

Research Article

Nuclear Fusion Pattern Recognition by Ensemble Learning

G. Farias ¹, **E. Fabregas** ², **I. Martínez** ¹, **J. Vega** ³, **S. Dormido-Canto** ²,
and **H. Vargas** ¹

¹*Pontificia Universidad Católica de Valparaíso, Avenida Brasil 2147, Valparaíso 2362804, Chile*

²*Departamento de Informática y Automática, Universidad Nacional de Educación a Distancia, Juan Del Rosal 16, Madrid 28040, Spain*

³*Laboratorio Nacional de Fusión, CIEMAT, Avenida Complutense 40, Madrid 28040, Spain*

Correspondence should be addressed to G. Farias; gonzalo.farias@pucv.cl

Received 26 May 2021; Accepted 19 June 2021; Published 30 June 2021

Academic Editor: Atila Bueno

Copyright © 2021 G. Farias et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nuclear fusion is the process by which two or more atomic nuclei join together to form a single heavier nucleus. This is usually accompanied by the release of large quantities of energy. This energy could be cheaper, cleaner, and safer than other technology currently in use. Experiments in nuclear fusion generate a large number of signals that are stored in huge databases. It is impossible to do a complete analysis of this data manually, and it is essential to automate this process. That is why machine learning models have been used to this end in previous years. In the literature, several popular algorithms can be found to carry out the automatic classification of signals. Among these, ensemble methods provide a good balance between success rate and internal information about models. Specifically, AdaBoost algorithm will allow obtaining an explicit set of rules that explains the class for each input data, adding interpretability to the models. In this paper, an innovative approach to perform an online classification, that is, to identify the discharge before it actually ends, using interpretable models is presented. In order to evaluate and reveal the benefits of rule-based models, an illustrative example has been implemented to perform an online classification of five different signals of the TJ-II stellarator fusion device located in Madrid, Spain.

1. Introduction

Energy is a crucial element for the subsistence of our modern civilization. Almost all human activities require energy to work. This requirement is increased year after year, especially due to the growing population, which is estimated by about 10 billion people in the middle of this century [1]. Nowadays, fossil fuels are the main source of energy because of their relatively low cost of production and high energetic capacity. However, they are not a long-term option. Alternatives like renewable energies are increasing their participation in modern life. However, the current technology of renewable sources is still not able to supply all energy needed [2, 3].

On the contrary nuclear sources can provide great quantities of energy. Although fusion energy is still developing, its potential is enormous, even compared with nuclear fission. Nuclear fusion is the process by which two or more atomic nuclei join together to form a single heavier

nucleus. This is usually accompanied by the release of large quantities of energy. Fusion is the process that powers active stars, the hydrogen bomb, and some experimental devices. Nuclear fusion could be cheaper, cleaner, and safer. Fusion power would provide much more energy than any other technology currently in use, and the fuel required for fusion, mainly deuterium, exists abundantly in the oceans. Fusion could, in theory, supply all the energy needs of the world for millions of years [4, 5].

Achieving full control of the energy generated by nuclear fusion devices involves an analysis over huge databases with thousands of signals that is impossible to do it manually. This amount of data implies performing the analysis (e.g., finding significance or regular patterns) in high dimensional spaces and it is essential to automate the process using machine learning [6–9]. To this end, we can find several algorithms in the literature in order to perform pattern recognition in an automatic way. In the context of the pattern classification problem, the most popular algorithms

are Support Vector Machines and Neural Networks; both have shown high performance in previous applications in fusion [10–12], but with an important inconvenient: these algorithms produce black-box models, where it is not possible to obtain explicitly a simple mathematical relationship that outputs the classification.

The aim of this article is to present a new approach that combines the pattern (waveform) classification online with internal information about the decision model (i.e., interpretability). In the literature, there are several examples of using black-box models to automate the classification problem. The main reason for this is the high success rates reported in different topics such as nuclear fusion [12–14]. However, a black-box model does not give us any hint about the reason for the classification; for example, we are not able to know what the most important input variable is involved in the decision.

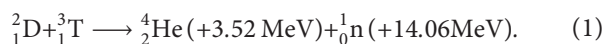
In this sense, ensemble methods provide a good balance between success rate and internal information about the model [15, 16]. Particularly, the AdaBoost algorithm allows obtaining an explicit set of simple rules that outputs the class of the signal from the input data [17]. Such property adds interpretability to the models, which could be useful to understand the reason for the classification process and, ultimately, for improving knowledge of the underlying physical phenomenon. This fact will allow a much more precise adjustment of the obtained models.

The main contributions of this article are as follows: (1) the waveform classification using ensemble methods generate rules-based models (if-then rules) that are not black boxes and could be useful to understand the entire process of the plasma discharges in nuclear fusion devices and (2) the classification system of waveform works online, which implies that we do not need to wait until the discharge finishes to obtain the class from the input data.

The rest of the paper is structured as follows: Section 2 introduces some basics aspects of the Nuclear Fusion Energy, the AdaBoost algorithm, and the signals used. Section 3 presents the offline and online approaches to classify the signals. A brief analysis of the models and features obtained is also presented. Finally, Section 4 summarizes the main conclusions.

2. Background

2.1. Nuclear Fusion Energy. In order to reproduce on the Earth the fusion power, some fusion reaction can be used. One of the most important is the deuterium-tritium cycle [18], which release 17.58 MeV as follows:



In a fusion device, the reaction is produced at very high temperatures, about 150 million degrees Celsius. To this temperature, the matter inside of fusion devices is found like plasma, which is a state of matter similar to gas with a portion of its particles ionized. Magnetic fields are used to confine plasma in the shape of a torus. The most common configurations for magnetic confinement of plasma are

stellarators and tokamaks. Figure 1 shows a simple and general scheme of the process of generating electrical energy from the nuclear fusion. The reactor uses deuterium (D) and tritium (T) to produce the reaction. The water is heated by the energy of the reaction and it feeds a turbine generator that produces the electricity.

The International Thermonuclear Experimental Reactor (ITER) is an international nuclear fusion research and engineering project, which is currently building the world's largest and most advanced experimental tokamak nuclear fusion reactor at the Cadarache (France) [18]. ITER is expected to demonstrate that more energy is obtained than is used to initiate the fusion process, something that has not been achieved by any experimental fusion reactor. After ITER, the first commercial demonstration fusion power plant, named DEMO [19], will be intended. Currently, there are many experimental fusion devices in operation. The Joint European Torus (JET) [20] is an experimental tokamak reactor located in Oxfordshire (UK). It is currently the largest facility of its kind in operation. TJ-II [21] is a medium size stellarator located at CIEMAT in Madrid (Spain). DIII-D is another tokamak machine developed by General Atomics in San Diego (USA) [22].

Experiments on fusion reactors are carried out by producing discharges or shot, in which plasma exists inside the torus. The duration of the shot is normally tens of seconds. ITER would keep the shot for about 30 minutes. During the discharges, many diagnostics around the reactor acquire data at high sampling frequencies. About 10 GBytes per discharge can be acquired in JET [20] (ITER could storage 1 TByte per shot). Bolometry, density, temperature, and soft X-rays are just some examples of the thousands of data sets acquired during a discharge. Huge databases, with an enormous amount of data, are a common situation in experimental fusion reactors.

However, nowadays only 10% of the generated data is processed, while the rest is not processed at all. Therefore, in order to achieve fusion energy as a clean, inexhaustible, safe, and cheap energy source, the current databases of experimental devices (tokamaks and stellarators) should be analyzed completely. Performing complete analysis will involve an optimal operation planning of ITER and, in turn, will be basic for a successful design of DEMO. For that reason, this project proposes the use of advanced pattern recognition and machine learning techniques in order to analyze in a faster and more efficient way massive fusion databases.

In this paper, the AdaBoost algorithm is used to build a rule-based model to classify five different waveform classes of the TJ-II stellarator. The plasmas in TJ-II are produced and heated with ECRH (2 gyrotrons, 300 kW each, 53.2 GHz, 2nd harmonic, X-mode polarization) and NBI (300 kW) [21]. Figure 2 shows a view of the TJ-II device.

2.2. AdaBoost Algorithm. The adaptive Boosting algorithm (AdaBoost) was proposed by Yoav Freund and Robert Shapire [17, 24]. AdaBoost is a general method to obtain a strong classifier (in our case a rule-based model) from a set of T weak classifiers (also called hypotheses, or rules in our

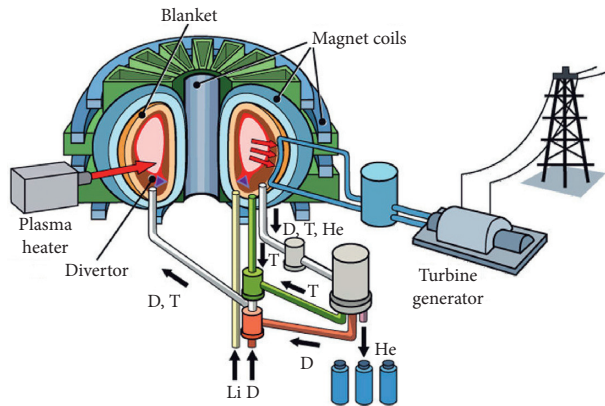


FIGURE 1: Electrical energy produced from fusion.

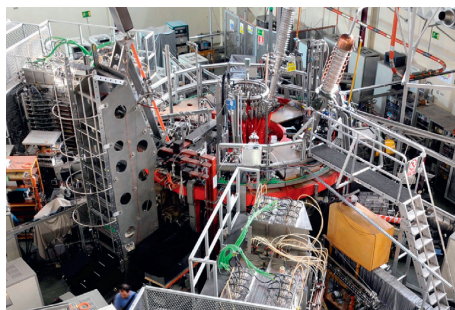


FIGURE 2: The TJ-II device [23].

case). This algorithm takes as input a set $S = \{F, y\}$, where F is a features vector of the signal that is going to be classified, y is the label of class for each signal (+1 or -1), and m is the total number of signals. Note that h_t is a rule, and T represents the number of rules that composes the rule-based model. Algorithm 1 is the pseudocode that shows the implementation of this algorithm.

The basic idea of boosting is to select the best weak (and simple) classifier after each iteration. The hypothesis selected is weighted according to its capacity to classify the training set correctly. Samples that were not correctly classified are also weighted in order to look for a suitable hypothesis for them in the next iteration. AdaBoost uses exponential error loss as an error criterion. The final model corresponds to a weighted sum of the selected weak hypotheses. The most important issue is that the resulting model is based on if-else rules, which means that the model is not based on a black-box type model. This represents an advantage compared to other classification algorithms [24].

In this way, AdaBoost can be used in a straight forward manner with signals. For example, a simple rule could be to predict a class if the average of the last 30 milliseconds is greater than a threshold. Thus, we can use if-then sentences such as $\text{if } (\text{avg}(\text{signal}) > \text{threshold}) \text{ then } +1, \text{ else } -1$ as a weak rule h_t (as in lines 9 and 10 of the pseudocode above). The output of the AdaBoost classifier will be finally the sign of the weighted sum of T rules (line 18 of the pseudocode) such as

in equation (2). Note that α_t corresponds to the importance or weight of each class:

$$\text{Class}(F) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t \cdot (F) \right). \quad (2)$$

The algorithm can be easily extended for a multiclass problem (more than two classes) using the approach the one versus the rest, which implies building a model to classify the waveforms of a particular class (+1) versus the waveforms that belong to a different class (-1). This process is repeated in order to build one classifier for each class. For example, in [24], there is a detailed explanation about fundamental concepts of AdaBoost. In [11, 25], there are good descriptions about implementing classifiers for two or more classes in nuclear fusion databases combined with other algorithms (autoencoder and wavelet).

Figure 3 shows an illustrative example of the AdaBoost algorithm from [26], which is a previous work of the authors. There are seven samples of two classes (red circle and blue cross) in the upper image. Let us assume that a new sample located at (3.5; 3.0) has to be classified in one of the two classes. We can use the seven samples to build (train) a supervised data-driven model to predict the class of the new sample by using AdaBoost. The feature vectors are the respective Cartesian coordinates x_1 and x_2 . After some iterations, the new sample is classified as a cross. The image

```

(1) Input:  $S = \{(F_i, y_i), \forall i = 1 \dots m\}$ 
(2) #S: Training set example
(3) # $D_1$ : Initial weight distribution
(4) # $m$ : Size of the training set
(5) # $y_i \in \{-1; +1\}$ 
(6)  $D_1(i) = 1/m, \forall i = 1 \dots m$ 
(7) #T rules that compose the rule-based model
(8) for  $t := 1$  to  $T$  do
(9)   #Get weak hypothesis  $h_t: F \rightarrow \{-1, +1\}$ 
(10)   $h_t = \text{GetWeakHypothesis}(F, D_t)$ 
(11)   $e_t = \sum_{i: h_t(F_i) \neq y_i} D_t(i)$ 
(12)   $\alpha_t = (1/2) \ln((1 - e_t)/e_t)$ 
(13)  #Update  $D_t$  distribution
(14)   $D_{t+1}(i) = D_t(i) \exp(-\alpha_t y_i h_t(F_i)) / Z_t, \forall i = 1, \dots, m$ 
(15)  # $Z_t$ : Normalization factor  $D_{t+1}$  is a distribution:
(16)  # $Z_t = \sum_{i=1}^m D_t(i) \exp(-\alpha_t y_i h_t(F_i))$ 
(17) end for
(18) return hypothesis combination:  $C(F) = \sum_{t=1}^T \alpha_t h_t(F)$ 
(19) end AdaBoost

```

ALGORITHM 1: AdaBoost algorithm.

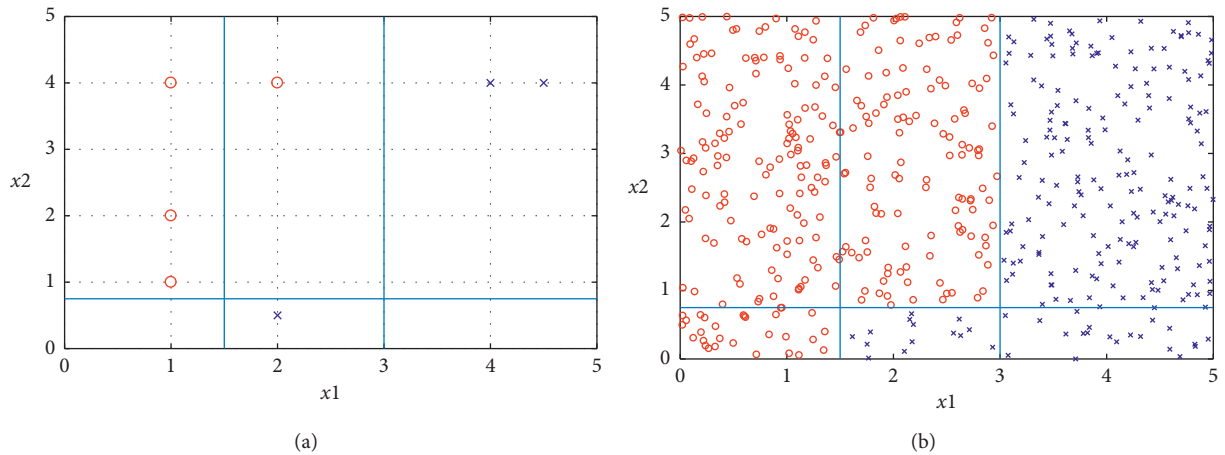


FIGURE 3: (a) Two classes (training samples). (b) Classification of 500 new samples.

below shows the result of testing the AdaBoost model for 500 new samples. Interested readers can find more technical details in the literature.

2.3. TJ-II Waveforms. The data generated by the experiments of the TJ-II device is stored in the relational database called TJ2RDB. This database allows searches to find shots with particular properties and to do scaling studies. More general information about this database can be found in [27, 28].

To illustrate the complexity of the data used for this work, we show the waveform of one of the signals involved in this research. Figure 4 shows the signal ECE7 for 200 shots. As it can be seen, for the same signal, the shape of the waveform is very different from a shot to another one. This

implies that the classification of these kinds of signals can be a difficult task if it is carried out manually.

In order to evaluate and reveal the benefits of this approach, we have implemented a proof-of-concept using an illustrative example of the online classification for five different waveforms. This explanatory classification problem has been selected because the proposed approach can be easily compared with other previous works, where black-box algorithms have been implemented.

Figure 5 shows the temporal evolution of the 5 waveforms used to test the proposed approach in this work. From top to bottom and from left to the right, the waveforms (classes) are ECE7, GR, GR2, HALFAC3, and IACCEL1.

Table 1 presents a brief description of the selected TJ-II signals. Note that the selection of other signals might provide

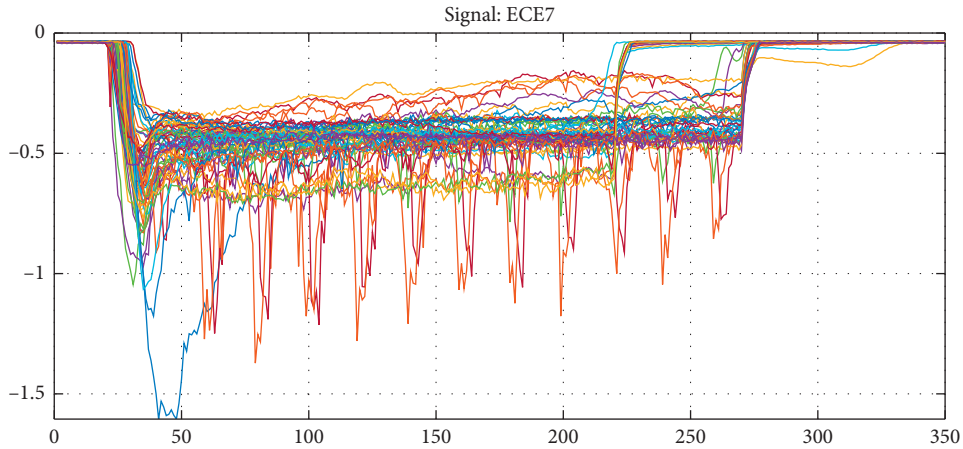


FIGURE 4: The plot depicts amplitude versus time (ms) of the ECE7 signal for 60 random shots.

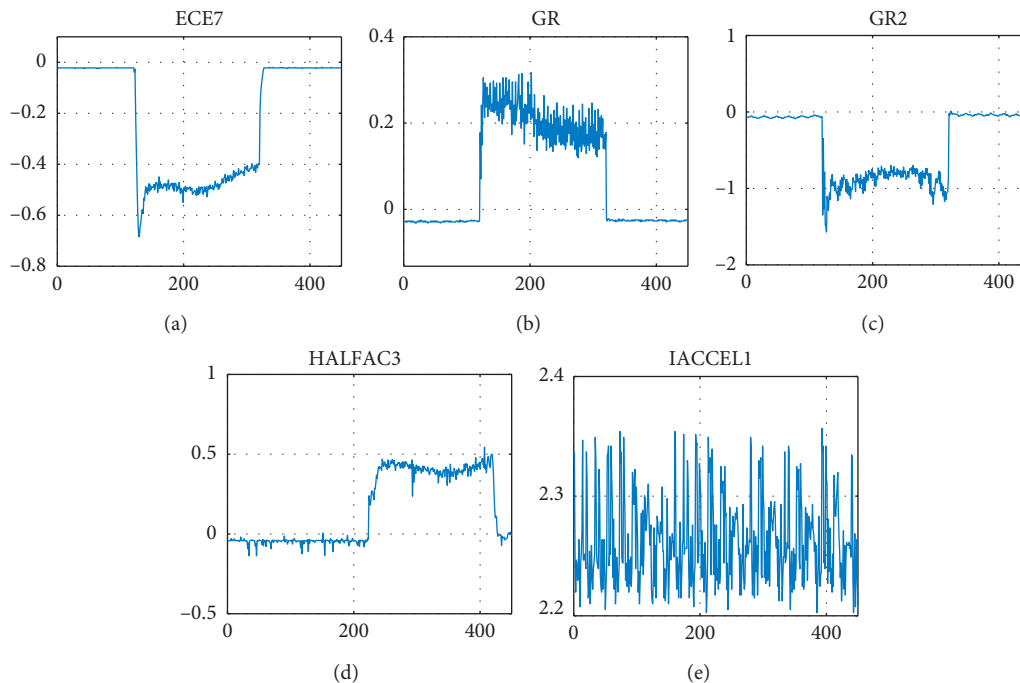


FIGURE 5: The plot depicts amplitude versus time (ms) of the five TJ-II waveforms classes: (a) ECE7, (b) GR, (c) GR2, (d) HALFAC3, and (e) IACCEL1 described in Table 1.

different results to those presented here, but the approach is enough general to obtain a classification with similar successful rates.

Finally, note that a supervised training scheme requires a previously labelled data set, and since in this context each signal is acquired by a separated sensor system, all the labels are known when data is stored. In a different context, the labelling process could imply the assistance of many specialists to obtain such data sets.

3. Waveforms Classification

The waveform classification is developed using AdaBoost with two approaches: (1) offline and (2) online. In the offline approach, the obtained model uses the entire signal to

perform the classification, which involves the classification is done after the discharge has finished. On the other hand, a sensitivity analysis was also performed to select a reduced set of features in order to classify the waveform before it actually finishes, which could be very interesting for real-time applications. For comparison purposes, the rule-based model has been tested with signals used in previous works.

3.1. Offline Classification. For the offline approach, the AdaBoost algorithm has been implemented to classify the TJ-II waveforms using all the samples of the discharge. In this case, 340 waveforms have been used in total (68 waveforms for each class). Each entire waveform is resampled to 1024 samples to form the feature vector (F) in

TABLE 1: TJ-II waveforms.

Signal	Description	Class	Samples
ECE7	Electron cyclotron em.	1	500
GR	First gyrotron	2	500
GR2	Second gyrotron	3	500
HALFAC3	H α line intensity	4	500
IACCELI	Neutral beam injector	5	500

order to feed the AdaBoost algorithm. Finally, AdaBoost will output a rule-based model (AdaBoost model) that allows classifying a new waveform. Figure 6 shows the block diagram of the implemented stages.

Table 2 shows three rules (h_t) and their associated weights (α_t) of the obtained rule-based model to classify GR signals (class 2). Note that the features F_{578} (magnitude of GR signal at sample 578), F_{842} (sample 842), and F_{1024} (sample 1024) are used to perform the classification.

In the case of class 3 (GR2 signal), the classification can be performed by using only the following rule: if ($F_1 < -1.151$) then +1 else -1, which implies that using only the first feature of a discharge (F_1), the approach is able to classify GR2 signals successfully.

In order to evaluate the model, we split the data into two subsets (cross-validation). The training stage was carried out with 60% of the data set (205 waveforms for each of the 5 classes) while 40% of the data set was used for the test stage (135 waveforms for each class).

Table 3 shows the results of the offline classification of the 5 types of signals. As it can be seen, the results are encouraging. All the success rates are above 93%.

The average success rate of the ensemble model is up to 98%, improving the results of previous works. In [25], a Wavelet Transform with Support Vector Machines (WT + SVM) [25], the results were up to 92%. More recently, in [11], a Stacked Autoencoder (a type of Neural Network) in combination with Support Vector Machines (NN + SVM), the results were up to 94%.

One interesting thing about the proposed approach is the ability to see the importance of each feature to perform the classification. Figure 7 shows the features selected (samples) by the algorithm to classify the ECE7 signals. The blue line represents shot and red circles represent the features used to classify this signal.

The size of the circles is proportional to the importance of the rule (α_t) that uses the feature. As it can be seen, the most important values are located before sample 200, which implies that some signals could be classified at the beginning of the discharge; that is, an online classification could be performed.

Based on the previous results, the idea of online classification came up. In this way, it is not necessary to wait until the end of the discharge in order to perform the classification. The next section presents the online classification algorithm.

3.2. Online Classification. This approach starts the classification at the very beginning of the discharge. First, the signal is preprocessed in sliding windows obtained by grouping 10

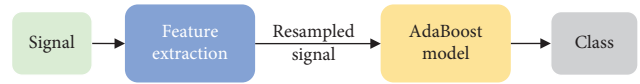


FIGURE 6: Block diagram of the offline application.

TABLE 2: Example of rules obtained to classify GR signals.

h_t	α_t
if ($F_{578} < 2.127$) then +1 else -1	0.3726
if ($F_{842} < 2.108$) then +1 else -1	0.3782
if ($F_{1024} < 0.131$) then +1 else -1	0.5675

TABLE 3: Results of the classification of the 5 signals.

Signal	Success rate (%)
ECE7	99.18
GR	98.35
GR2	100
HALFAC3	93.83
IACCELI	100

consecutive samples and taking only one representative sample for each window. In this way, the signal is reduced by a factor of 10. Then, the feature extraction stage is applied to obtain some specific characteristics of the signal that helps in the classification. Figure 8 shows the block diagram of the online approach.

Figure 9 shows an explanatory diagram of the algorithm. The red solid line represents the signal that is being analyzed. The blue dashed rectangle represents the sliding window, which contains the segment of the signal analyzed at the current iteration.

Then, the four features of this window are obtained: average value (F_1), minimum value (F_2), maximum value (F_3), and, finally, slope value (F_4), which is calculated by performing a least squares adjustment. After that, the AdaBoost model classifies the signal into one of the five classes. When the result of the AdaBoost is three consecutive positive values (+1), the signal is classified as this class. In this example, the signal is classified as HALFAC3 (Class 4) as is represented by the red dashed rectangle.

Similar to the offline case, we can easily obtain the attributes used by the algorithm to perform the classification. Table 4 shows the features used for each class. Note that classes 1, 2, and 5 use only three features.

Table 5 presents the confusion matrix that shows the results of the online classification of the five classes. Rows represent the class that is being classified and the columns represent the predictions of the classification for the actual

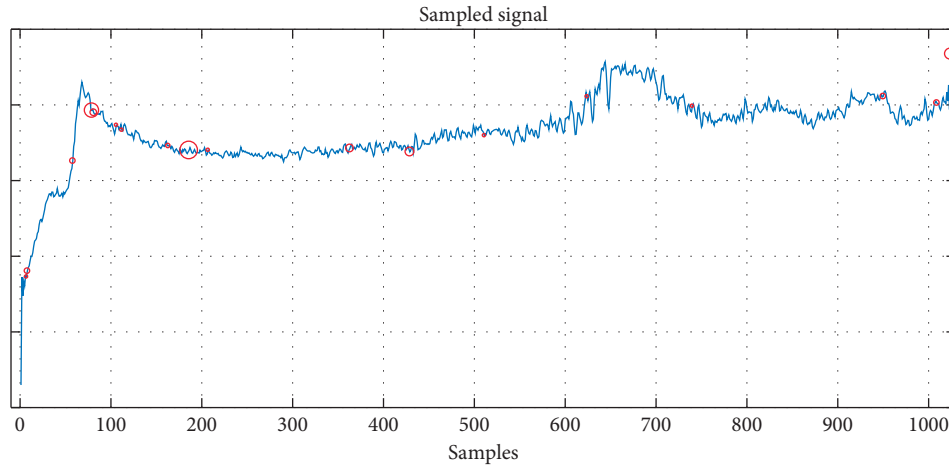


FIGURE 7: Features selected (samples marked as red circles) by the AdaBoost algorithm to perform the classification of ECE7 signals (Class 1).

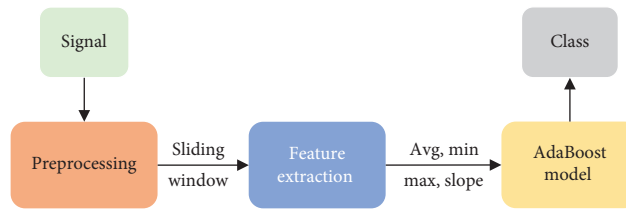


FIGURE 8: Block diagram of the online application of the approach.

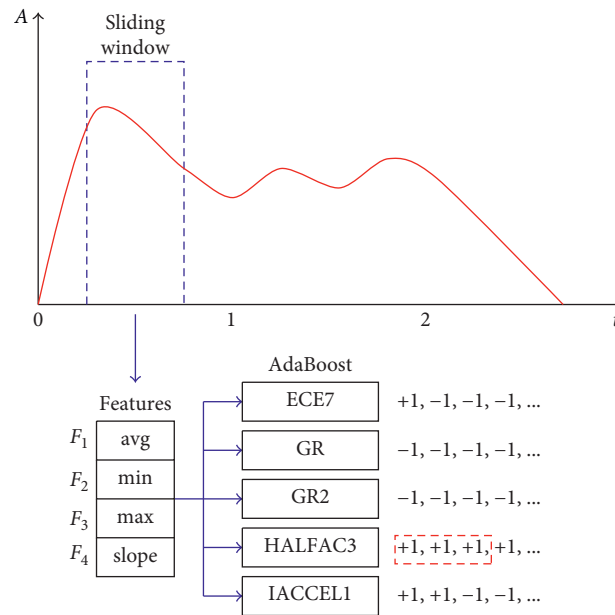


FIGURE 9: Diagram of the method.

signal. As it can be seen, almost all the 27 tested discharges for each class are correctly classified. This leads to the fact that the average success rate is over 99%.

Table 6 shows the time fraction of the discharge required to perform the online classification of the five classes for 27 randomly selected shots. The second column is the average

(in percentage) of time fraction for all shots to carry out the classification. The third column is the standard deviation of the time fraction (in percentage) needed to make the classification. The fourth and fifth are the minimum and maximum values. As it can be seen, the algorithm takes more time to classify the class HALFAC3 (0.23%) of the signal,

TABLE 4: Results of the classification of the 5 signals.

Class	Features
1	F_1, F_2, F_3
2	F_1, F_2, F_3
3	F_1, F_2, F_3, F_4
4	F_1, F_2, F_3, F_4
5	F_1, F_2, F_4

TABLE 5: Confusion matrix of the online classification.

	Predicted class				
	1	2	3	4	5
Actual class	1	27	0	0	0
	2	0	27	0	0
	3	0	0	27	0
	4	0	1	0	26
	5	0	0	0	0

TABLE 6: Results of the classification of the 5 signals.

Class	Avg (%)	Std (%)	Min (%)	Max (%)
ECE7	0.09	0.01	0.08	0.11
GR	0.14	0.22	0.08	1.21
GR2	0.09	0.01	0.08	0.11
HALFAC3	0.23	0.26	0.08	1.17
IACCEL1	0.09	0.01	0.08	0.11

which is a good result because this value is still short. The standard deviation is also short, which means that all the signals are classified with a small initial segment of each signal. The minimum value indicates that, for all classes, the algorithm never classifies a signal before 0.08% of the time. The maximum value indicates that the algorithm can classify all the signals before 1.21% fraction of the entire signals, which is a very good result.

The experiments were carried out on a PC with an Intel Core i7-8750H, 2.2 GHz, 16 GB of RAM, and Ubuntu 18.04.1 LTS operating system. For this configuration, the classification process time of each sliding window is less than 10 milliseconds (about 1 ms for feature extraction and less than 9 ms). Considering the nature of the rule-based model, this time could be clearly reduced when using embedded hardware such as field-programmable gate array (FPGA) or Application-Specific Integrated Circuit (ASIC).

4. Conclusions

This article proposes two approaches to perform a classification of five TJ-II waveforms using the ensemble algorithm AdaBoost. The first method is carried out in an offline manner and the signals are resampled to obtain distinctive attributes in the feature extraction stage. These features are translated into AdaBoost rules to classify the signals. With this method, the classifications can achieve high success rates and the classifiers are built with explicit relationships between features and rules of the AdaBoost algorithm, which allows designers to understand better the physical

underlying phenomenon. In the second approach, the classification is made for performing online classification. Firstly, the signal is preprocessed in sliding consecutive windows. Then, the feature extraction stage is performed to obtain the average, the minimum, the maximum, and the slope of the signal. These features are translated into rules of the AdaBoost algorithm that is capable of classifying the signals. The main advantage of this approach is that we do not need to wait until the discharge has finished in order to classify, which means that the classification can be performed online. Almost all the 27 tested discharges for each class are correctly classified. The average success rate is over 99%. The results show that the online classification can be performed by using only a very small fraction of the discharge.

Data Availability

The data used to support the findings of this study have not been made available.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Chilean Research and Development Agency (ANID) under the Project FONDECYT 1191188 and by the Spanish Ministry of Science and Innovation under Projects PID2019-108377RB-C31 and PID2019-108377RB-C32.

References

- [1] J. P. DeLong, O. Burger, and M. J. Hamilton, "Current demographics suggest future energy supplies will be inadequate to slow human population growth," *PLoS One*, vol. 5, no. 10, Article ID e13206, 2010.
- [2] A. Qazi, F. Hussain, N. A. Rahim et al., "Towards sustainable energy: a systematic review of renewable energy sources, technologies, and public opinions," *IEEE Access*, vol. 7, pp. 63837–63851, 2019.
- [3] D. C. Momete, "Analysis of the potential of clean energy deployment in the European union," *IEEE Access*, vol. 6, pp. 54811–54822, 2018.
- [4] J. K. Shultis and E. F. Richard, *Fundamentals of Nuclear Science and Engineering*, CRC Press, Boca Raton, FL, USA, 3rd edition, 2016.
- [5] F. I. T. Petrescu, A. Apicella, R. V. V. Petrescu et al., "Environmental protection through nuclear energy," *American Journal of Applied Sciences*, vol. 13, no. 9, pp. 941–946, 2016.
- [6] G. Farias, J. Vega, S. González et al., "Automatic determination of l/h transition times in DIII-D through a collaborative distributed environment," *Fusion Engineering and Design*, vol. 87, no. 12, pp. 2081–2083, 2012.
- [7] T. Yokoyama, T. Sueyoshi, Y. Miyoshi et al., "Disruption prediction by support vector machine and neural network with exhaustive search," *Plasma and Fusion Research*, vol. 13, p. 3405021, 2018.
- [8] G. W. Shin, J.-W. Juhn, G. I. Kwon, S. H. Son, and S. H. Hahn, "Automatic detection of L-H transition in Kstar by support

- vector machine,” *Fusion Engineering and Design*, vol. 129, pp. 341–344, 2018.
- [9] A. Murari, M. Lungaroni, E. Peluso et al., “Adaptive predictors based on probabilistic SVM for real time disruption mitigation on JET,” *Nuclear Fusion*, vol. 58, no. 5, 2018.
- [10] G. Farias, R. Dormido, M. Santos, and N. Duro, “Image classifier for the TJ-II Thomson scattering diagnostic: evaluation with a feed forward neural network,” in *Proceedings of the International Work-Conference on the Interplay between Natural and Artificial Computation*, pp. 604–612, Springer, Canary Islands, Spain, June 2005.
- [11] G. Farias, S. Dormido-Canto, J. Vega et al., “Automatic feature extraction in large fusion databases by using deep learning approach,” *Fusion Engineering and Design*, vol. 112, pp. 979–983, 2016.
- [12] G. Farias, E. Fabregas, S. Dormido-Canto et al., “Applying deep learning for improving image classification in nuclear fusion devices,” *IEEE Access*, vol. 6, pp. 72345–72356, 2018.
- [13] L. Makili, J. Vega, S. Dormido-Canto et al., “Upgrade of the automatic analysis system in the TJ-II Thomson Scattering diagnostic: new image recognition classifier and fault condition detection,” *Fusion Engineering and Design*, vol. 85, no. 3-4, pp. 415–418, 2010.
- [14] B. Cannas, P. C. de Vries, A. Fanni et al., “Automatic disruption classification in JET with the ITER-like wall,” *Plasma Physics and Controlled Fusion*, vol. 57, no. 12, Article ID 125003, 2015.
- [15] L. Rokach, “Pattern classification using ensemble methods,” *World Scientific*, vol. 75, Article ID 7238, 2010.
- [16] J. N. van Rijn, G. Holmes, B. Pfahringer, and J. Vanschoren, “The online performance estimation framework: heterogeneous ensemble learning for data streams,” *Machine Learning*, vol. 107, no. 1, pp. 149–176, 2018.
- [17] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.
- [18] M. Glugla, R. Lässer, L. Dörr, D. K. Murdoch, R. Haange, and H. Yoshida, “The inner deuterium/tritium fuel cycle of ITER,” *Fusion Engineering and Design*, vol. 69, no. 1-4, pp. 39–43, 2003.
- [19] D. Stork, *DEMO and the Route to Fusion Energy*, Karlsruhe Intl School on Fusion Technology, Karlsruhe, Germany, 2009.
- [20] P. H. Rebut, R. J. Bickerton, and B. E. Keen, “The Joint European Torus: installation, first results and prospects,” *Nuclear Fusion*, vol. 25, no. 9, pp. 1011–1022, 1985.
- [21] C. Alejaldre, J. Alonso, L. Almuoguer et al., “First plasmas in the TJ-II flexible heliac,” *Plasma Physics and Controlled Fusion*, vol. 41, no. 3A, 1999.
- [22] J. A. Boedo, D. Rudakov, R. Moyer et al., “Transport by intermittent convection in the boundary of the DIII-D tokamak,” *Physics of Plasmas*, vol. 8, no. 11, pp. 4826–4833, 2001.
- [23] National Fusion Laboratory of Spain (CIEMAT), *TJ-II Project*, CIEMAT, 2021, <http://www.fusion.ciemat.es/tj-ii-2/>, Madrid, Spain.
- [24] R. Rojas, “AdaBoost and the super bowl of classifiers a tutorial introduction to adaptive boosting,” Technical Report, Freie Universität, Berlin, Germany, 2009.
- [25] S. Dormido-Canto, G. Farias, R. Dormido et al., “TJ-II wave forms analysis with wavelets and support vector machines,” *Review of Scientific Instruments*, vol. 75, no. 10, pp. 4254–4257, 2004.
- [26] G. Farias, S. Dormido-Canto, J. Vega, I. Martínez, L. Alfaro, and F. Martínez, “AdaBoost classification of TJ-II thomson scattering images,” *Fusion Engineering and Design*, vol. 123, pp. 759–763, 2017.
- [27] J. Vega, C. Crémy, E. Sánchez, A. Portas, J. A. Fabregas, and R. Herrera, “Data management in the TJ-II multi-layer database,” *Fusion Engineering and Design*, vol. 48, no. 1–2, pp. 69–75, 2000.
- [28] E. Sánchez, A. B. Portas, and J. Vega, “A relational database for physical data from TJ-II discharges,” *Fusion Engineering and Design*, vol. 60, no. 3, pp. 341–346, 2002.