AI-FLARES
WP4100-D1: Report on the training phase for the machine learning and deep learning methods applied to HMI data

For details see the following AI-FLARES papers

- Guastavino S, Marchetti F, Benvenuto F, Campi C, Piana M. Operational solar flare forecasting via video-based deep learning. arXiv preprint arXiv:2209.05128. 2022 Sep 12.
- Guastavino S, Marchetti F, Benvenuto F, Campi C, Piana M. Implementation paradigm for supervised flare forecasting studies: A deep learning application with video data. Astronomy & Astrophysics. 2022 Jun 1;662:A105.
- Cicogna D, Berrilli F, Calchetti D, Del Moro D, Giovannelli L, Benvenuto F, Campi C, Guastavino S, Piana M. Flare-forecasting algorithms based on high-gradient polarity inversion lines in active regions. The Astrophysical Journal. 2021 Jul 1;915(1):38.
- Benvenuto F, Campi C, Massone AM, Piana M. Machine learning as a flaring storm warning machine: Was a warning machine for the 2017 September solar flaring storm possible? The Astrophysical Journal Letters. 2020 Nov 19;904(1):L7.

Section 1 - Prediction algorithms

AI-FLARES activity concerned with flare forecasting relied on two different conceptual approaches. Indeed, we developed both feature-based machine learning algorithms that used as input flare descriptors extracted from full disk magnetograms of solar active regions, and a deep learning network that can take as input either images or videos of these same magnetograms. Figure 1 shows the architecture used by AI-FLARES in the case of feature-based approaches. The flare prediction algorithms utilized within the project's framework are random forest and a hybrid unsupervised/supervised version of LASSO.

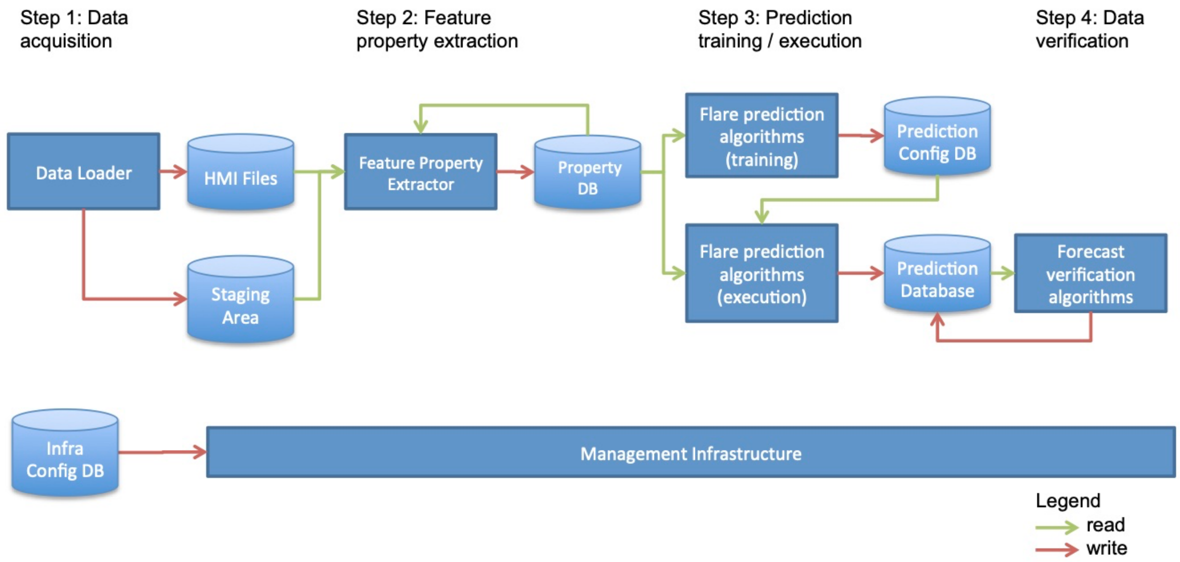

Figure 1: AI-FLARES architecture for feature-based machine learning algorithms

Figure 2 describes the deep learning architecture, which is based on a long-term recurrent convolutional network (LRCN) made of a convolutional neural network (CNN) and a long short-term memory (LSTM) network. Specifically, the CNN is made of the following sequence of layers: a 7×7 convolutional layer of 32 units; a 2×2 max-pooling layer; a 5×5 convolutional layer of 32 units; a 2×2 max-pooling layer; a 3×3 convolutional layer of 32 units; a 2×2 max-pooling layer; a 3×3 convolutional layer of 32 units; a 2×2 max-pooling layer; a dense layer of 64 units with dropout. Height and width strides were set to 2 for the convolutional layers and to 1 for the max-pooling. Each convolutional layer was L2-regularized and the corresponding output was standardized. Before applying the dense layer, the last pooling layer was flattened. The Rectified Linear Unit (ReLU) was used as an activation function in all layers. We also point out that the input videos, which consisted of 40 frames of 128 × 128 magnetograms each, were treated as time series, so that the CNN architecture described above was applied to each video frame in parallel. The outputs of the CNNs (40 vectors, each one composed by 64 features) were passed to the LSTM consisting of 50 units (also in this case dropout was applied). Finally, a dense sigmoid unit drove the output of the LSTM to be in the interval [0, 1], in order to perform binary classification.

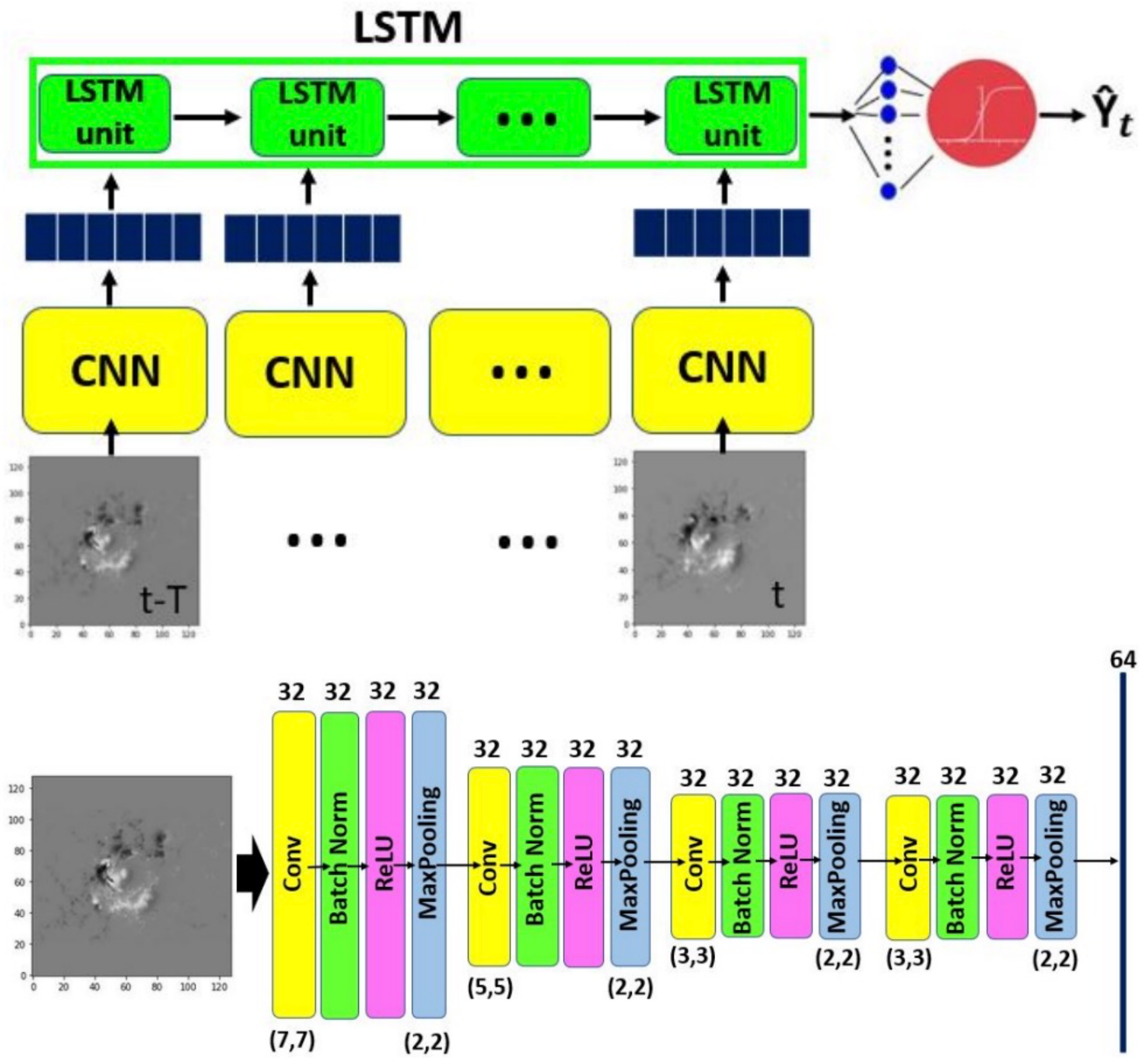

Figure 2. The AI-FLARE deep learning architecture for flare forecasting.

Section 2 - Training phase

Both the feature-based machine learning algorithms and the CNN-LSTM network were trained according to an epoch-based techniques, and the Adam scheme was adopted for the optimization of the weights. In this training phase, a crucial role is played by the choice of the loss function. AI-FLARES introduced a completely innovative approach to the design of loss functions, which is based on probabilistic confusion matrices. AI-FLARES score-oriented loss (SOL) functions allow an automated optimization of a given skill score without the need of an a posteriori choice of the optimal threshold that converts the probabilistic outcomes into binary classification. Specifically, a very effective SOL function is based on the optimization of the TSS, which is highly insensitive to the class imbalance ratio in the training set. The realization of this score-driven strategy is performed as follows. The classical confusion matrix depends on a fixed threshold parameter $\tau \in (0, 1)$, meaning that,

$$\mathrm{CM}(\tau) = \begin{pmatrix} \mathrm{TN}(\tau) & \mathrm{FP}(\tau) \\ \mathrm{FN}(\tau) & \mathrm{TP}(\tau) \end{pmatrix}.$$

For the construction of SOL functions, the threshold parameter τ is dealt with as a random variable associated with a specific probability density function. Letting $E_\tau[\cdot]$ be the expected value with respect to τ, we took an expected confusion matrix

$$\mathbb{E}_\tau[\mathrm{CM}(\tau)] = \begin{pmatrix} \mathbb{E}_\tau[\mathrm{TN}(\tau)] & \mathbb{E}_\tau[\mathrm{FP}(\tau)] \\ \mathbb{E}_\tau[\mathrm{FN}(\tau)] & \mathbb{E}_\tau[\mathrm{TP}(\tau)] \end{pmatrix}.$$

From this matrix it was possible to construct the expected TSS

$$\mathbb{E}_\tau[\mathrm{TSS}(\tau)] = \frac{\mathbb{E}_\tau[\mathrm{TP}(\tau)]}{\mathbb{E}_\tau[\mathrm{TP}(\tau) + \mathrm{FN}(\tau)]} - \frac{\mathbb{E}_\tau[\mathrm{FP}(\tau)]}{\mathbb{E}_\tau[\mathrm{FP}(\tau) + \mathrm{TN}(\tau)]} - 1,$$

and from this the TSS-driven loss function

$$\ell_{\mathrm{TSS}} := -\mathbb{E}_\tau[\mathrm{TSS}(\tau)].$$

This function is differentiable, and therefore can be easily minimized in the training phase. It has the crucial advantage that the corresponding skill score is automatically optimized, without the need of any a posteriori tuning of the thresholding parameter τ, which is set to the default value 0.5.

Section 3 - Prediction assessment

Flare forecasting is an intrinsically dynamic prediction problem. Therefore, SOL function for training neural networks can be designed in order to automatically optimize skill-scores that account for this dynamical nature of flare forecasting. AI-FLARES introduced a novel approach to evaluate the severity of prediction errors by considering that a false alarm predicting that an event will occur just before its actual occurrence, anyway eases the right decision from the user, while a delayed alarm is of little use. The idea is to exploit the sequential order, naturally given by the time, with which the prediction occurs in order to assign a cost for false positives and false negatives. In this way, we classify errors on the basis of their importance and we get a novel notion of confusion matrix and related skill scores. In this new framework, the severity of errors depends on their impact on the decision-making process and therefore we referred to these novel confusion matrix and skill scores as value-weighted. Such a strategy allows for evaluating the forecast value independently of a cost–benefit analysis.