Time Series Viewmakers for Robust Disruption Prediction

Dhruva Chayapathy
Alpharetta High School
dhruva.chayapathy@gmail.com

Tavis Siebert
UC Berkeley
tsiebert@berkeley.edu

Lucas Spangher
Google Research
spangher@google.com

Akshata Kishore Moharir
Microsoft
akshatan@terpmail.umd.edu

Om Manoj Patil
Alpharetta High School
om1.patil.om@gmail.com

Cristina Rea
MIT Plasma Science and Fusion Center
crea@psfc.mit.edu

Abstract

Machine Learning guided data augmentation may support the development of technologies in the physical sciences, such as nuclear fusion tokamaks. Here we endeavor to study the problem of detecting disruptions — i.e. plasma instabilities that can cause significant damages, impairing the reliability and efficiency required for their real world viability. Machine learning (ML) prediction models have shown promise in detecting disruptions for specific tokamaks, but they often struggle in generalizing to the diverse characteristics and dynamics of different machines. This limits the effectiveness of ML models across different tokamak designs and operating conditions, which is a critical barrier to scaling fusion technology. Given the success of data augmentation in improving model robustness and generalizability in other fields, this study explores the use of a novel time series viewmaker network to generate diverse augmentations or "views" of training data. Our results show that incorporating views during training improves AUC and F2 scores on DisruptionBench tasks compared to standard or no augmentations. This approach represents a promising step towards developing more broadly applicable ML models for disruption avoidance, which is essential for advancing fusion technology and, ultimately, addressing climate change through reliable and sustainable energy production.

1 Introduction

One of the most current notable efforts to get an idea from physical sciences into commercializable reality is nuclear fusion. Nuclear fusion has long been considered a "holy grail" in providing carbon-free energy without significant waste or land demands [Giuliani et al., 2023, Spangher et al., 2019]. A leading approach to generating fusion in laboratory plasmas is *magnetic confinement* via tokamak machines. However, the commercial viability of tokamaks hinges on developing their capability to accurately predict plasma disruptions: losses in plasma stability which may damage a tokamak and take it offline for months at a time.

Disruption are difficult to predict with first principles or physics-only models. Thus, a significant focus within the nuclear fusion community has been on the development of machine learning (ML)

Tackling Climate Change with Machine Learning: workshop at NeurIPS 2022.

models for disruption prediction. ML models are already in production at major tokamaks, including random forest predictors and simple neural networks [Rea et al., 2019, Hu et al., 2021, Pautasso et al., 2002], and research into various deep learning architectures has established the state of the art in disruption prediction capabilities [Zhu et al., 2020, Arnold et al., 2023, Spangher et al., 2023a, Zhu et al., 2020]. However, because the characteristics and physical dynamics of a tokamak can change dramatically depending on the machine's design [Granetz et al., 2018], and over time as the machine is upgraded [Kallenbach et al., 2017] or modified [Greenwald et al., 2014], it is imperative that prediction models can generalize to these different sources of data.

Data augmentation is one way to improve model generalizability [Iwana and Uchida, 2021] and robustness [Hong and Travis, 2022]. Since tokamak data is comprised of multivariate time series, common augmentation strategies can include jittering, slicing, and warping [Um et al., 2017, Wen and Keyes, 2019]. However, naively applying these techniques can lead to augmentations that fail to mimic plausible plasma behavior, impairing the model's ability to learn patterns relevant for disruption prediction [Fu et al., 2020]. Previously, a Stutent t process based approach has been used for tokamak data [Katharina et al., 2022], however it's direct impact on post-hoc models was not evaluated. In this work, we explore the use of "Viewmaker Networks" [Tamkin et al., 2021], models which learn to generate augmentations through adversarial learning. We hypothesize that viewmaker augmentations, or "views", are a solution for incorporating realistic yet diverse augmentations of disruption data which can improve the robustness of models toward disruptive discharges, and we demonstrate that using views in training which improve model AUC and F2 on DisruptionBench [Spangher et al., 2024] tasks over multiple disruption prediction models when compared to training on standard augmentations such as those present in the tsaug library or training without augmentation.

The novelty of our paper is as follows:
(1) We present an original adaptation of the image-based viewmaker network for time series augmentation and benchmark against other time series augmentation techniques.
(2) We are the first to rigorously test generative augmentation methods in the context of nuclear fusion datasets.

2 Background

Disruption benchmarks: DisruptionBench [Spangher et al., 2024] is the first disruption prediction benchmark aimed at assessing model generalizability across tokamaks. The data is comprised of roughly 30k trials (referred to here as "discharges") from three tokamaks: General Atomics' DIII-D, MIT's Alcator C-MOD, and the Chinese Academy of Science's EAST. Training and tests sets for each task can be generated by specifying the following parameters: (1) **New Machine**: One of three tokamaks is designated as the "new machine." The test set is comprised of only "new machine" discharges while the discharges in the training set are specified by the "data-split" parameter. (2) **Data-Split**: In this work, we focus on the following four cases with relation to two training machines and one test machine: Case 1, Zero-Shot; Case 2, Few-Shot; Case 3: Many-Shot. Case 4: Train and test from a single machine. For details on how true and false positives are calculated in DisruptionBench, please refer to Appendix B.1

Dataset provenance and preparation: Our dataset is open[1] [Zhu et al., 2020]. Each tokamak discharge lasts from seconds to minutes, generating gigabytes of multi-modal data, of which we focus on discharges' global state variables listed in Table 2. The tokamaks may be generally characterized by the information in Table 3, which demonstrates significant difference in average discharge length, sampling rate, and number of samples.

Each discharge is of length n, and we discard discharges that are shorter than 125ms. Each tokamak's diagnostics store metrics from each discharge at a different sampling rate. We normalize across tokamaks by discretizing at 5ms and interpolating via forward-fill. Roughly 10% of discharges end in disruptions. We truncate our time series to $X = 1, ..., n - \nu$, where n is the full length and $\nu = 40ms$, the minimum amount of time for a disruption mitigation system to activate. For a description of the features chosen, please see Table 2.

Viewmaker networks: Viewmaker Networks [Tamkin et al., 2021] are one approach to domain-agnostic contrastive learning. They adversarially train an encoder network and a generative network

[1] Please find our data at the following url: `https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/XIOHW1`.

(viewmaker) similar to a Generative Adversarial Network (GAN) set-up [Goodfellow et al., 2014] while using a SimCLR loss [Chen et al., 2020a]. For a more complete description of Viewmakers, please see Appendix A.

3 Models and methods

Time series viewmakers: Inspired by series decomposition blocks in [Wu et al., 2022], we first decompose our time series $X \in \mathbb{R}^{T\times d}$ into trend-cycle and seasonal components: $X_t = \mathrm{AvgPool}(\mathrm{Pad}(X))$, $X_s = X - X_t$. Because we decompose the original time series, (i.e. $X = X_s + X_t$) we use a separate generator to perturb each component. The perturbations are then combined, smoothed, constrained, and added to the original time series. See Figure 1 for an illustration of time series viewmakers.

We find the selection of generator networks V_t and V_s to be critical to performance. It is important that these generators generate **stochastic** and **challenging** perturbations. To better capture long-term temporal dependencies in the data, we change the viewmaker/generator from residual blocks to an LSTM-Transformer based model similar to the LSTMFormer detailed later. Noise is concatenated between each transformer layer to ensure stochastic perturbations. Furthermore, we adopt an adversarial SimCLR loss and a distortion budget $\epsilon = 0.1$. We trained our viewmaker for 10000 steps with a loss weight of 2.5 and a loss temperature of 0.05339. However, we encourage others to experiment with different loss functions, distortion budgets, generator networks, and encoders.[2]

Tested models: For benchmarking the effectivness of these augmentations, we experiment with 3 post-hoc models: LSTMFormer, FCN, and GPT-2. Compared to transformers, the sequential processing of LSTMs makes them particularly effective in encoding temporal relations within sequences. Therefore, the LSTMFormer first uses an LSTM layer to generate time-conscious positional encodings, similar to [Zeyer et al., 2019, Lim et al., 2020]. The output of the LSTM layer is then propagated through a normalization layer [Ba et al., 2016], a four-block transformer encoder [Vaswani et al., 2017], and a classification head. The FCN is a convolutional neural network variant first introduced in [Wang et al., 2016]. GPT-2 introduced for disruption prediction in [Spangher et al., 2023b] is an autoregressive transformer decoder model which uses a pretraining and a curriculum training scheme.

4 Results and discussion

Similar to the DisruptionBench paper [Spangher et al., 2024], we use AUC and F2 score to assess the performance of each augmentation strategy on the four cases specified in Section 2, setting C-MOD as the new machine (see Figure 3 for an explanation of this choice). F2 score is used since false negatives are more costly in the case of disruption avoidance. The results are summarized in Table 1.

Better disruptive shot prediction: As shown in Table 1, the use of viewmaker views resulted in the highest average AUC scores across all post-hoc models with an improvement of 0.14% for LSTMFormer, 2.3% for FCN, and 1.17% and for GPT-2. Views also contributed to the highest mean F2 score for FCN and GPT-2 with increases of 0.97% and 4.02% respectively. We also note that for case 3 (many shot case) we see a average improvement of 7.6% in F2, and for case 2 (few shot case) we see a significant improvement of 18.1%. On the other hand, tsaug seems to lag behind the viewmaker on average and actually drops mean F2 substantially for LSTMFormer (-18.23%) and FCN (-19.55%).

Learning disruptive features: The results on cases 2 and 3 in particular suggest that viewmakers help models learn underlying disruptive features even with limited samples. To understand how a model trained on views might behave in a production system, we plot a confusion matrix using one sampled shot from each category: TP, FP, TN, FN (Figure 5). It seems that features learned from views can simultaneously help models detect disruptions faster as well as overfit less to false disruptive cues compared to naive or no augmentations.

In-domain augmentations: We hypothesize that learning of disruptive features is a product of viewmakers' ability to generate more physically-realistic augmentations of the original data. To explore this further, we compared the dynamic time warping (DTW) similarity of 250 randomly

[2]Our code is public and may be found at the following url: `https://github.com/utterwqlnut/plasma-data-augmentation.git`

Table 1: DisruptionBench results for LSTMFormer, FCN, and GPT-2 trained on using no augmentations, tsaug standard augmentations, and viewmaker augmentations during training. "New Machine" is C-MOD in each case. Refer to Section 2 for further descriptions of each case

		LSTMFormer		FCN		GPT-2	
	Aug Type	AUC	F2	AUC	F2	AUC	F2
Case 1: Zero-Shot	None	0.571	**0.499**	0.518	**0.505**	0.575	0.525
	TS	0.571	0.358	**0.561**	0.166	**0.592**	**0.535**
	Viewmaker	**0.609**	0.342	0.546	0.503	0.527	0.404
Case 2: Few-Shot	None	**0.592**	0.389	0.631	**0.513**	0.695	0.523
	TS	0.537	0.189	0.579	0.413	0.723	0.616
	Viewmaker	0.568	**0.545**	**0.637**	0.494	**0.731**	**0.628**
Case 3: All machines	None	0.808	0.708	0.847	0.758	0.711	0.524
	TS	0.769	0.641	0.810	0.72	0.74	0.58
	Viewmaker	**0.823**	**0.735**	**0.853**	**0.777**	**0.754**	**0.611**
Case 4: Only CMOD	None	**0.792**	**0.707**	0.791	0.7	0.742	0.617
	TS	0.79	0.696	0.781	0.691	0.675	0.512
	Viewmaker	0.766	0.669	**0.814**	**0.725**	**0.744**	**0.632**
Mean	None	0.691	**0.576**	0.697	0.619	0.681	0.547
	TS	0.667	0.471	0.683	0.498	0.683	0.561
	Viewmaker	**0.692**	0.573	**0.713**	**0.625**	**0.689**	**0.569**

sampled disruptive discharges from each machine against their views and tsaugs (see Figure 4). For this sample, the average DTW for views was 409.14 while for tsaugs it was 770.39, indicating that views are much more faithful to the original data.

5 Limitations

Throughout the different post hoc models, training with viewmaker augmentations can cause loss in precision: -2.3% in FCN, and -9.4% in GPT-2; however, for LSTMFormer we see a +5.2% improvement (see Table 4). The similarity of many disruptive and non-disruptive shots could be one reason for this tradeoff (see Figure 6) while class imbalance might be another. While ideally recall and precision would both increase, for disruption avoidance control the cost of false negatives significantly outweigh the costs of false positives. Finally, our access to compute was limited, so we did not have the ability to present runs on multiple seeds, which would strengthen the statistical significance of our results and allow us to test different combinations of data-splits and new machines.

6 Conclusion and future work

Avenues for future work can include modifying viewmaker features, integrating the trained encoder into a new classifier, and comparing viewmakers to other time series generators such as TimeGAN [Yoon et al., 2019]. Additionally, more post-hoc models should be tested (see [Rea et al., 2019, Hu et al., 2021, Pautasso et al., 2002, Zhu et al., 2020]), and new tokamaks should be added to DisruptionBench.

In this work, we propose a novel time series variant of viewmaker networks and investigate its utility in generating data augmentations or "views" of tokamak discharge time series to improve model robustness for nuclear fusion disruption prediction and disruption avoidance control. Our findings suggest that using viewmaker views improve model AUC and F2 across a diverse set prediction tasks compared to naive or no augmentations, paritcularly in few and many shot cases. By improving the robustness of these models, we move closer to making tokamaks a commercial and tangible realization of decades of work in the physical sciences.

References

U. Giuliani, S. Grazian, P. Alotto, M. Agostini, C. Bustreo, and G. Zollino. Nuclear Fusion impact on the requirements of power infrastructure assets in a decarbonized electricity system. *Fusion Engineering and Design*, 192:113554, 2023. ISSN 0920-3796. doi: 10.1016/j.fusengdes.2023.113554.

Lucas Spangher, J Scott Vitter, and Ryan Umstattd. Characterizing fusion market entry via an agent-based power plant fleet model. *Energy Strategy Reviews*, 26:100404, 2019. doi: 10.1016/j.esr.2019.100404.

Christina Rea, KJ Montes, KG Erickson, RS Granetz, and RA Tinguely. A real-time machine learning-based disruption predictor in DIII-D. *Nuclear Fusion*, 59(9):096016, 2019. doi: 10.1088/1741-4326/ab28bf/.

WH Hu, Cristina Rea, QP Yuan, KG Erickson, DL Chen, Biao Shen, Yao Huang, JY Xiao, JJ Chen, YM Duan, et al. Real-time prediction of high-density east disruptions using random forest. *Nuclear Fusion*, 61(6):066034, 2021. doi: 10.1088/1741-4326/abf74d.

G Pautasso, Ch Tichmann, S Egorov, T Zehetbauer, O Gruber, M Maraschek, K-F Mast, V Mertens, I Perchermeier, G Raupp, et al. On-line prediction and mitigation of disruptions in asdex upgrade. *Nuclear Fusion*, 42(1):100, 2002. doi: 10.1088/0029-5515/42/1/314.

J. X. Zhu, C. Rea, K. Montes, R. S. Granetz, R. Sweeney, and R. A. Tinguely. Hybrid deep-learning architecture for general disruption prediction across multiple tokamaks. *Nuclear Fusion*, 61(2):026007, December 2020. ISSN 0029-5515. doi: 10.1088/1741-4326/abc664.

William F Arnold, Lucas Spangher, and Christina Rea. Continuous convolutional neural networks for disruption prediction in nuclear fusion plasmas, 2023.

Lucas Spangher, William Arnold, Alexander Spangher, Andrew Maris, and Cristina Rea. Autoregressive transformers for disruption prediction in nuclear fusion plasmas, 2023a.

RS Granetz, C Rea, K Montes, R Tinguely, N Eidietis, O Meneghini, DL Chen, B Shen, BJ Xiao, K ERICKSON, et al. Machine learning for disruption warning on Alcator C-Mod, DIII-D, and EAST tokamaks. In *Proc. 27th IAEA Fusion Energy Conference, IAEA, Vienna*, 2018. doi: 10.1088/1741-4326/ab1df4.

A Kallenbach, ASDEX Upgrade Team, EUROfusion MST1 Team, et al. Overview of asdex upgrade results. *Nuclear Fusion*, 57(10):102015, 2017. doi: 10.1088/1741-4326/aa64f6.

Martin Greenwald, A Bader, S Baek, M Bakhtiari, He Barnard, W Beck, W Bergerson, I Bespamyatnov, P Bonoli, D Brower, et al. 20 years of research on the alcator c-mod tokamak. *Physics of Plasmas*, 21(11), 2014.

Brian Kenji Iwana and Seiichi Uchida. An empirical survey of data augmentation for time series classification with neural networks. *PLOS ONE*, 16(7):e0254841, July 2021. ISSN 1932-6203. doi: 10.1371/journal.pone.0254841. URL `http://dx.doi.org/10.1371/journal.pone.0254841`.

Yang Hong and Desell Travis. Robust augmentation for multivariate time series classification. 2022. URL `https://doi.org/10.48550/arXiv.2201.11739`.

Terry T. Um, Franz M. J. Pfister, Daniel Pichler, Satoshi Endo, Muriel Lang, Sandra Hirche, Urban Fietzek, and Dana Kulić. Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, ICMI '17. ACM, November 2017. doi: 10.1145/3136755.3136817. URL `http://dx.doi.org/10.1145/3136755.3136817`.

Tailai Wen and Roy Keyes. Time series anomaly detection using convolutional neural networks and transfer learning, 2019.

Biying Fu, Florian Kirchbuchner, and Arjan Kuijper. Data augmentation for time series: traditional vs generative models on capacitive proximity time series. In *Proceedings of the 13th ACM international conference on pervasive technologies related to assistive environments*, pages 1–10, 2020.

Rath Katharina, Rügamer David, Bischl Bernd, von Toussaint Udo, Rea Cristina, Maris Andrew, Granetz Robert, and Albert Christopher. Data augmentation for disruption prediction via robust surrogate models. 2022. URL `https://doi.org/10.1017/S0022377822000769`.

Alex Tamkin, Mike Wu, and Noah Goodman. Viewmaker networks: Learning views for unsupervised representation learning, 2021.

Lucas Spangher, Matteo Bonotto, William Arnold, Dhruva Chayapathy, Tommaso Gallingani, Alexander Spangher, Francesco Cannarile, Daniele Bigoni, Eliana De Marchi, and Cristina Rea. Disruptionbench: A robust benchmarking framework for machine learning-driven disruption prediction. 2024.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020a.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting, 2022.

Albert Zeyer, Parnia Bahar, Kazuki Irie, Ralf Schlüter, and Hermann Ney. A comparison of transformer and lstm encoder decoder models for asr. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 8–15, 2019. doi: 10.1109/ASRU46091.2019.9004025.

Bryan Lim, Sercan O. Arik, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting, 2020.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline, 2016. URL `https://arxiv.org/abs/1611.06455`.

Lucas Spangher, Arnold William, Spangher Alexande, Maris Andrew, and Rea Cristina. Autoregressive transformers for disruption prediction in nuclear fusion plasmas. 2023b. URL `https://doi.org/10.48550/arXiv.2401.00051`.

Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar. Time-series generative adversarial networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper_files/paper/2019/file/c9efe5f26cd17ba6216bbe2a7d26d490-Paper.pdf`.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views, 2019.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, 2020.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning, 2020b.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings, 2022.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series, 2022.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

Matthias Minderer, Olivier Bachem, Neil Houlsby, and Michael Tschannen. Automatic shortcut removal for self-supervised representation learning, 2020.

T. C. Jernigan, L. A. Baylor, S. K. Combs, D. A. Humphreys, P. B. Parks, and J. C. Wesley. Massive gas injection systems for disruption mitigation on the diii-d tokamak. In *21st IEEE/NPS Symposium on Fusion Engineering SOFE 05*, pages 1–3, 2005. doi: 10.1109/FUSION.2005.252977.

S Jachmich, Uron Kruezi, Michael Lehnen, M Baruzzo, Larry R Baylor, D Carnevale, Dylan Craven, NW Eidietis, O Ficker, TE Gebhart, et al. Shattered pellet injection experiments at jet in support of the iter disruption mitigation system design. *Nuclear Fusion*, 62(2):026012, 2021. doi: 10.1088/1741-4326/ac3c86.

Zander Keith, Chirag Nagpal, Cristina Rea, and R.A. Tinguely. Risk-Aware Framework Development for Disruption Prediction: Alcator C-Mod and DIII-D Survival Analysis. 2024.

Martin Greenwald. Density limits in toroidal plasmas. *Plasma Physics and Controlled Fusion*, 44(8): R27, 2002.

A Viewmaker networks

A.1 Contrastive learning

Contrastive learning is a technique which aims to help models learn an embedding space in which views derived from the same input are grouped together while dissimilar views are pushed apart. The success of contrastive learning approaches has been extensively researched, particularly on computer vision benchmarks [Bachman et al., 2019, He et al., 2020, Chen et al., 2020a,b], but also in other domains [Gao et al., 2022, Yue et al., 2022]. While generating handcrafted views for tasks such as image classification have been refined, the process remains a handicap for domains such as time series.

SimCLR loss is one approach to contrastive learning proposed in [Chen et al., 2020a] for visual data. If given N pairs of views generated from the same input X, the SimCLR loss is defined to be

$$\begin{aligned} \mathcal{L} &= \frac{1}{2N}\sum_{k=1}^{N}[\ell(2k-1,2k)+\ell(2k,2k-1)], \\ \ell(i,j) &= -\log\frac{\exp(s(i,j))}{\sum_{k=1}^{2N}\mathbb{1}_{k\neq i}\exp(s(i,k))} \end{aligned} \tag{1}$$

Here, $s(i,j)$ is the cosine similarity of embeddings of views i and j.

A.2 Viewmaker components

To visually reinforce the brief description of viewmakers in Section 2, we encourage readers to refer to the top illustration in Figure 1, which gives a high-level overview of the components of a viewmaker network as described in [Tamkin et al., 2021]. The bottom illustration highlights the specific changes to the viewmaker created for this work which is discussed in Section 3.

In order to force the encoder to learn useful latent representations of the original data, the viewmaker network presents the encoder with stochastic perturbations of the original data projected onto an ℓ_p ball. Such projections have been studied in the context of adversarial robustness (e.g. in [Goodfellow et al., 2015]). The strength of the perturbations is further controlled by a distortion-budget: a hyperparameter which limits the radius of the ℓ_p ball to ensure that views don't become too difficult to learn useful representations from.

The advantage to using viewmakers over handcrafted augmentations is their ability to learn augmentations from the data directly. In the context of disruption prediction, viewmakers could help models pick up hidden trends and physical characteristics of disruptive discharges [Minderer et al., 2020] that are machine-agnostic: a critical step in the pursuit of more generalized models.

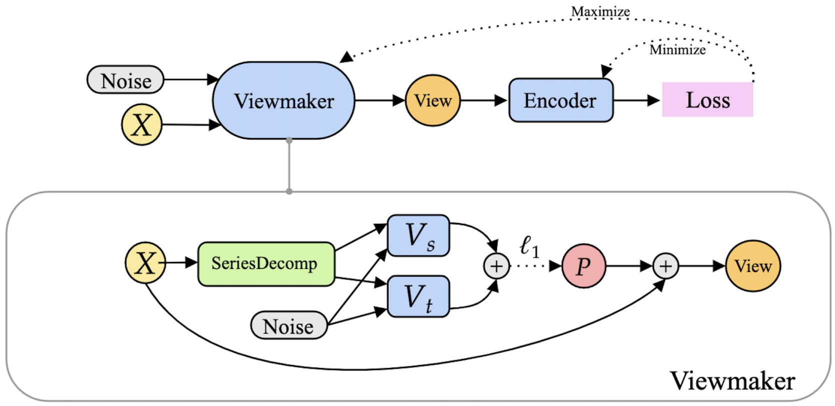

Figure 1: Overview of the time series viewmaker. V_t and V_s are generator networks which generate perturbed time series from X_t and X_s respectively.

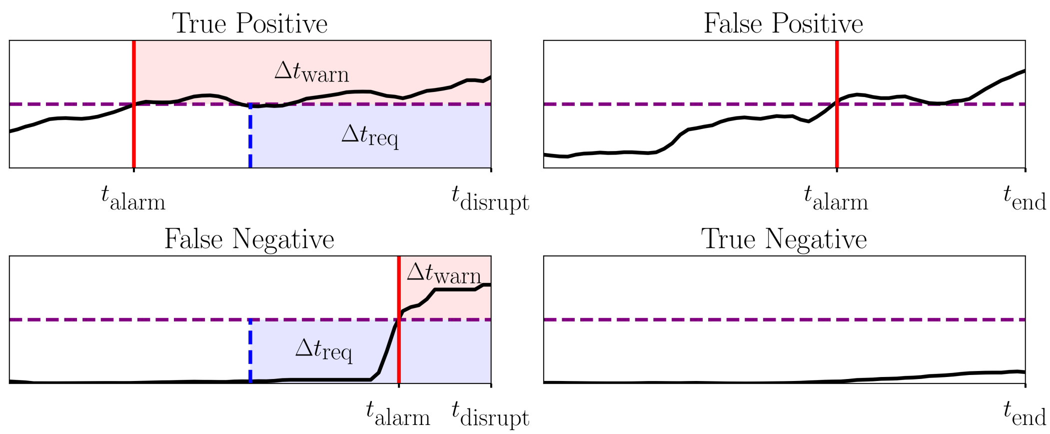

Figure 2: Illustration of the categorization of true and false positives, and true and false negatives. Figure adapted from ref. [Keith et al., 2024].

B Background (continued)

B.1 DisruptionBench

Designation of true or false positives in DisruptionBench is calculated similar to how a production-scale Disruption Mitigation System (DMS) would behave. The DMS needs at least some time to engage[3], Δt_{req}. Two thresholds $T_{low}, T_{high} \in (0, 1)$ and a hysteresis value h is set, and a prediction model outputs *disruptivity* scores, i.e. the probability that a disruption will happen in the future. If the disruptivity is above T_{high} for h time steps before Δt_{req}, the prediction is categorized as positive at time Δt_{warn}. Else, the discharge reaches its planned end and the model predicts a negative. If the disruptivity dips below T_{low}, the hysteresis counter is reset. For an illustration of how this translates into true and false positives and negatives, please see Figure 2.

C Results (continued)

[3]Two common methods of mitigating the harm from a disruption include: (1) puffing a heavy inert gas like Neon or Argon around the chamber's walls [Jernigan et al., 2005] or (2) shooting a frozen hydrogen pellet into the middle of the plasma reaction [Jachmich et al., 2021]. Both methods introduce particles that absorb the energy of the plasma.

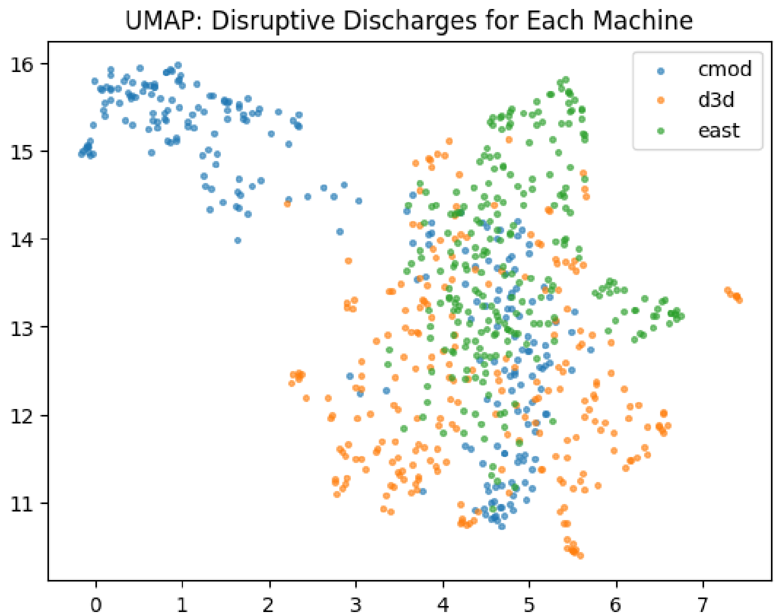

Figure 3: A UMAP clustering of disruptive discharges from each machine. Many discharges from C-MOD exhibit distinct behavior from those from DIII-D and EAST. Thus, to challenge models in their ability to learn general disruptive features, we designate C-MOD as the "new machine" parameter in our experiments.

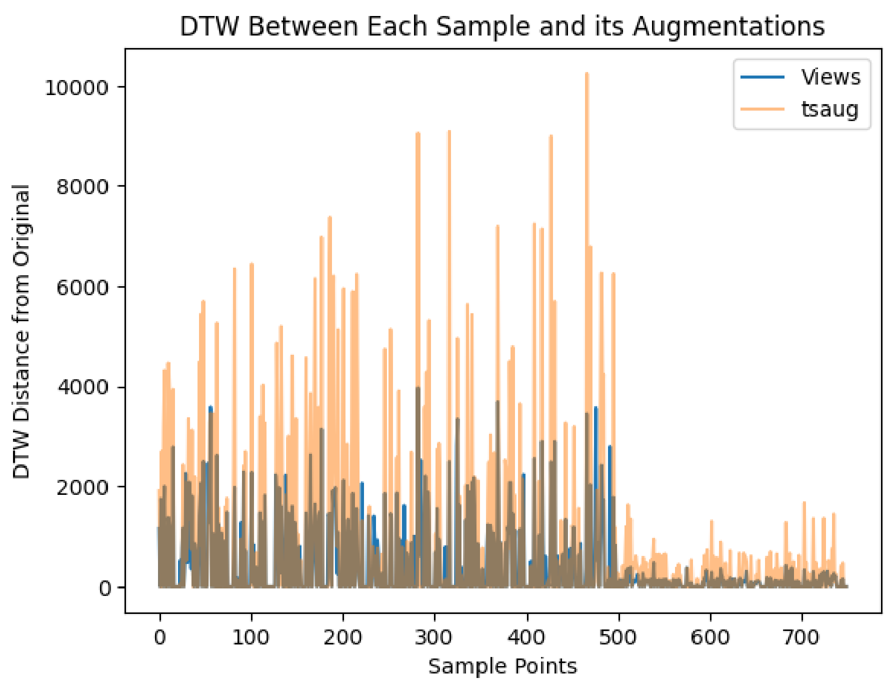

Figure 4: Comparison of dynamic time warping (DTW) similarities between disruptive discharges and their views (in cyan) and their tsaugs (in orange). Discharges were sampled evenly at random across each machine.

Table 2: The input features of the model, their definitions, and a categorization of the type of instability the signal indicates. The features listed are the same across all machines. While this table is adapted from [Zhu et al., 2020], some of our symbology, features, and feature descriptions are unique.

Feature	Definition	Relevant Instability	Units
β_p	Plasma pressure normalized by poloidal magnetic pressure	MHD	Unitless
ℓ_i	Normalized plasma internal inductance	MHD	Unitless
q_{95}	Safety factor at 95% of normalized flux surfaces	MHD	Unitless
$n = 1$ mode	$n = 1$ component of the perturbed magnetic field	MHD	T
n/n_G	Electron density normalized by Greenwald density limit [Greenwald, 2002]	Dens. limit	Unitless
Lower Gap	Gap between plasma and lower divertor	Shaping	m
κ	Plasma elongation	Shaping	Unitless
$I_{\text{p,error}}/I_{\text{p,prog}}$	Plasma current deviation from the programmed plasma trace (error), normalized by the programmed plasma trace (prog)	Impurities, hardware, runaway electrons	Unitless
V_{loop}	Toroidal "loop" voltage	Impurities	V
Tokamak Indicators	Three additional features with binary encodings of whether the data came from Alcator C-Mod, DIII-D, or EAST	n.a.	Unitless

Table 3: Metrics on a dataset composed of multiple tokamaks.

Tokamak	τ	Number of Discharges	Average Discharge Length	Initial Sampling Rate
C-Mod	50 ms	4000	0.52 s	0.005 ms
DIII-D	150 ms	8000	3.7 s	0.01 ms
EAST	400 ms	11000	5.3 s	0.025 ms

Table 4: Full DisruptionBench results (including Recall and Precision) for LSTMFormer, FCN, and GPT-2 trained on using no augmentations, tsaug standard augmentations, and viewmaker augmentations during training. "New Machine" is C-MOD in each case.

		LSTMFormer				FCN				GPT-2			
	Aug Type	AUC	Recall	Precision	F2	AUC	Recall	Precision	F2	AUC	Recall	Precision	F2
Case 1: Zero-Shot	None	0.571	**0.772**	0.207	**0.499**	0.518	**0.912**	0.181	**0.505**	0.575	**0.860**	0.205	0.525
	TS	0.571	0.404	0.247	0.358	**0.561**	0.140	**0.615**	0.166	**0.592**	0.860	**0.213**	**0.535**
	Viewmaker	**0.609**	0.333	**0.380**	0.342	0.546	0.842	0.193	0.503	0.527	0.561	0.190	0.404
Case 2: Few-Shot	None	**0.592**	0.439	0.269	0.389	0.631	**0.684**	0.257	**0.513**	0.695	0.561	**0.410**	0.523
	TS	0.537	0.175	**0.270**	0.189	0.579	0.509	0.236	0.413	0.723	0.789	0.328	0.616
	Viewmaker	0.568	**0.965**	0.199	**0.545**	**0.637**	0.614	**0.278**	0.494	**0.731**	**0.807**	0.333	**0.628**
Case 3: All machines	None	0.808	0.807	0.474	0.708	0.847	0.824	**0.573**	0.758	0.711	0.526	0.517	0.524
	TS	0.769	0.702	**0.476**	0.641	0.810	0.877	0.420	0.720	0.740	0.596	**0.523**	0.580
	Viewmaker	**0.823**	**0.877**	0.446	**0.735**	**0.853**	**0.930**	0.469	**0.777**	**0.754**	**0.649**	0.493	**0.611**
Case 4: Only CMOD	None	**0.792**	**0.912**	0.371	**0.707**	0.791	0.877	0.388	0.700	0.742	0.719	**0.394**	0.617
	TS	0.790	0.860	**0.395**	0.696	0.781	0.877	0.373	0.691	0.675	0.579	0.351	0.512
	Viewmaker	0.766	0.842	0.366	0.669	**0.814**	**0.877**	**0.427**	**0.725**	**0.744**	**0.772**	0.367	**0.632**
Mean	None	0.691	0.733	0.330	**0.576**	0.697	**0.824**	0.350	0.619	0.681	0.667	**0.382**	0.547
	TS	0.667	0.535	0.347	0.471	0.683	0.601	**0.411**	0.498	0.683	**0.706**	0.354	0.561
	Viewmaker	**0.692**	**0.754**	**0.347**	0.573	**0.713**	0.816	0.342	**0.625**	**0.689**	0.697	0.346	**0.569**

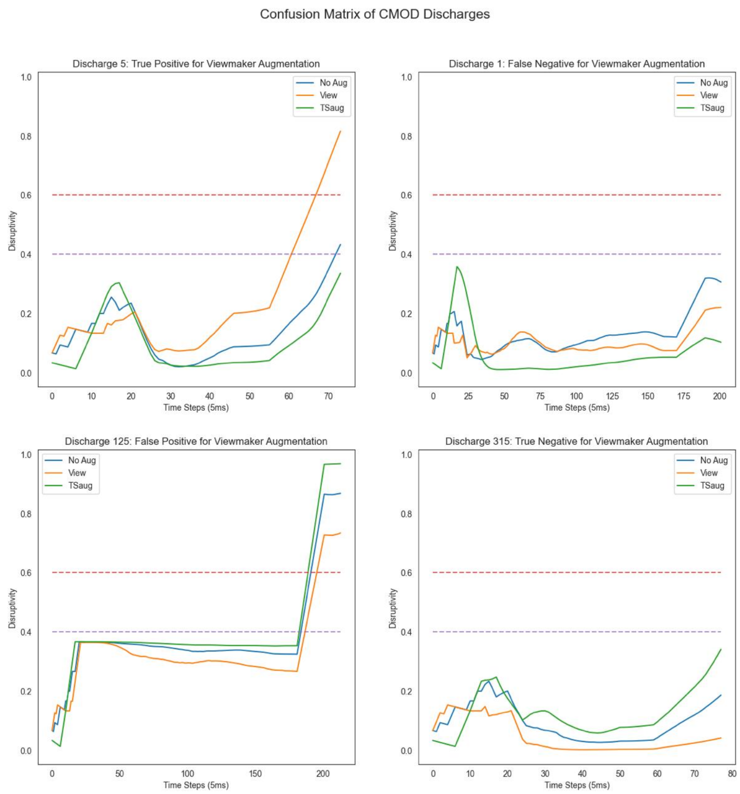

Figure 5: A comparison of four discharges from Alcator C-Mod for unrolled disruptivity on the test set, chosen based on outcomes of an LSTMFormer trained on the three augmentation strategies. While behaviors differ slightly, training on views seems to improve awareness of disruptive features, leading to faster disruption recognition and more confident true negative predictions.

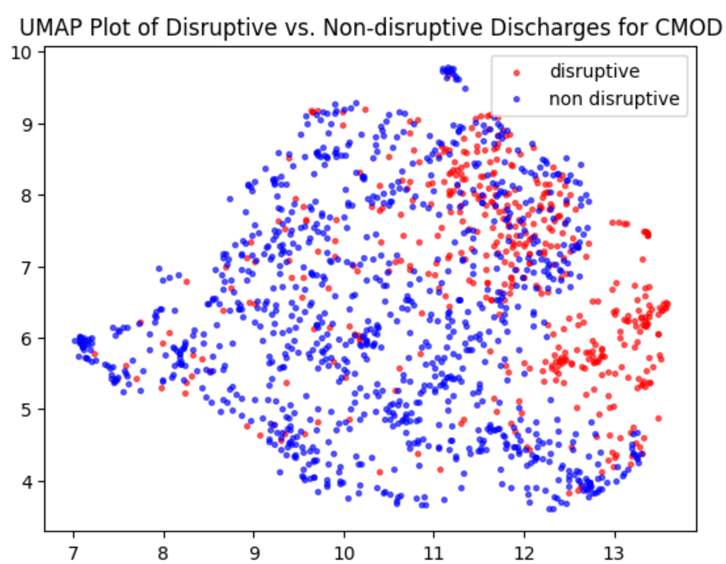

Figure 6: UMAP comparison of disruptive and non-disruptive discharges randomly sampled from Alcator C-MOD.