one timestep. The other filters have much more complicated, long term structure.