

Article

Performance Comparison of Machine Learning Disruption Predictors at JET

Enrico Aymerich ^{1,*}, Barbara Cannas ¹, Fabio Pisano ¹, Giuliana Sias ¹, Carlo Sozzi ², Chris Stuart ³, Pedro Carvalho ³, Alessandra Fanni ¹ and the JET Contributors [†]

¹ Department of Electrical and Electronic Engineering, University of Cagliari, 09123 Cagliari, Italy

² Istituto per la Scienza e Tecnologia dei Plasmi, Consiglio Nazionale Delle Ricerche, Via R. Cozzi, 53, 20125 Milano, Italy

³ UK Atomic Energy Authority, Culham Science Centre, Abingdon OX14 3DB, UK

* Correspondence: enrico.aymerich@unica.it; Tel.: +39-0706755871

† See the author list of J. Mailloux et al. 2022 Nucl. Fusion 62 042026.

Abstract: Reliable disruption prediction (DP) and disruption mitigation systems are considered unavoidable during international thermonuclear experimental reactor (ITER) operations and in the view of the next fusion reactors such as the DEMONstration Power Plant (DEMO) and China Fusion Engineering Test Reactor (CFETR). In the last two decades, a great number of DP systems have been developed using data-driven methods. The performance of the DP models has been improved over the years both for a more appropriate choice of diagnostics and input features and for the availability of increasingly powerful data-driven modelling techniques. However, a direct comparison among the proposals has not yet been conducted. Such a comparison is mandatory, at least for the same device, to learn lessons from all these efforts and finally choose the best set of diagnostic signals and the best modelling approach. A first effort towards this goal is made in this paper, where different DP models will be compared using the same performance indices and the same device. In particular, the performance of a conventional Multilayer Perceptron Neural Network (MLP-NN) model is compared with those of two more sophisticated models, based on Generative Topographic Mapping (GTM) and Convolutional Neural Networks (CNN), on the same real time diagnostic signals from several experiments at the JET tokamak. The most common performance indices have been used to compare the different DP models and the results are deeply discussed. The comparison confirms the soundness of all the investigated machine learning approaches and the chosen diagnostics, enables us to highlight the pros and cons of each model, and helps to consciously choose the approach that best matches with the plasma protection needs.

Keywords: tokamak; disruption prediction; machine learning; deep learning; data analysis; plasma diagnostics



Citation: Aymerich, E.; Cannas, B.; Pisano, F.; Sias, G.; Sozzi, C.; Stuart, C.; Carvalho, P.; Fanni, A.; the JET Contributors. Performance Comparison of Machine Learning Disruption Predictors at JET. *Appl. Sci.* **2023**, *13*, 2006. <https://doi.org/10.3390/app13032006>

Academic Editor: Xianpeng Wang

Received: 29 December 2022

Revised: 26 January 2023

Accepted: 1 February 2023

Published: 3 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Tokamak nuclear fusion devices suffer from several instabilities that may interact nonlinearly with each other, degenerating into the ultimate plasma destabilization and possibly into a disruption [1]. During a disruption, the plasma current drops to within a few milliseconds generating huge electromechanical and thermal forces, which may severely damage the plasma-facing components (PFC). Therefore, a strict requirement to develop a strategy for disruption prevention is needed, to control the plasma for the entire duration of the discharge.

Many efforts have been made in the last decades to identify the precursors, causes, and consequences of disruptions in tokamaks with the ultimate goal of developing automated schemes and strategies to mitigate or avoid disruptions. Both physics-based and data-driven disruption prediction (DP) models have been investigated. The physics-based methods have the advantage of being directly interpretable and more scalable among

different devices. However, the presently available first-principles DP models, even if they are able to describe the relevant physics of disruptions in tokamaks in great detail [2], are not sufficiently fast to be run in real-time. On the contrary, a lot of data-driven models have been proposed as a viable solution to the DP problem, developed for several tokamaks all over the world, such as the JET (EU) [3–9], AUG (DE) [10–12], DIII-D, NSTX, and c-Mod (USA) [13,14], EAST and J-TEXT (CN) [14,15], JT-60U (JP) [16], and ADITYA (IN) [17], many of which have been implemented and are currently operating in the control systems of the devices. This large number of proposals is due to the availability of both data from a lot of experimental campaigns performed in the different experimental fusion devices and of a lot of data-driven prediction approaches, such as statistical or machine learning methods [18].

Due to much larger dimensions than existing tokamaks, next generation devices, such as ITER and DEMO in Europe or CFETR in China, when operated at the full plasma current will not be able to withstand the stresses coming from unmitigated disruptions. Even if the performance of the proposed DP models seems quite good, no experimental data will be available on which to build data-driven models. For this reason, an integrated multi-machine approach is foreseen where existing facilities of different capabilities, sizes, and diagnostics will be used to provide DP models for these future fusion energy plants. Such an integrated approach needs to have different choices: (i) the choice of a common set of diagnostic signals, which are available in all the machines and in real time, and from which the best disruption prediction performance is obtained; (ii) the choice of the data-driven approach, better suited to the nature of the available data and to the disruption trigger to be provided; (iii) the choice of a common dataset, independent from the training set used to build the DP model, for testing the prediction performance; and (iv) the choice of common metrics to evaluate the model's performance, depending on the task, (avoidance, prevention, or mitigation).

As previously cited, in the literature quite a large plethora of data-driven methods have been proposed: statistical [12,19,20], machine learning [10,21–23] or, more recently, deep learning [24,25]. Among them, the most commonly used are traditional neural networks such as MLP-NN [4,10,11,15,22] or support vector machines (SVM) [5,6,23,26–28]), those belonging to manifold learning (self-organizing maps [29] and generative topographic mapping (GTM) [8,9,30]), decision trees [13,21,31] and deep neural networks [24,32–35]. Unfortunately, an accurate comparison of the different proposals in order to ultimately select the one that provides the best results for the specific application is prevented by the choice of different sets of diagnostic signals, different datasets for testing, even if the same device is considered, and different metrics to evaluate the performance of the models.

In this paper, a first step toward the implementation of this integrated approach is presented. A common database of diagnostic signals from the ITER-like Wall (ILW) JET campaigns is selected to build the training and validation sets with which three different machine learning (ML) DP models have been built: an MLP-NN, a GTM manifold learning model [9], and a convolutional deep learning neural network (CNN) model [33].

The performance of the three models has been compared using the same test experiments and the same performance metrics, namely the recall, the specificity, the precision, the confusion matrix, the receiver operating characteristics (ROC), and the area under the ROC curve (AUC), which are the most used metrics in binary classification problems. However, in the disruption prediction literature, the accuracy, the warning time distribution, and the assertion time provide a more immediate reading of the predictor's result, and they have been provided in the paper. An accurate analysis of the results of the comparison between predictors has allowed the pros and cons of DP models to be highlighted and to learn, from all the efforts of recent years, which of them are the most promising models for future devices.

The paper is organized into the following. Material and Methods in Section 2 reports, in Section 2.1, the background of the disruption prediction task. Section 2.2 reports the database of the JET diagnostics used to train, validate, and test the ML models, the feature

engineering performed to extract meaningful input information compatible with the architectures of such models, and the labelling of the discharge samples to associate these inputs to consistent outputs. In Section 2.3, the descriptions of the well-known machine learning methods used to develop the different comparison methods are briefly summarized whereas, in Section 2.4, the performance metrics used to perform the comparison are recalled. Section 3 details the implemented architectures of the DP models and their results, while Section 4 reports the results of the comparison and draws the conclusions, providing initial guidance on which diagnostics and models are most promising for the development of a cross-machine disruption prediction system and looking to the future for its integration into next-generation devices.

2. Materials and Methods

2.1. Background

In this paper, the performance of three ML DP models, which implement three different ML paradigms, have been compared using the Joint European Torus (JET) tokamak as the test bed. The three ML paradigms are the MLP-NN, GTM, and CNN. All the proposed ML methods perform a binary classification of an $n - D$ vector of the plasma feature x , measured at a given time t (time sample), into one class of a dependent variable y ($y = 1$, i.e., disrupted), or into the other ($y = 0$, i.e., non-disrupted).

It is well-known that tokamaks are machines that operate in a pulsed way, where each discharge presents different operational states during its evolution, depending on the chain of events, which possibly leads to the discharge to disrupt. In Figure 1, the sketch of the plasma current (I_p) in a disrupted discharge is shown. During the ramp-up phase, the plasma current is increased until it reaches the flat-top value. In a disruption, the stable phase is lost, and a chain of events follows, which will develop towards a thermal quench leading to the plasma disruption. During this unstable phase, some disruption precursors appear.

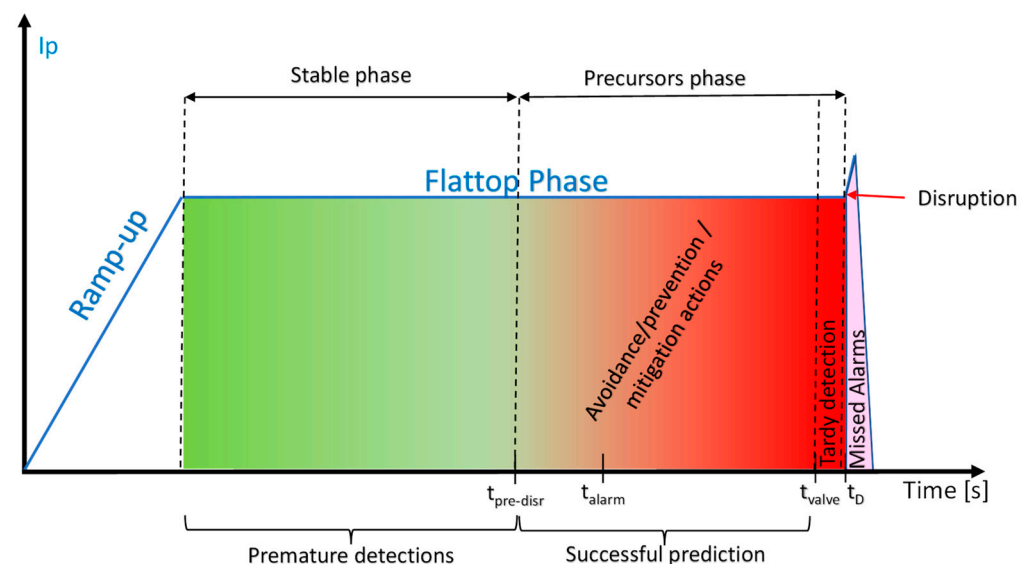


Figure 1. Sketch of the plasma current evolution in a disrupted discharge.

To train an ML DP model, examples of both disrupted and non-disrupted plasma states (discharge time samples) must be collected. To this end, for each disruption in the training set, it is mandatory to identify, as precisely as possible, the so-called $t_{pre-disr}$, which determines the beginning of the precursors phase. This task, far from being easy, has been solved in most of the literature, assuming the same value for all the disruptions on the basis of statistics or heuristics, inevitably introducing contradictory information in the prediction models. Only recently, some of these authors developed an algorithm to automatically

determine a consistent value of $t_{pre-disr}$ ($t_{pre-disr-AUT}$) for the different disruptions [9]. Once the value of $t_{pre-disr-AUT}$ is determined, the discharge sample, represented by the feature vector x , in the time window from $t_{pre-disr-AUT}$ to the disruption time t_D , is associated to the output label = 1 (i.e., as disrupted). Note that for disruptions mitigated by a massive gas injection (MGI), the time of the valve activation (t_{valve}) is considered in place of t_D . The samples of disrupted discharges, from the beginning of the flat-top to $t_{pre-disr-AUT}$, and all the flat-top samples in the regularly terminated discharges can be labelled as non-disrupted ($y = 0$).

The ML DP model is built by using a training set constituted by two instances $\{x, y\}$. Usually, a validation set, independent from training and test sets, has to be used to choose the best free parameters of the DP model. Once a new feature vector x , not belonging to the training set, is fed to the model, an output y in the range $[0, 1]$ is returned, which represents the likelihood of the vector belonging to a disrupted state. The time evolution of the output gives information on the possible evolution of the discharge towards a disruption. In Figure 2, the evolution of the disruptive likelihood (red) is reported for a disrupted pulse. The DP triggers an alarm when the disrupted likelihood exceeds a fixed threshold (horizontal dashed line) for at least an optimized assertion time $t_{assertion}$. In the same Figure 2, the vertical purple line and vertical black line indicate $t_{pre-disr-AUT}$ and the alarm time t_{alarm} , respectively.

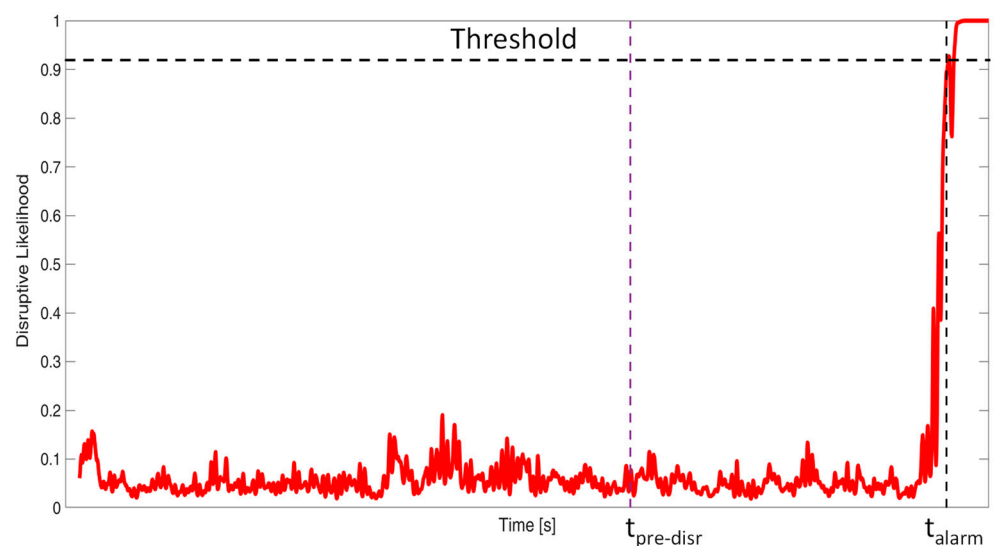


Figure 2. Disruptive likelihood evolutions in a disrupted discharge.

Depending on the alarm time, i.e., the resulting warning time ($\Delta t_{warning}$), which is the time interval between t_{alarm} and t_D , several interventions can be put in place, such as disruption active avoidance or mitigation (see Figure 1). In these cases, the DP answer is assumed as a successful prediction (SP). If the warning time is not even sufficient for mitigation, the prediction is classified as tardy detection (TD). At the JET, the minimum warning time is 10 ms, which is the time required for the massive gas injection system (MGS) to mitigate the discharge. A missed alarm (MA) occurs if the DP system does not trigger any alarm. Moreover, if the alarm is triggered before the appearance of the disruption precursors, the detection is considered premature (PRD). At the JET, a conclusive definition of premature alarms has not yet been established, so in the following, premature detections will not be counted.

Of course, the disrupted likelihood should remain below the threshold for the duration of a regularly terminated pulse, otherwise the prediction is classified as a false alarm (FA).

2.2. Database

The data from this study come from a database, created and maintained by the University of Cagliari, containing hundreds of disrupted and regularly terminated discharges from some of the JET experimental campaigns performed after the installation of the ITER-Like Wall (ILW) [9,33], from 2011 to 2020. Only the discharges with a flat-top plasma current higher than 1.5 MA, and a flat-top length greater than 200 ms, where all the diagnostics described in Table 1 were available and consistent, have been selected. Moreover, the discharges caused by a vertical displacement event and those in limiter configurations were excluded. Moreover, only disruptions occurring at the flat-top phase have been considered here. In total, the database for this work contains 193 disrupted and 219 regularly terminated discharges, which is the same as in [33]. The flat-top starting time has been assumed as the first time instant where the plasma is in an X-point configuration. Both 0-D and 1-D diagnostic signals have been collected to extract the feature vector x and are reported in Table 1. The literature demonstrated the benefit impact of the recent introduction of these 1-D plasma profiles [8,9,28,30,33,36,37]. The temperature and density profiles come from high-resolution Thompson scattering (HRTS), the radiated power profile comes from the horizontal lines of sight of the bolometer, the internal inductance comes from the code BetaLi for the estimation of the poloidal beta and internal inductance, while the locked mode amplitude comes from the saddle loops (LMS) and is normalized by the plasma current.

Table 1. Diagnostic signals, acronyms, and units.

Plasma Signal	Acronym	Diagnostics	Dimension
Electron temperature	T_e	HRTS	1-D
Electron density	n_e	HRTS	1-D
Radiated power	P_{rad}	Bolometer	1-D
Total radiated power	$P_{rad-TOT}$	Bolometer	0-D
Total input power	P_{TOT}	BetaLi	0-D
Internal inductance	l_i	BetaLi	0-D
Normalized locked mode	LM_{norm}	LMS	0-D

These diagnostic signals are, usually, preprocessed (data-reduced, filtered, sampled, normalized, and windowed) and a feature engineering process can follow to extract the features more suitable to the chosen DP architecture from the raw signals.

Table 2 reports the number of discharges and the originating campaigns for the training, validation, and test sets. Note that, as highlighted in [33], the training set contains experiments carried out in the early operation phase of the JET-ILW, and the operational space of the JET has changed in the following experimental campaigns. In fact, more recent JET-ILW experiments are run at a higher power and plasma current, causing higher disruption rates. This change may affect the disruption patterns with respect to those observed in the training set discharges [24,38]. This makes such test sets particularly challenging to evaluate the robustness of the disruption predictors at the change of the operating conditions as well as the aging effect, inherent in any data-driven predictor, and to estimate the suitability of the selected set of diagnostics. The comparison of the three ML DP models has been performed referring to the same diagnostic signals and the same training, validation, and test pulses reported in Table 2. Note that this is the number of experiments included in the dataset; each experiment consists of several multidimensional samples sampled with a high frequency (2 ms). Considering the training and validation data, the dataset is composed of more than 100,000 multidimensional samples, a sufficient number to train the models.

Table 2. Datasets for developing the models.

Sets	Disrupted	Non-Disrupted	JET Campaigns
Training set	63	54	2011–2013
Validation set	22	16	2011–2013
Test set	108	149	2011–2020

2.3. Machine Learning Methods

As previously cited, in this paper the performance of three different ML paradigms have been compared. In the following, the basics of the three approaches are summarized. For training these models, a workstation with 8 CPU cores and with an NVIDIA RTX 3060 GPU was used, and the training time was in the order of hours for the CNN. However, the inference time for the trained model is in the order of ms per sample and is compatible with a real-time application.

2.3.1. Multi-Layer Perceptron Neural Networks (MLP-NN)

An MLP-NN consists of layers of units, where the units in a layer are connected with all the units in the next layer. Referring to the binary classification of disrupted or non-disrupted samples, an MLP can model the non-linear relationship among the input feature vector x and the corresponding output y . Usually, a three-layer MLP-NN is used, where the input–output relationship is described by the following algebraic system:

$$\begin{cases} \text{Input Layer} & W_1 \cdot x + b_1 = g \\ \text{Hidden Layer} & h = f(g) \\ \text{Output Layer} & \sigma(W_2 \cdot h + b_2) = y \end{cases} \quad (1)$$

where W_1, b_1 and W_2, b_2 are the weights matrices and the biases of the input and output layers, respectively, g and h are the input and output vectors of the hidden layer, and $f(\cdot)$ and $\sigma(\cdot)$ are the hidden and output layer activation functions, respectively. For the classification task, σ is generally a SoftMax function while f is a sigmoid function.

During the training, the weights and biases are adjusted in order to minimize the prediction error, using a back-propagation algorithm [39]. In order to avoid overfitting, the training is stopped when the validation error begins to rise. Any numerical optimization algorithm could be used to optimize network performance, but the most applied use either the gradient of the performance or the Jacobian of the network errors with respect to the weights. Once the MLP-NN model is trained, the output related to a new discharge provides its disruptive likelihood.

2.3.2. Generative Topographic Mapping (GTM)

GTM [40] is a probabilistic manifold learning method, which allows us to embed a high-dimensional (n-D) space into a low-dimensional (typically 2-D), possibly nonlinear, latent space. The latter is a grid of K prototype nodes that have coordinates in both the embedded and input space. During the training, the algorithm performs mapping $f(z, W, \beta)$ from the set of training feature vectors $X = \{x_1, x_2, \dots, x_I\} \in \mathbb{R}^n$ into the prototypes $Z = \{z_1, z_2, \dots, z_K\} \in \mathbb{R}^2$ by linearly combining a radially symmetric Gaussian’s basis function (RBF) Φ , where W and β are the adaptive parameters of the model. The model parameters are optimized with the expectation maximization (EM) algorithm:

$$\max_{W, \beta} \mathcal{L} = \sum_{i=1}^I \ln \left(\frac{1}{K} \sum_{k=1}^K p(x_i | z_k, W, \beta) \right) \quad (2)$$

where

$$p(x_i | W, \beta) = \frac{1}{K} \sum_{k=1}^K \left(\frac{\beta}{2\pi} \right)^{-\frac{n}{2}} \cdot \exp \left\{ -\frac{\beta}{2} f(z_k, W) - x_i^2 \right\}$$

Once the model has been optimized, the corresponding posterior distribution over the latent space can be computed through the Bayes theorem referring to the prior distribution of the latent variable, where the asterisks indicate the optimal parameter values.

$$p(\mathbf{Z}_k|x_i) = \frac{p(x_i|\mathbf{Z}_k, \mathbf{W}^*, \beta^*) \cdot p(\mathbf{z}_k)}{\sum_{k'=1}^K p(x_i|\mathbf{Z}_{k'}, \mathbf{W}^*, \beta^*) \cdot p(\mathbf{Z}_{k'})} \tag{3}$$

To visualize the input space on the 2D map, statistic measures such as the mean or the mode, can be used to visualize the posterior probability distribution over the latent space. Note that the overfitting can be managed by setting the latent space properties, as suggested in [40].

2.3.3. Convolutional Neural Networks (CNN)

A CNN is a deep neural network particularly suited to processing images and is able to produce a high-accuracy performance without the need for handmade feature extraction engineering [41]. Its architecture contains a cascade of blocks that filter the input data to extract significant features and perform the classification task: (i) convolutional units (CU_k), each composed by the cascade of a convolutional layer (C_k), a batch normalization layer (N_k), and a non-linear activation layer with rectified linear unit (ReLU) functions (A_k); (ii) max-pooling P_{max} and/or average-pooling P_{avg} layers; (iii) drop-out layers (D); (iv) fully connected MLP (FC) layers; (v) soft-max layers (S); and (vi) the classification output layer (CO). The main advantage of this architecture, for DP tasks, is that 1D profile diagnostics can be directly used as network inputs without the necessity to synthesize 0D signals from them, such as peaking factors [8,36]. During the supervised training, the network parameters are optimized with stochastic gradient descent algorithms. Once the CNN model is trained, as in the case of the MLP-NN models, the output related to a new discharge provides its disruptive likelihood, as in Figure 2.

2.4. Performance Metrics

In binary classification, a true positive (TP) is counted if a positive instance is predicted as positive, whereas it is counted as a false negative (FN) if it is predicted as negative. A negative instance predicted as negative is defined as a true negative (TN), whereas it is counted as a false positive (FP) when predicted as positive. These four values can be summarized in a 2×2 confusion matrix, where each row contains the instances in the actual class whereas each column contains the instances in the predicted class.

Note that such definitions do not take into account the warning time $\Delta t_{warning}$ provided by the predictor to act on the plasma. However, they can be adapted to the disruption prediction definition definitions, introduced in Section 2.1., including tardy detections (TD) and missed alarms (MA) in the counting of a FN, and when evaluated, premature detections (PRD) in a FP. Thus, a direct correspondence between the two approaches for the performance evaluation can be found when the instance is the discharge. Note that a TP corresponds to an SP, and TNs are evaluated as the difference between negative instances N (number of non-disrupted discharges in the test set) and those counted as an FA. The positive instances are indicated as P (number of disrupted discharges in the test set).

Therefore, some performance indices can be used, valid for both the previous definitions:

- $PRECISION = \frac{TP}{TP+FP}$
- $RECALL = \frac{TP}{TP+FN} = \frac{TP}{P}$
- $SPECIFICITY = \frac{TN}{TN+FP} = \frac{TN}{N}$
- $ACCURACY = \frac{TP+TN}{P+N}$

In addition, the F-score indicators, which encompass the information of *PRECISION* and *RECALL*, can be defined as:

$$F_\alpha = \left(1 + \alpha^2\right) \cdot \frac{PRECISION \cdot RECALL}{\alpha^2 \cdot RECALL + PRECISION}$$

F_1 score is the harmonic mean between *PRECISION* and *RECALL*, whereas F_2 assigns a higher cost to the disrupted misclassifications.

For a binary classifier parametrized by a threshold, as in our case, the relative trade-off between benefits and costs can be displayed by the ROC curve, which draws the true positive rate $TPR = \frac{TP}{P}$ as a function of the false negative rate $FNR = \frac{FN}{P}$ by varying the threshold. Moreover, the area under the ROC curve (AUC) can also be used to assess the ability of the model to distinguish between the two classes.

However, in the disruption prediction literature, the most informative figure of merit is defined by the accumulated fraction of detected disruptions as a function of $\Delta t_{warning}$. It allows one to read, in a unique graph, the successful prediction and the tardy detections, as well as a general overview of the premature detections and the alarm anticipation times.

All these metrics will be presented in the following to compare the performance of the three DP models.

3. Models Implementation and Results

When training a data-driven algorithm, pre-processing of the input data is essential for the successful development of the model. The data from the HRTS, the electron temperature, and the density profiles have been pre-processed using a procedure based on the correlation of the measurements of each line of sight to those of their neighbours, and also by exploiting the estimated error measurement from the diagnostic [42]. The unreliable measurements can then be replaced by the interpolated values between the two closest lines of sight. Moreover, the outer nine lines of sight (major radius greater than 3.78 m) have been discarded, as they do not provide reliable measurements in the analyzed dataset. For the Bolometer 1D data, negative power values have been substituted with null values, whereas unreliable positive ones are saturated to 1 MW/m², which is a threshold empirically found. All the diagnostic data are resampled causally, which means using only current and past inputs, with a sampling time of 2 ms. Note that a causal resampling is necessary to develop algorithms in a real-time framework.

A data reduction is performed as described in [8] in order to represent the two classes (disrupted and non-disrupted) with the same percentage of training samples, and the repetition of discharges with the same or similar settings is avoided. In this way, the algorithm can be trained without up-weighting the errors on the disruptive samples.

Many ML models require to have, as an input, a plasma feature vector of n values corresponding to the time series, sampled at the same time t . In order to encode the spatial information contained in the 1-D plasma profiles, 0-D peaking factors of temperature (Te_{pf}) and plasma density (ne_{pf}) were defined as the ratio between the mean values of the measurements over the core region of the plasma cross-section divided by the mean value of the measurements of the outer region of the plasma. Similarly, the radiated power (Rad_{pf-CVA}) peaking factor is evaluated by dividing the mean values of the core measurements by the mean values of the outer channels, excluding the divertor lines of sight (channels 1–8), and instead $Rad_{pf-XDIV}$ is defined by averaging the contribution of the divertor channels and dividing it by the mean value of the remaining channels, excluding the core ones (channel 13–16) [8,36]. They are used here as plasma features of the MLP-NN and GTM models together with the corresponding samples of internal inductance l_i , and the radiated fraction of the total input power $P_{frac} = \frac{P_{rad-TOT}}{P_{TOT}}$. The normalized locked mode amplitude LM_{norm} signal is used in the MLP-NN model as a further element of the feature vector while it is used in the GTM-based prediction model as an external signal in the multiple condition alarm scheme proposed in [8].

Conversely, CNNs have the ability to learn spatiotemporal features directly from the plasma profiles, overcoming the limits of the previously cited 0-D peaking factors, such as some heuristic choices in the definition and the inevitable loss of information. Hence, in the CNN DP model [33], for each plasma profile, a spatiotemporal matrix is built, whose elements assume the value of the measure in the corresponding line of sight of the corresponding diagnostics and the corresponding time sample. The obtained images are

vertically stacked, normalized with respect to the signal ranges in the training set, and segmented using an overlapping sliding window of 200 ms, obtaining the corresponding feature vector elements. The 0-D signals l_i , P_{frac} , and LM_{norm} are also sampled in this model at the same sample frequency as the 1-D data.

3.1. MLP-NN Disruption Prediction Model

In Table 3, the training parameters of a single hidden layer MLP-NN model are reported. The parameter σ determines the change in the weight for the second derivative approximation, and the parameter λ regulates the indefiniteness of the Hessian. In Figure 3a, the input features are shown for a test disrupted discharge (#94218) belonging to a JET campaign temporally far from those used in the model training. The disruptive likelihood is reported in Figure 3b. An alarm is triggered when the disruptive likelihood remains above the optimized threshold for at least a number of samples, corresponding to an assertion time obtained by multiplying this number of samples by the sampling time equal to 2 ms. The assertion time can be introduced to avoid wrong alarms due to spikes in the disruptive likelihood. As reported in Table 3, the MLP-NN does not need an assertion time, which is then equal to 0 ms. This means that optimizing the threshold of the model is sufficient to make the MLP-NN response robust in terms of detecting the presence of disruption precursors, while keeping the number of false alarms low. The vertical dashed line in Figure 3b identifies the alarm time t_{alarm} , resulting in a warning time $\Delta t_{warning} = 408$ ms.

Table 3. MLP-NN training parameters.

Parameters	Value
Optimizer	Back Propagation and Scaled Conjugate Gradient algorithm [43]
Number of input neurons	7
Number of hidden neurons	10
Number of output neurons	1
Weights initialization	Random
Learning rate (σ, λ)	$(5 \cdot 10^{-5}, 5 \cdot 10^{-7})$
Best epoch	23
Validation stop (consecutive evaluations)	75
Assertion Time (ms)	0

In Table 4, the confusion matrix is reported (green: successful results; red: mistakes) together with the prediction performance indices. All the indices have very good values with an excellent balance between the correct predictions of the disrupted pulses and a very limited number of false alarms in the regularly terminated pulses. All these numbers overcome the results in the literature, e.g., [4], where an MLP was trained with only 0-D signals without the introduction of information, even if synthesized, from plasma profiles. The use of plasma profiles information really introduces a big benefit on the predictor performance.

Despite these very high-performance index values, and despite the extreme simplicity of the model architecture, the MLP-NNs suffer to be ‘black boxes’ models, which provide a good prediction but are very difficult to interpret. For this reason, other ML predictor architectures have been nominated in recent years to be those selected for future fusion devices.

Once the GTM model has been trained and successively colored, it can be used to track the dynamics of a new discharge by projecting the temporal sequence of its samples on the map. In Figure 4, the trajectory of the discharge is represented with a dashed line, the color of which darkens during the time evolution up to the tip of the arrow representing the end point.

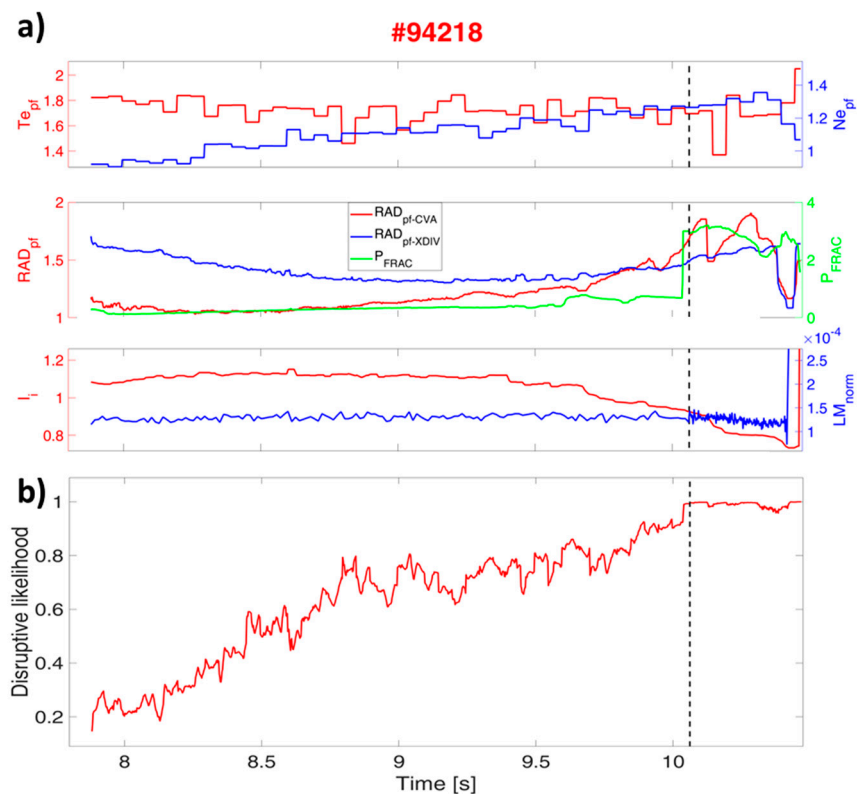


Figure 3. JET-disrupted discharge #94218: (a) Time evolution of the seven plasma dimensionless parameters: temperature ($T_{e_{pf}}$), plasma density ($n_{e_{pf}}$), and radiated power (Rad_{pf-CVA}), and ($Rad_{pf-XDIV}$) peaking factors, internal inductance l_i , radiated fraction of the total input power (P_{frac}), normalized locked mode amplitude (LM_{norm}) signal; (b) disruptive likelihood of the disrupted discharge #94218 supplied by MLP-NN. The dashed black line identifies the alarm time.

Table 4. Confusion matrix and performance indices of the MLP-NN prediction model evaluated on the test set.

		Predicted	
		P + N = 257	P = 133
Actual	P = 108	TP = 103	FN = 5
	N = 149	FP = 5	TN = 144
	PRECISION = 0.954	RECALL = 0.954	SPECIFICITY = 0.966
	ACCURACY = 0.961	$F_1 = 0.954$	$F_2 = 0.954$
Train	SP% = 100%	MA% = 0%	FA% = 1.85%
Test	SP% = 95.37%	MA% = 2.78%	FA% = 3.36%

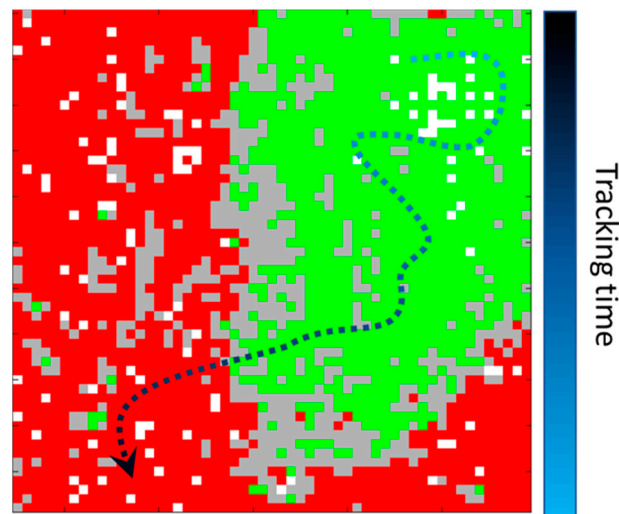


Figure 4. 2-D GTM map of a n-D space, colored on the basis of the unit composition, and trajectory of a disrupted pulse. The color bar shows the time evolution of the discharge, and its color darkens during the time up to the tip of the arrow representing the end point. All the synthesized features except LM_{norm} , used to train the MLP-NN, have been used also to train the GTM model.

Usually, a disrupted discharge evolves in the green region until disruption precursors appear, moving the trajectory towards the red disruptive region.

The disruptive likelihood of the discharge is obtained by evaluating the percentage of the disrupted samples contained in the units visited by the trajectory.

In our implementation, the validation set has not been used to evaluate the possibility of overfitting due to the huge computation time required. However, it has been instead joined to the training set to create and color the map.

The free parameters of the GTM model, reported in Table 5, have been optimized with a Tabu Search procedure [44]. In Table 5, the resulting GTM map composition is also reported. The obtained GTM map of the JET operational space is reported in Figure 5a, where the same disrupted pulse #94218 is tracked. The trajectory of the discharge firstly evolves within the green “safe” region and then enters the red disruptive region. The lighter points of the trajectory correspond to the beginning of the discharge, whereas the darker one corresponds to the end at the disruption time t_D . The corresponding disruptive likelihood is reported in Figure 5b. The vertical dashed line identifies the alarm time t_{alarm} .

Table 5. GTM training parameters.

Parameters	Value
Optimizer	Expectation Maximization
Map dimension	50×50 grid
Type of RBF	Radially symmetric
Number of RBF	400
Width σ of the RBF	0.8
Alarm threshold	100%
Log likelihood	9.85×10^5
Disrupted units	50.48%
Non-disrupted units	28.16%
Mixed units	18.52%
Empty clusters	5.48%
Mean assertion time (ms)	20

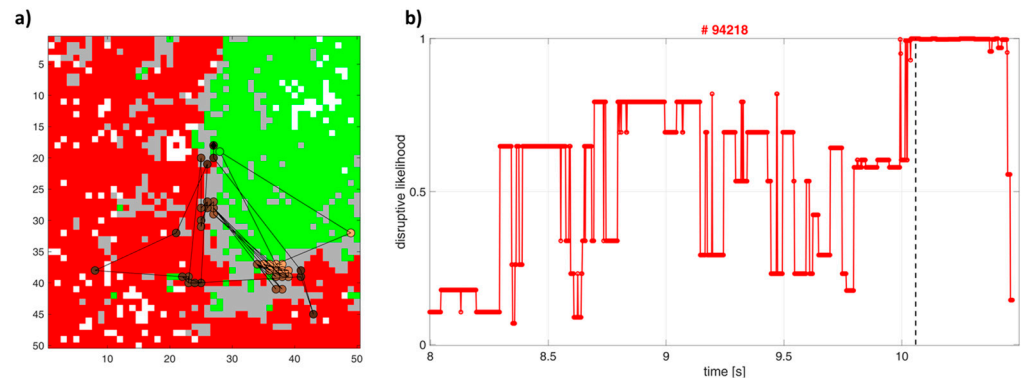


Figure 5. (a) GTM map of JET operational space with trajectory of the disrupted discharge #94218; (b) disruptive likelihood of the disrupted discharge #94218. The dashed black line identifies the alarm time.

The disruptive likelihood usually has a discontinuous trend with numerous peaks that could trigger incorrect alarms if an adequate threshold and assertion time were not optimized. Moreover, the normalized locked mode signal, not used to train the GTM model, is used in the multiple condition alarm scheme shown in Figure 6, as proposed in [8]. The assertion time $t_{assertion}$ is defined here as the time that the predictor waits before activating the alarm from the moment the disruptive likelihood exceeds the prefixed threshold. Note that the assertion time $t_{assertion}$ dynamically varies during the discharge. As the sampling time is assumed equal to 2 ms, a mean assertion time of 20 ms is obtained in the entire dataset. For the disrupted discharge #94218 in Figure 5, the GTM correctly predicts the disruption with a resulting warning time $\Delta t_{warning} = 410$ ms.

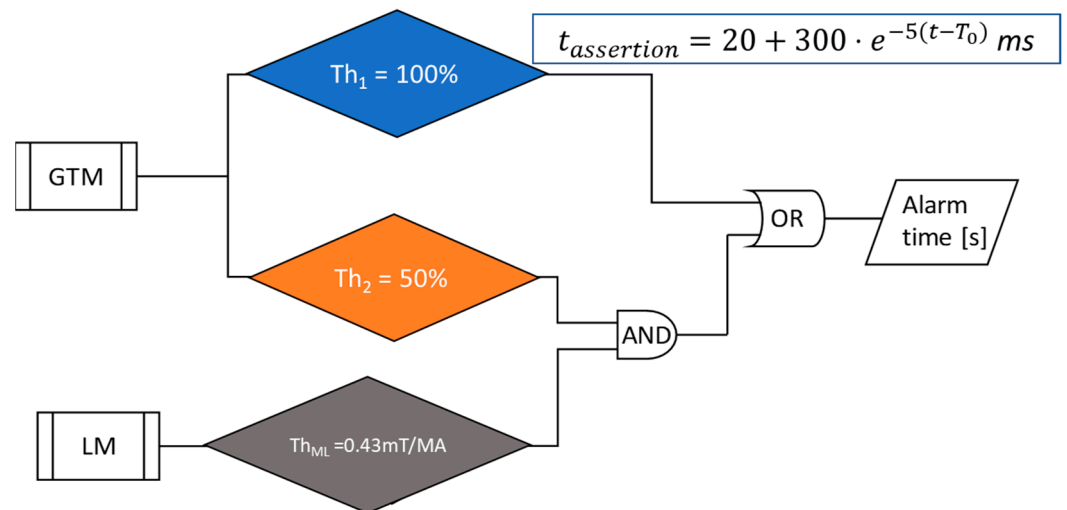


Figure 6. Multiple conditions alarm scheme of the GTM disruption predictor (T_0 is the starting time of the flattop).

Table 6 reports the confusion matrix (green: successful results; red: mistakes) and the values of the same prediction performance indices reported in Table 4. The recall is very high, which means a very high percentage of successful disruption predictions (97.22% in the test), but the specificity degrades compared to the MLP-NN due to the greater number of false alarms.

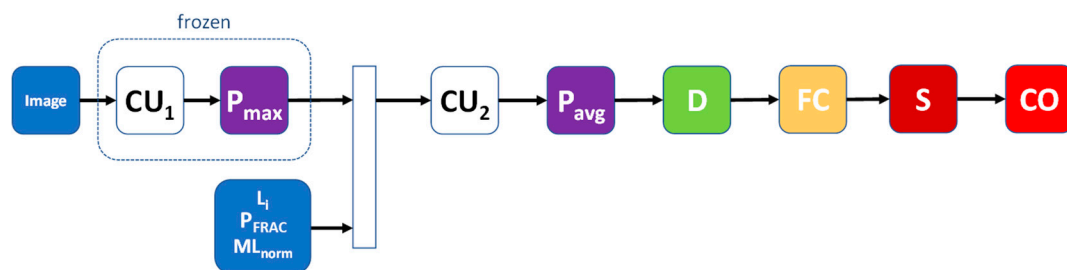
Table 6. Confusion matrix and performance indices of the GTM prediction model evaluated on the test set.

	P + N = 257	Predicted	
		P = 133	N = 124
Actual	P = 108	TP = 105	FN = 3
	N = 149	FP = 28	TN = 121
	PRECISION = 0.789	RECALL = 0.972	SPECIFICITY = 0.812
	ACCURACY = 0.879	F ₁ = 0.871	F ₂ = 0.929
Train	SP% = 100%	MA% = 0%	FA% = 0%
Test	SP% = 97.22%	MA% = 1.85%	FA% = 18.79%

Despite the lower performance with respect to the MLP-NN, the GTM model has had a considerable appreciation for its remarkable capabilities of visualizing the plasma operational space and the trajectories of the discharges on the map. This allows one to perform disruption prevention actions by monitoring the proximity of the discharge to the safe operational boundary.

3.2. CNN Disruption Prediction Model

The architecture of the CNN disruption predictor is reported in Figure 7. Due to the ability of the CNN to process images, the plasma profiles, which are 1-D signals, have been treated as a single image, as previously described. The other 0-D signals are fed in the CNN downstream of the first filter block. Note that the first filter block has been previously trained only with the 1-D diagnostic data, and its weights have been frozen. Then, in a second training phase, the CU₂ and FC blocks were trained using all diagnostic signals.

**Figure 7.** Architecture of the CNN disruption predictor.

The free parameters of the training process are reported in Table 7.

Table 7. CNN training parameters.

Parameters	Value
Optimizer	Stochastic gradient descent with momentum
Initial learning rate	1×10^{-4}
Learning rate drop factor	0.1
Learning rate drop period (epochs)	20
Momentum	0.9
MiniBatch size	16
Validation frequency (iterations)	50
Validation stop (consecutive evaluations)	100
Weight decay (L2 regularization)	1×10^{-4}
Assertion time (ms)	0

Figure 8a reports the image of the plasma profiles of the disrupted discharge #94218. By feeding the CNN with a sliding window of 200 ms on the test discharge, the corresponding disruptive likelihood outcomes are as reported in Figure 8b. The vertical dashed line identifies the alarm time t_{alarm} . The CNN is able to correctly predict the disruption with a warning time $\Delta t_{warning} = 372$ ms.

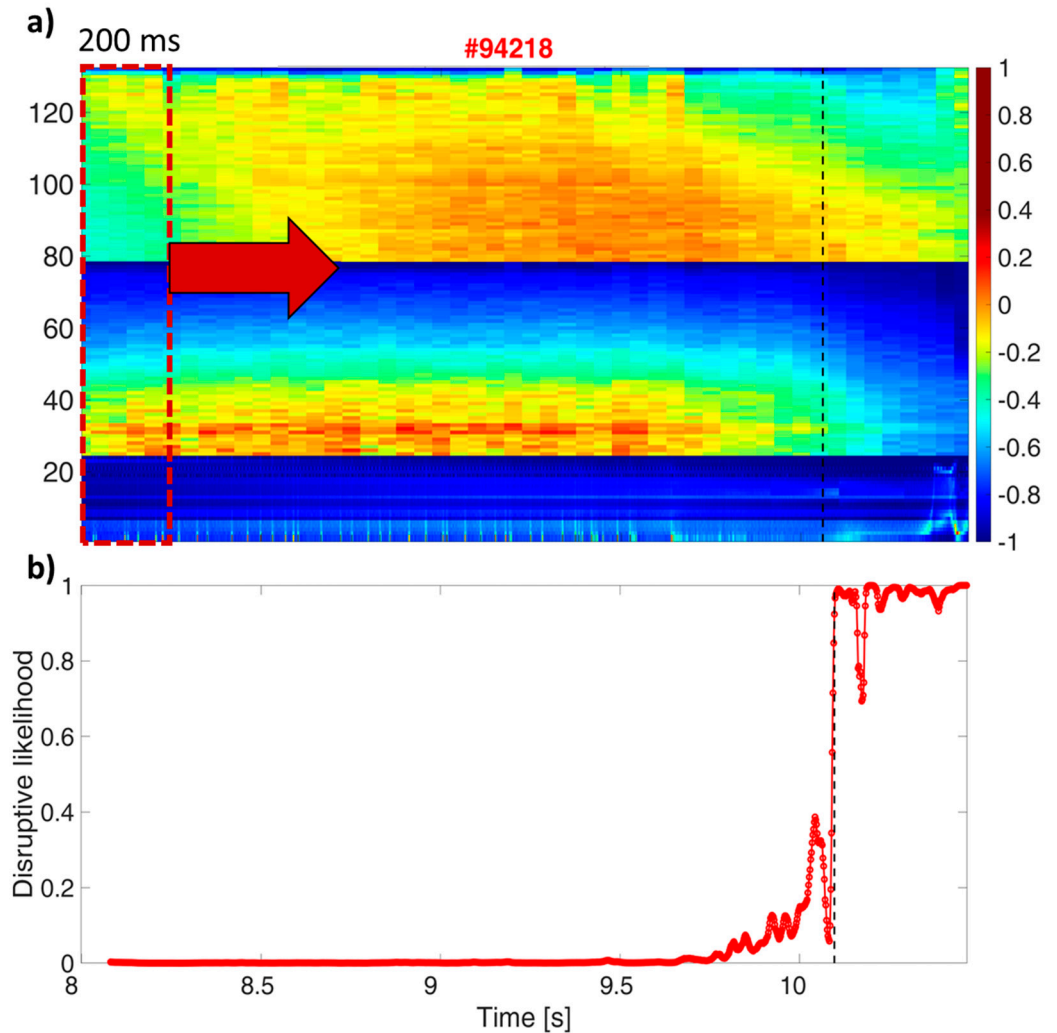


Figure 8. (a) CNN input image of the disrupted discharge #94218; (b) disruptive likelihood of the disrupted discharge #94218. The dashed black line identifies the alarm time.

Table 8 reports the confusion matrix (green: successful results; red: mistakes) and the values of the prediction performance indices of the CNN model. As in the case of the MLP-NN model, all the indices have high values obtaining a tradeoff between successful predictions and false alarms. Note that the plasma profiles have been directly used to feed the CNN model without the feature engineering process implemented for the MLP-NN and GTM.

Table 8. Confusion matrix and performance indices of the MLP-NN prediction model evaluated on the test set.

		Predicted	
		P + N = 257	N = 147
Actual	P = 108	TP = 102	FN = 6
	N = 149	FP = 8	TN = 141
	PRECISION = 0.927	RECALL = 0.944	SPECIFICITY = 0.946
	ACCURACY = 0.946	F ₁ = 0.936	F ₂ = 0.941
Train	SP% = 100%	MA% = 0%	FA% = 1.852%
Test	SP% = 94.44%	MA% = 2.778%	FA% = 5.369%

4. Discussion and Conclusions

An indicator of the performance of more immediate readings in the prediction of disruptions is the accumulated fraction of detected disruption as a function of the warning time $\Delta t_{warning}$. It provides the value, per unit, of successful alarms activated by at least the corresponding $\Delta t_{warning}$ in advance, giving also, in a unique graph, a general overview of the premature detections and the alarm anticipation times. Moreover, it allows the reading of the successful prediction fraction (SP), which corresponds to the intersection between the accumulative curve and the minimum anticipation time ($t_D - t_{valve} = 10$ ms at JET, red dashed vertical line), and the TD fraction as 1-SP. This is also a powerful means for comparing different models. Figure 9 reports such a comparison for the three proposed ML DP models.

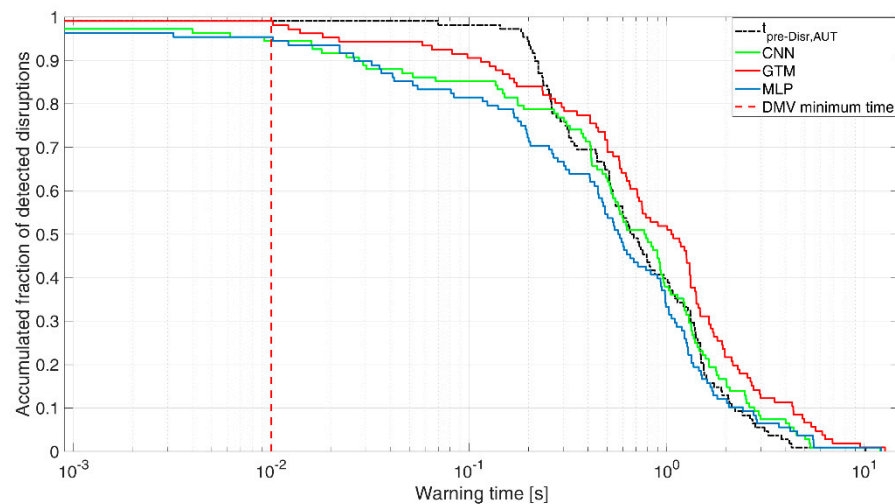


Figure 9. Accumulated fraction of detected disruptions by the MLP-NN (blue line), the GTM (red line), and the CNN (green line) models versus the warning time in the test set. The vertical red dashed line allows us to identify tardy detections. The dashed black curve refers to the $t_{pre-disr-AUT}$.

It is possible to see that the GTM (red line) has the earliest warning times, and its cumulative distribution of alarms is often to the right of $t_{pre-disr-AUT}$, which is our target. These early alarms can be associated to the high number of false alarms of the GTM, which have a less smooth disruptive likelihood and need an assertion time to trigger the alarm. Then, the CNN and MLP-NN have similar cumulative distributions, with the MLP-NN which triggers one alarm more just before the red dashed vertical line, and the CNN which triggers some tardy alarms after it. The CNN follows the $t_{pre-disr-AUT}$ distribution until 300 before the disruption, while the MLP-NN tends to stay to the left of the $t_{pre-disr-AUT}$

curve. The discharges where the three models missed the alarms are different from each other, but all except one pulse are characterized by a late mode-locking which causes the disruption. In the case of the GTM and of the FC-NN, the mode-lock absolute value is too low, and the predictors do not trigger an alarm, while in the CNN case, the mode-lock signal rise is not steep enough to trigger the alarm.

Note that the statistical algorithm developed to automatically detect $t_{pre-disr-AUT}$ [9] is non-causal and it cannot be used as a disruption predictor.

Despite the better results in terms of performances, the MLP-NN and the CNN are mostly employed as black box algorithms and do not allow the extraction of significant information on the disruption type and possible recovery strategies, while the GTM allows the tracking of the position of the discharge and the instability mechanism to be associated with the position of the point in the map. Among the MLP-NN and the CNN, the latter provides an overall higher number of alarms and a generally higher warning time, keeping a low number of false alarms.

Figure 10 reports, for the three predictors, the ROC curve. It is possible to see how the CNN and the MLP have the best compromise between the detection of disruptions and the number of false positives (false alarms), which is also visible in the AUC reported in Table 9. Moreover, looking at the points of the ROC, it is possible to verify that the CNN performance on the test is more robust than that on the MLP-NN. In fact, both models have an optimal threshold above 0.9, but the CNN has an overall accuracy of 89% even with the lower thresholds, up to 0.7. It is possible to also confirm this remark by comparing the three disruptive likelihoods in Figures 3b, 4b and 6b. The CNN has a lower disruptive likelihood in the stable phase of the disruption, and then rises abruptly in the last part of the discharge, where it is possible to also see a clear variation in the images of the profiles.

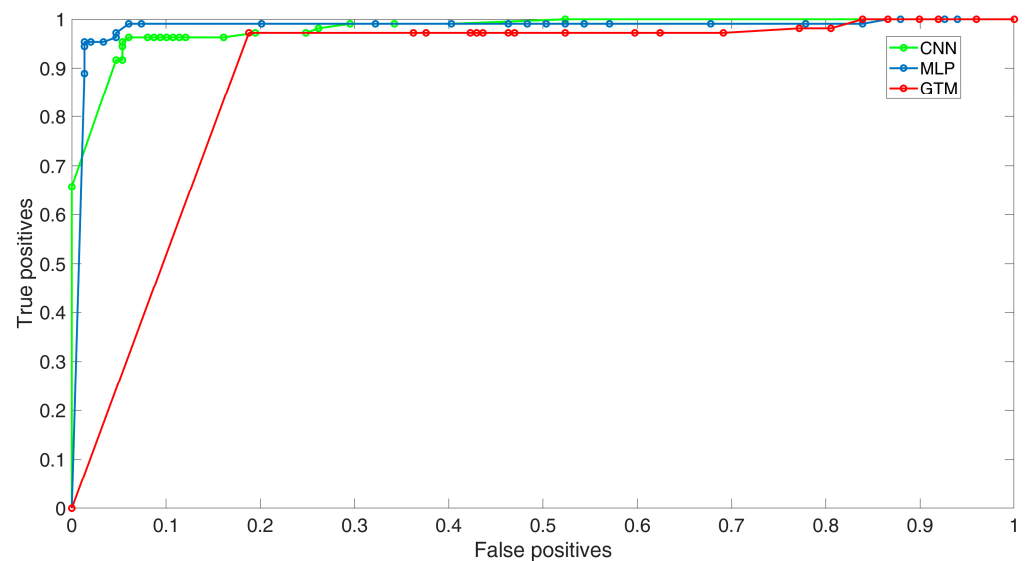


Figure 10. Receiver operating curve (ROC) of the MLP-NN (blue line), the GTM (red line), and the CNN (green line) models.

Table 9. AUC and assertion time for MLP-NN, GTM, and CNN models.

DP Model	AUC	Assertion Time (ms)
MLP-NN	0.98	0
GTM	0.89	20
CNN	0.98	0

In the same Table 9, the assertion time of the three models are reported.

Note that the more recent campaigns at the JET contain a higher number of mitigated discharges. The approach for the inclusion of mitigated discharges is the adoption of the DMS activation time as the t_D . This choice will unavoidably introduce an uncertainty in the classification of the discharge. However, since these shots are present in the test set only, we can notice that the predictors actually recognize the presence of disruption precursors before the valve activation, with a limited number of false alarms.

Concluding, in recent years, a plethora of different machine learning models, plasma features, and performance metrics have been proposed for the development of disruption predictors. This work aims to provide a systematic comparison of some of the most adopted models and to select common metrics for the results evaluation. Using the same training and test set, an MLP-NN, a GTM, and a CNN have been trained as disruption predictors starting from the same set of plasma parameters: the electron temperature, density and radiation profiles, the locked mode signal, the radiated fraction of the total input power, and the internal inductance. The GTM and the MLP-NN have been trained using a set of processed signals developed from the plasma profiles, the peaking factors, while the CNN is able to directly process the spatiotemporal images of the diagnostics. Every evaluated method demonstrated the capability of producing early warning times and, in the case of the MLP-NN and of the CNN, with a reduced number of false alarms. Despite the GTM performances being a bit below the other two, its advantage is the interpretability of the model output and the possibility to quantify the distance of the tracked discharge from the non-disrupted area of the map. On the other hand, the CNN has the advantage of being able to process the input images without the need of extracting physics-based features due its capability to process image data. In the future, 1-D profiles or images coming from other diagnostics can be exploited. As an example, in [45], the radiation profiles coming from the vertical lines of sight of the bolometer have been used with very encouraging results. The lower interpretability of neural network models could be addressed by exploiting analysis algorithms such as class activation mapping [46] and by developing predictors which identify specific events.

Nevertheless, the use of appropriate diagnostic signals, of a physics-based feature extraction and of automatically detected $t_{pre-disr-AUT}$, specific for each disruption, allowed us to train the models on a reduced number of discharges, to enable the detection of destabilization phenomena with a larger warning, and to maintain the performance on more recent discharges (up to the 2020 campaign), with very limited aging of the predictors.

Several metrics were adopted in evaluating the predictors' performance, from the confusion matrix to the typical metrics adopted in the machine learning community, such as recall, precision, etc. However, among all the proposed metrics, the accumulated fraction of detected disruptions against the warning time, together with the corresponding false alarms rate provided a clear and synthetic overview of the performances of the models, suggesting the use of these metrics in the evaluation of the predictors.

Concerning the portability of data-driven models to different devices, even if in the last 30 years many attempts have been made to implement data-driven disruption predictors for single machines, few papers present cross-predictor approaches to this problem [24,26,34,37,47–49]. The main result of these investigations is that, for machines dominated by the same disruptive chains of events, there is a comparable ensemble of physics mechanisms leading to the disruptions that can be described similarly in a unified framework of physics-based indicators [37]. Thus, the knowledge that data-driven algorithms learn on existing devices can be re-used to explain the disruptive behaviour on another device. This is an encouraging result in view of more extended studies to validate the transferability of the data-driven predictors. Alternatively, the use of at least some data or simulations from the new machine will be needed.

Author Contributions: Conceptualization, E.A., B.C., A.F., F.P. and G.S.; methodology, E.A., B.C., F.P. and G.S.; software, E.A., B.C. and F.P.; validation, A.F. and G.S.; formal analysis, E.A., B.C., A.F., F.P. and G.S.; investigation, E.A.; resources, E.A. and G.S.; data curation, E.A., C.S. (Carlo Sozzi), P.C. and C.S. (Chris Stuart); writing—original draft preparation, A.F.; writing—review and editing, E.A., B.C., A.F., F.P. and G.S.; visualization, F.P.; supervision, B.C.; project administration, A.F.; funding acquisition, A.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200—EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Boozer, A.H. Theory of Tokamak Disruptions. *Phys. Plasmas* **2012**, *19*, 058101. [[CrossRef](#)]
2. Hoelzl, M.; Huijsmans, G.T.A.; Pamela, S.J.P.; Bécoulet, M.; Nardon, E.; Artola, F.J.; Nkonga, B.; Atanasiu, C.V.; Bandaru, V.; Bhole, A.; et al. The JOREK Non-Linear Extended MHD Code and Applications to Large-Scale Instabilities and Their Control in Magnetically Confined Fusion Plasmas. *Nucl. Fusion* **2021**, *61*, 065001. [[CrossRef](#)]
3. Cannas, B.; Fanni, A.; Marongiu, E.; Sonato, P. Disruption Forecasting at JET Using Neural Networks. *Nucl. Fusion* **2003**, *44*, 68–76. [[CrossRef](#)]
4. Cannas, B.; Fanni, A.; Sonato, P.; Zedda, M.K.; Jet-Efda Contributors. A Prediction Tool for Real-Time Application in the Disruption Protection System at JET. *Nucl. Fusion* **2007**, *47*, 1559–1569. [[CrossRef](#)]
5. Cannas, B.; Delogu, R.S.; Fanni, A.; Sonato, P.; Zedda, M.K. Support Vector Machines for Disruption Prediction and Novelty Detection at JET. *Fusion Eng. Des.* **2007**, *82*, 1124–1130. [[CrossRef](#)]
6. Dormido-Canto, S.; Vega, J.; Ramírez, J.M.; Murari, A.; Moreno, R.; López, J.M.; Pereira, A.; JET-EFDA Contributors. Development of an Efficient Real-Time Disruption Predictor from Scratch on JET and Implications for ITER. *Nucl. Fusion* **2013**, *53*, 113001. [[CrossRef](#)]
7. Murari, A.; Lungaroni, M.; Peluso, E.; Gaudio, P.; Vega, J.; Dormido-Canto, S.; Baruzzo, M.; Gelfusa, M.; JET Contributors. Adaptive Predictors Based on Probabilistic SVM for Real Time Disruption Mitigation on JET. *Nucl. Fusion* **2018**, *58*, 056002. [[CrossRef](#)]
8. Pau, A.; Fanni, A.; Carcangiu, S.; Cannas, B.; Sias, G.; Murari, A.; Rimini, F. A Machine Learning Approach Based on Generative Topographic Mapping for Disruption Prevention and Avoidance at JET. *Nucl. Fusion* **2019**, *59*, 106017. [[CrossRef](#)]
9. Aymerich, E.; Fanni, A.; Sias, G.; Carcangiu, S.; Cannas, B.; Murari, A.; Pau, A.; JET Contributors. A Statistical Approach for the Automatic Identification of the Start of the Chain of Events Leading to the Disruptions at JET. *Nucl. Fusion* **2021**, *61*, 036013. [[CrossRef](#)]
10. Cannas, B.; Fanni, A.; Pautasso, G.; Sias, G.; Sonato, P. An Adaptive Real-Time Disruption Predictor for ASDEX Upgrade. *Nucl. Fusion* **2010**, *50*, 075004. [[CrossRef](#)]
11. Cannas, B.; Fanni, A.; Pautasso, G.; Sias, G. Disruption Prediction with Adaptive Neural Networks for ASDEX Upgrade. *Fusion Eng. Des.* **2011**, *86*, 1039–1044. [[CrossRef](#)]
12. Aledda, R.; Cannas, B.; Fanni, A.; Pau, A.; Sias, G. Improvements in Disruption Prediction at ASDEX Upgrade. *Fusion Eng. Des.* **2015**, *96–97*, 698–702. [[CrossRef](#)]
13. Rea, C.; Granetz, R.S.; Montes, K.; Tinguely, R.A.; Eidietis, N.; Hanson, J.M.; Sammuli, B. Disruption Prediction Investigations Using Machine Learning Tools on DIII-D and Alcator C-Mod. *Plasma Phys. Control. Fusion* **2018**, *60*, 084004. [[CrossRef](#)]
14. Montes, K.J.; Rea, C.; Granetz, R.S.; Tinguely, R.A.; Eidietis, N.; Meneghini, O.M.; Chen, D.L.; Shen, B.; Xiao, B.J.; Erickson, K.; et al. Machine Learning for Disruption Warnings on Alcator C-Mod, DIII-D, and EAST. *Nucl. Fusion* **2019**, *59*, 096015. [[CrossRef](#)]
15. Zheng, W.; Hu, F.R.; Zhang, M.; Chen, Z.Y.; Zhao, X.Q.; Wang, X.L.; Shi, P.; Zhang, X.L.; Zhang, X.Q.; Zhou, Y.N.; et al. Hybrid Neural Network for Density Limit Disruption Prediction and Avoidance on J-TEXT Tokamak. *Nucl. Fusion* **2018**, *58*, 056016. [[CrossRef](#)]
16. Yokoyama, T.; Miyoshi, Y.; Hiwatari, R.; Isayama, A.; Matsunaga, G.; Oyama, N.; Igarashi, Y.; Okada, M.; Ogawa, Y. Prediction of High-Beta Disruptions in JT-60U Based on Sparse Modeling Using Exhaustive Search. *Fusion Eng. Des.* **2019**, *140*, 67–80. [[CrossRef](#)]
17. Agarwal, A.; Mishra, A.; Sharma, P.; Jain, S.; Daniel, R.; Ranjan, S.; Manchanda, R.; Ghosh, J.; Tanna, R.; Team, A. Deep Sequence to Sequence Learning-Based Prediction of Major Disruptions in ADITYA Tokamak. *Plasma Phys. Control. Fusion* **2021**, *63*, 115004. [[CrossRef](#)]

18. Vega, J.; Murari, A.; Dormido-Canto, S.; Rattá, G.A.; Gelfusa, M. Disruption Prediction with Artificial Intelligence Techniques in Tokamak Plasmas. *Nat. Phys.* **2022**, *18*, 741–750. [[CrossRef](#)]
19. Aledda, R.; Cannas, B.; Fanni, A.; Sias, G.; Pautasso, G. Multivariate Statistical Models for Disruption Prediction at ASDEX Upgrade. *Fusion Eng. Des.* **2013**, *88*, 1297–1301. [[CrossRef](#)]
20. Zhang, Y.; Pautasso, G.; Kardaun, O.; Tardini, G.; Zhang, X.D.; ASDEX Upgrade Team. Prediction of Disruptions on ASDEX Upgrade Using Discriminant Analysis. *Nucl. Fusion* **2011**, *51*, 063039. [[CrossRef](#)]
21. Murari, A.; Vagliasindi, G.; Arena, P.; Fortuna, L.; Barana, O.; Johnson, M.; JET Contributors. Prototype of an Adaptive Disruption Predictor for JET Based on Fuzzy Logic and Regression Trees. *Nucl. Fusion* **2008**, *48*, 035010. [[CrossRef](#)]
22. Wang, S.Y.; Chen, Z.Y.; Huang, D.W.; Tong, R.H.; Yan, W.; Wei, Y.N.; Ma, T.K.; Zhang, M.; Zhuang, G. Prediction of Density Limit Disruptions on the J-TEXT Tokamak. *Plasma Phys. Control. Fusion* **2016**, *58*, 055014. [[CrossRef](#)]
23. Rattá, G.A.; Vega, J.; Murari, A.; Vagliasindi, G.; Johnson, M.F.; de Vries, P.C.; JET Contributors. An Advanced Disruption Predictor for JET Tested in a Simulated Real-Time Environment. *Nucl. Fusion* **2010**, *50*, 025005. [[CrossRef](#)]
24. Kates-Harbeck, J.; Svyatkovskiy, A.; Tang, W. Predicting Disruptive Instabilities in Controlled Fusion Plasmas through Deep Learning. *Nature* **2019**, *568*, 526–531. [[CrossRef](#)] [[PubMed](#)]
25. Ferreira, D.R.; Carvalho, P.J.; Fernandes, H. Deep Learning for Plasma Tomography and Disruption Prediction from Bolometer Data. *IEEE Trans. Plasma Sci.* **2020**, *48*, 36–45. [[CrossRef](#)]
26. Rattá, G.A.; Vega, J.; Murari, A. Viability Assessment of a Cross-Tokamak AUG-JET Disruption Predictor. *Fusion Sci. Technol.* **2018**, *74*, 13–22. [[CrossRef](#)]
27. Rattá, G.A.; Vega, J.; Murari, A. Improved Feature Selection Based on Genetic Algorithms for Real Time Disruption Prediction on JET. *Fusion Eng. Des.* **2012**, *87*, 1670–1678. [[CrossRef](#)]
28. Rattá, G.A.; Vega, J.; Murari, A.; Gadariya, D.; JET Contributors. PHAD: A Phase-Oriented Disruption Prediction Strategy for Avoidance, Prevention, and Mitigation in JET. *Nucl. Fusion* **2021**, *61*, 116055. [[CrossRef](#)]
29. Aledda, R.; Cannas, B.; Fanni, A.; Sias, G.; Pautasso, G. Mapping of the ASDEX Upgrade Operational Space for Disruption Prediction. *IEEE Trans. Plasma Sci.* **2012**, *40*, 570–576. [[CrossRef](#)]
30. Cannas, B.; Fanni, A.; Murari, A.; Pau, A.; Sias, G.; JET Contributors. Automatic Disruption Classification Based on Manifold Learning for Real-Time Applications on JET. *Nucl. Fusion* **2013**, *53*, 093023. [[CrossRef](#)]
31. Hu, W.H.; Rea, C.; Yuan, Q.P.; Erickson, K.G.; Chen, D.L.; Shen, B.; Huang, Y.; Xiao, J.Y.; Chen, J.J.; Duan, Y.M.; et al. Real-Time Prediction of High-Density EAST Disruptions Using Random Forest. *Nucl. Fusion* **2021**, *61*, 066034. [[CrossRef](#)]
32. Churchill, R.M.; Tobias, B.; Zhu, Y. Deep Convolutional Neural Networks for Multi-Scale Time-Series Classification and Application to Tokamak Disruption Prediction Using Raw, High Temporal Resolution Diagnostic Data. *Phys. Plasmas* **2020**, *27*, 062510. [[CrossRef](#)]
33. Aymerich, E.; Sias, G.; Pisano, F.; Cannas, B.; Carcangiu, S.; Sozzi, C.; Stuart, C.; Carvalho, P.; Fanni, A. Disruption Prediction at JET through Deep Convolutional Neural Networks Using Spatiotemporal Information from Plasma Profiles. *Nucl. Fusion* **2022**, *62*, 066005. [[CrossRef](#)]
34. Zhu, J.X.; Rea, C.; Montes, K.; Granetz, R.S.; Sweeney, R.; Tinguely, R.A. Hybrid Deep-Learning Architecture for General Disruption Prediction across Multiple Tokamaks. *Nucl. Fusion* **2020**, *61*, 026007. [[CrossRef](#)]
35. Guo, B.H.; Shen, B.; Chen, D.L.; Rea, C.; Granetz, R.S.; Huang, Y.; Zeng, L.; Zhang, H.; Qian, J.P.; Sun, Y.W.; et al. Disruption Prediction Using a Full Convolutional Neural Network on EAST. *Plasma Phys. Control. Fusion* **2020**, *63*, 025008. [[CrossRef](#)]
36. Pau, A.; Fanni, A.; Cannas, B.; Carcangiu, S.; Pisano, G.; Sias, G.; Sparapani, P.; Baruzzo, M.; Murari, A.; Rimini, F.; et al. A First Analysis of JET Plasma Profile-Based Indicators for Disruption Prediction and Avoidance. *IEEE Trans. Plasma Sci.* **2018**, *46*, 2691–2698. [[CrossRef](#)]
37. Rea, C.; Montes, K.J.; Pau, A.; Granetz, R.S.; Sauter, O. Progress Toward Interpretable Machine Learning-Based Disruption Predictors Across Tokamaks. *Fusion Sci. Technol.* **2020**, *76*, 912–924. [[CrossRef](#)]
38. Garcia, J.; Casson, F.J.; Challis, C.; Frigione, D.; Van Eester, D.; Garzotti, L.; Hobirk, J.; Kappatou, A.; Lerche, E.A.; Mailloux, J.; et al. Integrated Scenario Development at JET for DT Operation and ITER Risk Mitigation. In Proceedings of the 28th IAEA Fusion Energy Conference (FEC 2020), Virtual, 10–15 May 2021.
39. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning Representations by Back-Propagating Errors. *Nature* **1986**, *323*, 533–536. [[CrossRef](#)]
40. Bishop, C.M.; Svensén, M.; Williams, C.K.I. GTM: The Generative Topographic Mapping. *Neural Comput.* **1998**, *10*, 215–234. [[CrossRef](#)]
41. O’Shea, K.; Nash, R. An Introduction to Convolutional Neural Networks. *arXiv* **2015**, arXiv:1511.08458.
42. Leyland, M.J.; Beurskens, M.N.A.; Flanagan, J.C.; Frassinetti, L.; Gibson, K.J.; Kempenaars, M.; Maslov, M.; Scannell, R.; JET Contributors. Edge Profile Analysis of Joint European Torus (JET) Thomson Scattering Data: Quantifying the Systematic Error Due to Edge Localised Mode Synchronisation. *Rev. Sci. Instrum.* **2016**, *87*, 013507. [[CrossRef](#)]
43. Møller, M.F. A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. *Neural Netw.* **1993**, *6*, 525–533. [[CrossRef](#)]
44. Carcangiu, S.; Fanni, A.; Montisci, A. Multiobjective Tabu Search Algorithms for Optimal Design of Electromagnetic Devices. *IEEE Trans. Magn.* **2008**, *44*, 970–973. [[CrossRef](#)]
45. Enrico, A.; Giuliana, S.; Fabio, P.; Barbara, C.; Alessandra, F. CNN Disruption Predictor at JET: Early versus Late Data Fusion Approach. In Proceedings of the 32nd Symposium of Fusion Technology (SOFT32), Dubrovnik, Croatia, 18–23 September 2022.

46. Ferreira, D.R.; Martins, T.A.; Rodrigues, P.; JET Contributors. Explainable Deep Learning for the Analysis of MHD Spectrograms in Nuclear Fusion. *Mach. Learn. Sci. Technol.* **2021**, *3*, 015015. [[CrossRef](#)]
47. Windsor, C.G.; Pautasso, G.; Tichmann, C.; Buttery, R.J.; Hender, T.C.; JET Contributors; The ASDEX Upgrade Team. A Cross-Tokamak Neural Network Disruption Predictor for the JET and ASDEX Upgrade Tokamaks. *Nucl. Fusion* **2005**, *45*, 337–350. [[CrossRef](#)]
48. Yang, Z.; Xia, F.; Song, X.; Gao, Z.; Li, Y.; Gong, X.; Dong, Y.; Zhang, Y.; Chen, C.; Luo, C.; et al. Real-Time Disruption Prediction in the Plasma Control System of HL-2A Based on Deep Learning. *Fusion Eng. Des.* **2022**, *182*, 113223. [[CrossRef](#)]
49. Sias, G.; Cannas, B.; Fanni, A.; Murari, A.; Pau, A. A Locked Mode Indicator for Disruption Prediction on JET and ASDEX Upgrade. *Fusion Eng. Des.* **2019**, *138*, 254–266. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.