scientific reports

Check for updates

OPEN

Prediction of severe thunderstorm events with ensemble deep learning and radar data

Sabrina Guastavino[1]✉, Michele Piana[1,2], Marco Tizzi[3], Federico Cassola[3], Antonio Iengo[3], Davide Sacchetti[3], Enrico Solazzo[4] & Federico Benvenuto[1]

The problem of nowcasting extreme weather events can be addressed by applying either numerical methods for the solution of dynamic model equations or data-driven artificial intelligence algorithms. Within this latter framework, the most used techniques rely on video prediction deep learning methods which take in input time series of radar reflectivity images to predict the next future sequence of reflectivity images, from which the predicted rainfall quantities are extrapolated. Differently from the previous works, the present paper proposes a deep learning method, exploiting videos of radar reflectivity frames as input and lightning data to realize a warning machine able to sound timely alarms of possible severe thunderstorm events. The problem is recast in a classification one in which the extreme events to be predicted are characterized by a an high level of precipitation and lightning density. From a technical viewpoint, the computational core of this approach is an ensemble learning method based on the recently introduced value-weighted skill scores for both transforming the probabilistic outcomes of the neural network into binary predictions and assessing the forecasting performance. Such value-weighted skill scores are particularly suitable for binary predictions performed over time since they take into account the time evolution of events and predictions paying attention to the value of the prediction for the forecaster. The result of this study is a warning machine validated against weather radar data recorded in the Liguria region, in Italy.

One of the most interesting problems in weather forecasting is the prediction of extreme rainfall events such as severe thunderstorms possibly leading to flash floods. This problem is very challenging especially when we consider areas characterized by a complex, steep orography close to a coastline, where intense precipitation can be enhanced by specific topographic features: this is the case for the example of Liguria, an Italian region located on the northwest Mediterranean Sea and characterized by the presence of mountains over 2000 m high at only a few kilometres away from the coastline. This specific morphology gives rise to several catchments with steep slopes and limited extension[1]. Autumn events, when deep Atlantic troughs more easily enter the Mediterranean area and activate very moist and unstable flow lifted by the mountain range, may cause catastrophic flooding on these coastal areas, which are characterized by a high population density (see[2,3] for a review of the climatology and typical atmospheric configurations of extreme precipitation over the Mediterranean area). Just as an example, the November 4th 2011 flood in Genoa caused six deaths and economic damage up to 100 million euros[4–7]). A common feature in these extreme events is the presence of a quasi-stationary convective system with a spatial extension of few kilometers[8–12]

Medium and long range either deterministic or ensemble Numerical Weather Prediction (NWP) models still struggle to correctly predict both the intensity and the location of these events, which can be triggered and enhanced by very small-scale features. High resolution convection-permitting NWP models manage to partly return a more realistic description of the dynamics of severe thunderstorms. Many studies addressed the role played by different components or settings of NWP models in order to better describe severe convective systems over the Liguria area, such as model resolution, initial conditions, microphysics schemes or small-scale patterns of the sea surface temperature[6,13–19].

However, the intrinsically limited predictability of convective systems requires the use of shorter-term *nowcasting* models, e.g. in order to feed automatic early warning systems, which may support meteorologists and hydrologists in providing accurate and reliable forecasts and thus reducing the consequences of these extreme events. These forecasting systems typically rely on two kinds of approaches. On the one hand, either stochastic

[1]The MIDA Group, Dipartimento di Matematica, Università di Genova, Genova, Italy. [2]CNR - SPIN Genova, Genova, Italy. [3]ARPAL, Genova, Italy. [4]ARPA Lombardia, Milan, Italy. ✉email: guastavino@dima.unige.it

or deterministic models are formulated utilizing partial differential equations in fluid dynamics, and numerical methods are implemented for their reduction, nesting hydrological models into meteorological ones[20–22]. On the other hand, more recent data-driven techniques take as input a time series of radar (and in case satellite) images belonging to a historical archive and provide as output a synthetic image representing the prediction of the radar signal at a subsequent time point; this approach can rely on some extrapolation technique, e.g. based on a storm-tracking system[23] or a diffusive process in Fourier space[24], or on deep learning networks[25–36]. Mixed techniques have been also proposed, blending NWP outputs with data-driven synthetic predictions[37]. The aim of these studies is to make time series prediction by exploiting image-based deep learning techniques, such as U-net[26], Convolutional Long Short-Term Memory (ConvLSTM)[28], improvements of ConvLSTM as Trajectory Gated Recurrent Unit (TrajGRU)[30,33,34], and Generative Adversial Networks (GANs)[35,36], which produce reflectivity images in the next future. From the predicted reflectivity images the rainfall quantity can be extrapolated but no indication of the presence of lightning can be provided. In our work, we focus on the forecasting of extreme thunderstorm events and therefore previous methods mentioned above do not directly apply to our problem. On the contrary, we present a novel method which recasts the problem into a classification one by using the lightning density as a fundamental feature for characterizing an extreme event. Towards this aim, we exploit a deep neural network, originally conceived for video classification, to predict the probability that an extreme event occurs. We use as input time series of multichannel radar images and we define the labels on the basis of a certain level of precipitation and lightning density. The deep-learning model combines a convolutional neural network (CNN) with a long short-term memory (LSTM) network[38,39] in order to construct a long-term recurrent convolutional network (LRCN)[40]. The prediction assessment is performed by means of the recently introduced value-weighted skill scores[41] which allows ranking prediction errors on the basis of their distribution along time, preferring to show up a warning well in advance of the actual occurrence of an event rather than not to show it at all. Finally, we exploit the iterative nature of the network training process to collect a set of predictions from which we select a subset of valuable ones on the basis of their value-weighted skill score. This procedure falls within the class of ensemble learning techniques. We remark that the term "ensemble" as used here refers to deep learning methods and not to the NWP algorithms. The main methodological novelties of this approach are the following.

1. The prediction problem is reformulated into a binary classification one in which labels depend on both heavy rainfall conditions and lightning density;
2. forecasting verification is performed by the use of value-weighted skill scores on the basis of an automatic ensemble strategy.

Other works have been translated the forecasting problem into a binary prediction , but the focus was on moderate rain, i.e. when the rainfall is beyond a certain threshold, mainly > 5 mm/h or at most > 30 mm/h. To our knowledge, the present work is the first attempt to predict severe thunderstorm events on the basis of lightnings and radar video data. Moreover, forecast verification is completely different with respect to previous works. Usually, skill scores compare the predictions with observations in a time independent way, i.e, a score remains unchanged if we permute the temporal order of events and predictions in the same way. On the contrary, the value-weighted skill scores take into account the time evolution of events and predictions paying attention on the value of the prediction for the forecaster. Indeed, this approach provides probabilistic outcomes concerning the event occurrence and related quantitative parameters, thus realizing an actual warning machine for the forecasting of extreme events. The results of this study is a data-driven warning system for supporting the decision making in the case of extreme rainfall events tailored for the Ligurian region. This system takes advantage of the value-weighted skill scores which, in the framework of an ensemble learning approach, allow the deep network to provide predictions more accurate than those obtained when standard quality-based skill scores are applied.

The paper is organized as follows. In "Constant altitude plan position indicator reflectivity data in Liguria" section we describe the considered weather radar and lightning data, and in "Long-term recurrent convolutional network" section we give details on the architecture of the LRCN model used in this study. In "Ensemble deep learning" section we recall the definition of value-weighted skill scores, and we describe the proposed ensemble deep learning technique. In "Experimental results" section we show the effectiveness of the method in prediction of extreme rainfall events using radar-based data. Our conclusions are offered in "Conclusions and future work" section.

Constant altitude plan position indicator reflectivity data in Liguria

Precipitation activity and locations of rain, showers, and thunderstorms are commonly monitored in real-time by polarimetric Doppler weather radars; return echoes from targets (such as hydrometeors) allow the measurement of the reflectivity field on different conical surfaces, one at each elevation angle of the radar; however, reflectivity values at a certain height can be interpolated to 2D maps, which are also known as Constant Altitude Plan Position Indicator (CAPPI) images[42]; such a representation is particularly useful for compositing reflectivity data measured by different radars over overlapping regions, returning a reflectivity field for the larger area covered by a radar network.

In our study CAPPI reflectivity fields measured by the Italian Radar Network within the Civil Protection Department are considered. CAPPI images, measured in dBZ, are sampled every 10 minutes at a spatial resolution of $0.005^\circ \simeq 0.56$ km in latitude and $0.005^\circ \simeq 0.38$ km in longitude. We used CAPPI images at three different heights (2 km, 3 km, and 5 km above sea level (ASL)) and cut each image over an area comprising the Liguria region (as shown in Fig. 1). In detail, for each image the latitude ranges in [43.4° N, 45.0° N] and the longitude ranges in [7.1° E, 10.4° E], so that images have size 321×661 and cover an area of about 180 km in latitude and

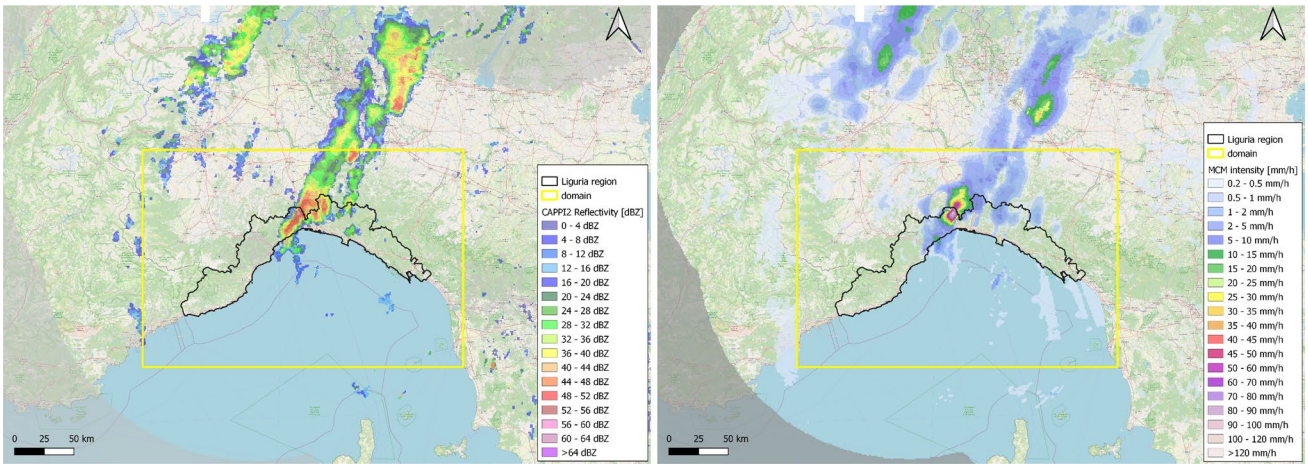


Figure 1. An example of a 2-km CAPPI reflectivity frame (left) and an MCM rain rate frame (right) (both referred to 21/10/2019 23:00 UTC); the selected area surrounding Liguria is delimited in yellow. The maps are downloaded from OpenStreetMap.

250 km in longitude. We used 1.5-hour-long movies of CAPPI images to construct temporal feature sequences to predict the occurrence of extreme rainfall event in the hour after the last time step.

The training set exploited to optimize the LRCN is generated by means of a labeling procedure involving modified conditional merging (MCM) data and lightning data. MCM data[43] combine radar rain estimates and rain gauge measurements with an hourly frequency and provide the amount of rainfall integrated over 1 hour (in these data the content of each pixel is measured in mm per hour and the spatial resolution is $0.013267^\circ \simeq 1$ km in longitude and $0.008929^\circ \simeq 1$ km in latitude; see Fig. 1). Lightning data are recorded by the LAMPINET network of Military Aeronautics[44] and have a resolution of 1 microsecond.

The labeling process associates each CAPPI video to the concept of severe convective rainfall event, whose definition relies on the following two items:

- MCM data must contain at least 3 contiguous pixels exceeding 50 mm/h within the selected area;
- at least 10 lightning strikes must occur in a 10-minute time range in the area comprising 5 km around each one of the MCM pixels with over-threshold content.

It is worth noticing that 50 mm/h is regarded as a threshold for heavy rain in the Liguria region; however, the first condition accounts for the fact that an over-threshold value associated to an isolated pixel may be associated to spurious non-meteorological echoes like, for instance, the passage of a plane. On the other hand, the second condition implies that the extreme events considered must always involve the occurrence of thunderstorms.

Long-term recurrent convolutional network

The idea of this work is to address the prediction of extreme events in the short term as a radar image video classification problem. Following the work of[40] we propose the use of a Long-term recurrent convolutional network (LRCN) which combines a convolutional neural network (CNN) and a long short-term memory (LSTM) network to create spatio-temporal deep learning models[45,46]. In this application, the input is made of time series of 10 radar reflectivity images (representing a video 1.5 hours long) at the three CAPPI 2, CAPPI 3 and CAPPI 5 levels, which refer to 2 km, 3 km and 5 km ASL, respectively. Images have been resized to a 128×256 pixel size in order to guarantee a good trade-off between computational efficiency and image resolution. The CNN is used to automatically extract spatial features from the image set. The features are decomposed into sequential components and fed to the LSTM network to be analyzed. Finally, the output of the LSTM layer is fed into the fully connected layer, and the sigmoid activation function is applied to generate the probability distribution of the positive class. Figure 2 shows the architecture of the LRCN model.

In this work the CNN architecture of the LRCN model consists in three blocks, each one composed by a convolutional layer with stride (2, 2), followed by a batch normalization layer to improve stability; the Rectified Linear Unit (ReLU) function[47] is adopted as an activation function and the max pooling operation with size (4,4) and stride (2, 2) is applied. We initialize all the convolutional weights by sampling from the scaled uniform distribution[48]. The three convolutional layers are characterized by 8, 16 and 32 kernels with size (5, 5), (3, 3) and (3, 3), respectively. The input are sequences of size (T, 128, 256, 3), where T represents the number of frames in each movie, 128 and 256 correspond to the image size (in pixel) and 3 represents the three levels of CAPPI data. In all operations we take advantage of the "Timedistributed" layer, available in the Keras library[49], which allows the in parallel training of the T convolutional flows. Figure 3 illustrates this CNN architecture. Then, the CNN output is flattened to create the sequence of feature vectors to feed into the LSTM network. In our experiments, the LSTM layer has 50 hidden neurons. Finally, the dropout layer is used to prevent overfitting[50]: the dropout value is set to 0.5, meaning that 50% of neurons are randomly dropped from the neural network during training

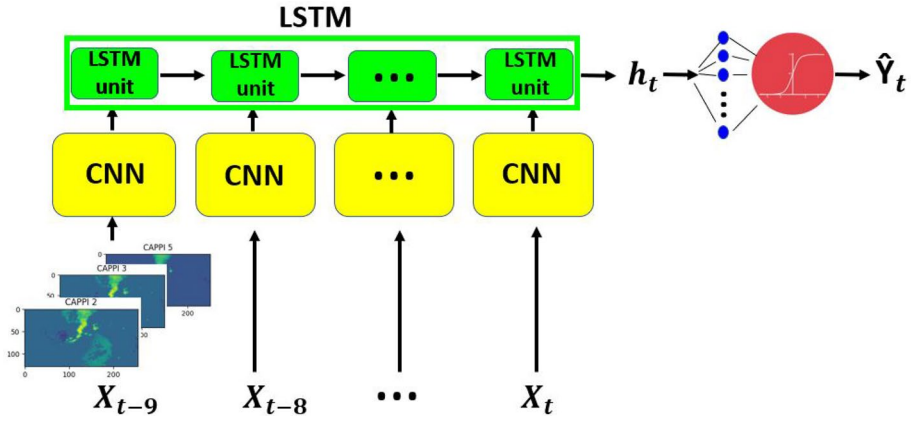


Figure 2. The LRCN architecture.

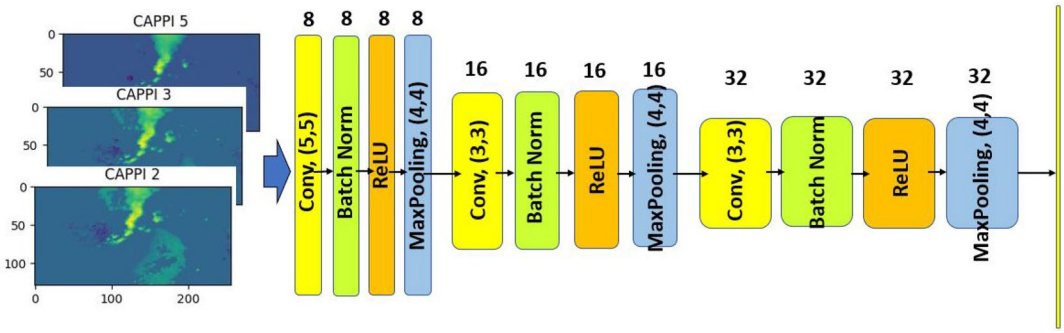


Figure 3. The CNN architecture.

in each iteration. The hyperparameters of the NN are estimated by an empirical trial-and-error optimization process on several experiments.

Loss function. Once the architecture of the NN is set up, we can denote with θ the NN weights and we can interpret the NN as a map f_θ, mapping a sample X to a probability outcome $f_\theta(X) \in [0,1]$, since the sigmoid activation function is applied in the last layer. We recall that, in our application, the sample X is a video of CAPPI reflectivity images and $f_\theta(X)$ represents the predicted probability of the occurrence of an extreme rainfall event in the next hour after the end time of the CAPPI video X within the selected area (in fact, we are not interested in the exact location of the possible event). In the training process we consider an optimization problem

$$\min_\theta \ell(F_\theta(\mathbf{X}), \mathbf{Y}), \tag{1}$$

where $\{\mathbf{X}, \mathbf{Y}\} = \{(X_i, Y_i)\}_{i=1}^n$ is the training set (Y_i represents the actual label of the sample X_i according to the definition given in "Constant altitude plan position indicator reflectivity data in Liguria") Section, $F_\theta(\mathbf{X}) = (f_\theta(X_i))_i$ represents the probability outcomes of the NN on the set $\mathbf{X}$ and ℓ represents the loss function measuring the discrepancy between the true label $\mathbf{Y}$ and the predicted output $F_\theta(\mathbf{X})$. In classification problems the most used loss function is the binary cross-entropy. In the case of imbalanced data sets, modifications of the cross-entropy loss are considered, such as the following one:

$$\ell(F_\theta(\mathbf{X}), \mathbf{Y}) = -\left(\sum_{i=1}^n \beta_1 Y_i \log(f_\theta(X_i)) + \beta_0 (1 - Y_i) \log(1 - f_\theta(X_i))\right), \tag{2}$$

where β_0, β_1 are positive weights defined according to the data set imbalance. We define the weights as

$$\beta_1 = \frac{1}{\#\{i \in \{1, \ldots, n\} : Y_i = 1\}} \text{ and } \beta_0 = \frac{1}{\#\{i \in \{1, \ldots, n\} : Y_i = 0\}}, \tag{3}$$

where $Y_i = 1$ indicates the presence of extreme rainfall event and $Y_i = 0$ indicates the absence of extreme rainfall event. We refer to the chosen loss function as the class balanced cross-entropy.

Ensemble deep learning

During the iterative optimization process a set of deep neural networks $X \to f_\theta(X)$ by varying of θ is generated. The proposed ensemble deep learning technique selects a subset of this set as follows. For each θ, it transforms the probabilistic outcome $f_\theta(X)$ of f_θ into a binary prediction and then it evaluates on the validation set such a prediction according to its value-weighted skill score. To describe this strategy in detail we start by the value-weighted skill score..

Evaluation skill scores. The result of a binary classifier is usually evaluated by computing the confusion matrix, also known as the contingency table. Let us denote with $\mathbb{M}_{2,2}(\mathbb{N})$ the set of 2-dimensional matrices with natural elements. Let $\mathbf{Y} = (Y_i) \in \{0,1\}^n$ be a binary sequence representing the actual labels of a given dataset of examples, and let $\hat{\mathbf{Y}} = (\hat{Y}_i) \in \{0,1\}^n$ be a binary sequence representing the prediction. Then the classical (quality-based) confusion matrix $\tilde{\mathbf{C}} \in \mathbb{M}_{2,2}(\mathbb{N})$ is given by:

$$\tilde{\mathbf{C}}(\hat{\mathbf{Y}}, \mathbf{Y}) = \begin{pmatrix} \mathrm{TN} & \mathrm{FP} \\ \mathrm{FN} & \mathrm{TP} \end{pmatrix},$$

where $\mathrm{TP} = \sum_{i=1}^{n} \mathbb{1}_{\{Y_i=1,\hat{Y}_i=1\}}$ represents the true positives, i.e. the number of samples correctly classified as the positive class; $\mathrm{TN} = \sum_{i=1}^{n} \mathbb{1}_{\{Y_i=0,\hat{Y}_i=0\}}$ represents the true negatives, i.e. the number of samples correctly classified as the negative class; $\mathrm{FP} = \sum_{i=1}^{n} \mathbb{1}_{\{Y_i=0,\hat{Y}_i=1\}}$ represents the false positives, i.e. the number of negative samples incorrectly classified as the positive class; $\mathrm{FN} = \sum_{i=1}^{n} \mathbb{1}_{\{Y_i=1,\hat{Y}_i=0\}}$ represents the false negatives, i.e. the number of positive samples incorrectly classified as the negative class.

A specific classical (quality-based) skill score is given by a map $\mathrm{S} : \mathbb{M}_{2,2}(\mathbb{N}) \to \mathbb{R}$ defined on the confusion matrix $\tilde{\mathbf{C}}$. In this study we considered two skill-scores, i.e., the critical success index (CSI)

$$\mathrm{CSI}(\tilde{\mathbf{C}}(\hat{\mathbf{Y}}, \mathbf{Y})) = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}, \tag{4}$$

which is commonly used in meteorological applications[34]; and the true skill statistic (TSS)

$$\mathrm{TSS}(\tilde{\mathbf{C}}(\hat{\mathbf{Y}}, \mathbf{Y})) = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} - \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \tag{5}$$

which is particularly appropriate for imbalanced data sets[51]. The CSI varies from $[0, 1]$, while the TSS varies from $[-1, 1]$ and for both scores the optimal value is 1.

However, such metrics do not account for the distribution of predictions along time and are not able to provide a quantitative preference to those alarms that predict an event well in advance with respect to its actual occurrence, and to penalize predictions sounding delayed false alarms. To overcome these limitations, value-weighted confusion matrices have been introduced[41]. The aim of the value-weighted approach is to mitigate errors such as false positives that precede false negatives (the case of predictions well in advance) and false negatives which are preceded by true positives (the case of on going events already predicted) as they have little impact on the prediction from the point of view of the forecaster. In fact, a value-weighted confusion matrix is defined as

$$\mathbf{C}_{\mathrm{w}}(\hat{\mathbf{Y}}, \mathbf{Y}) = \begin{pmatrix} \mathrm{TN} & \mathrm{wFP} \\ \mathrm{wFN} & \mathrm{TP} \end{pmatrix}, \tag{6}$$

with

$$\mathrm{wFP} = \sum_{i=1}^{n} w(z_i^-, z_i^+) \mathbb{1}_{\{Y_i=0,\hat{Y}_i=1\}}, \tag{7}$$

$$\mathrm{wFN} = \sum_{i=1}^{n} w(\hat{z}_i^+, \hat{z}_i^-) \mathbb{1}_{\{Y_i=1,\hat{Y}_i=0\}}. \tag{8}$$

where the weights $w(z_i^-, z_i^+)$ and $w(z_i^-, z_i^+)$ are constructed as follows. First, the function w is

$$w(s,t) = \begin{cases} 2 & \text{if } s, t \equiv 0 \\ 1 - \max(\mathrm{w} \circ t) & \text{otherwise} \end{cases} \tag{9}$$

where $\mathrm{w} := \left(\frac{1}{2}, \frac{1}{3}, \ldots, \frac{1}{T+1}\right)$ and $\mathrm{w} \circ t$ indicates the element-wise product. Second, given the label Y_i observed at the sampled time i, then $z_i^- = (Y_{i-1}, Y_{i-2}, \ldots, Y_{i-T})$, is the sequence of the T elements before Y_i and $z_i^+ = (Y_{i+1}, Y_{i+2}, \ldots, Y_{i+T})$ is the sequence of the T elements after Y_i. Analogously, given the label $\hat{Y}_i$ predicted at time i, then $\hat{z}_i^- = (\hat{Y}_{i-1}, \hat{Y}_{i-2}, \ldots, \hat{Y}_{i-T})$, and $\hat{z}_i^+ = (\hat{Y}_{i+1}, \hat{Y}_{i+2}, \ldots, \hat{Y}_{i+T})$. The weight function $w : \mathbb{R}^T \times \mathbb{R}^T \to \mathbb{R}$ is then constructed in such a way to emphasize false positives associated with alarms predicted in the middle of $2T + 1$-long time windows when no actual event occurs and false negatives associated with missed events in the middle of $2T + 1$-long time windows in which no alarm is raised.

The introduction of this value-weighted confusion matrix allows the construction of the associated value-weighted Critical Success Index wCSI and the value-weighted True Skill Statistic wTSS, respectively.

Ensemble strategy. We consider an ensemble procedure to provide an automatic classifier from the probability outcomes provided by the deep NN. Consider the first N epochs of the training process of the deep neural network f_θ. Denote with $\theta_j := \theta_j(\{\mathbf{X}, \mathbf{Y}\})$ the neural netwrork weights for each epoch j computed from the training set. The procedure has been introduced in[41], and it can be summarized in the following steps:

1. For each epoch j we select the classification threshold $\overline{\tau}_j$, i.e. the real number that maximizes a given skill score

$$\overline{\tau}_j = \arg\max_{\tau\in[0,1]} \mathrm{S}(\mathbf{C}(P^{\tau}_{\theta_j}(\mathbf{X}), \mathbf{Y}))). \tag{10}$$

where $P^{\tau}_{\theta_j}(\mathbf{X}) := (\mathbf{1}_{\{f_{\theta_j}(X_i)>\tau\}})_{i=1,\dots,n}$ is the binary prediction on the set of samples $\mathbf{X}$ and $\mathbf{1}_{\{\cdot\}}$ denotes the indicator function. Then, we denote by

$$\overline{P}_{\theta_j}(\mathbf{X}) := P^{\overline{\tau}}_{\theta_j}(\mathbf{X}) \tag{11}$$

the binary prediction on the set $\mathbf{X}$ obtained by using the optimized threshold value.
2. Choose the subset of valuable predictions by selecting the predictors with a skill score higher than a given a quality level α on the validation set $\{\tilde{\mathbf{X}}, \tilde{\mathbf{Y}}\} = \{(\tilde{X}_i, \tilde{Y}_i)\}_{i=1}^{m}$, i.e

$$\mathscr{J}_\alpha := \{j \in \{1, \dots, N\} : \mathrm{S}(\mathbf{C}(\overline{P}_{\theta_j}(\tilde{\mathbf{X}}), \tilde{\mathbf{Y}}))) > \alpha\}. \tag{12}$$

3. We define the ensemble prediction as the median value of all the selected predictions. Given a new sample X, we have

$$\hat{Y}^{\theta} = m(\{\overline{P}_{\theta_j}(X) : j \in \mathscr{J}_\alpha\}). \tag{13}$$

where m indicates the median function. In the case where the number of zeros is equal to the number of ones, we assume $\hat{Y}^{\theta} = 1$.

In the second step of the above scheme, the parameter α in Eq. (12) has to be given. Differently from[41], where the above procedure was introduced and α was arbitrarily chosen, we propose to compute it as follows.

(i) For each $\gamma \in [\gamma_0, \gamma_1)$ with $0 < \gamma_0 < \gamma_1 < 1$, consider the epochs for which the skill score S computed on the validation set is higher than a given fraction γ of the maximum possible score S on the validation set by varying epochs

$$\mathscr{J}_\gamma := \{j \in \{1, \dots, N\} : \mathrm{S}(\mathbf{C}(\overline{P}_{\theta_j}(\tilde{\mathbf{X}}), \tilde{\mathbf{Y}})) > \gamma \max_{l\in\{1,\dots,N\}} \{\mathrm{S}(\mathbf{C}(\overline{P}_{\theta_l}(\tilde{\mathbf{X}}), \tilde{\mathbf{Y}}))\}\}. \tag{14}$$

and compute the corresponding ensemble prediction on the validation set

$$\hat{\mathbf{Y}}^{\theta}_{\gamma} = m(\{\overline{P}_{\theta_j}(\tilde{\mathbf{X}}) : j \in \mathscr{J}_\gamma\}). \tag{15}$$

(ii) Select the optimal parameter $\overline{\gamma}$ as the one which maximizes the skill score S computed on the validation set

$$\overline{\gamma} := \arg\max_{\gamma\in[\gamma_0,\gamma_1)} \mathrm{S}(\mathbf{C}(\hat{\mathbf{Y}}^{\theta}_{\gamma}, \tilde{\mathbf{Y}})) \tag{16}$$

and define the level α as follows

$$\alpha := \overline{\gamma} \max_{j\in\{1,\dots,N\}} \{\mathrm{S}(\mathbf{C}(\overline{P}_{\theta_j}(\tilde{\mathbf{X}}), \tilde{\mathbf{Y}}))\}. \tag{17}$$

As a result of this procedure, the estimated value of α only depends on the validation set.

We show the pipeline diagram explaining the ensemble method in Fig. 4.

In order to ensure statistical robustness of the entire ensemble procedure, we repeat it M times by randomizing the initial values of the weights, i.e. by training the deep neural network M times and we take the best ensemble prediction on the validation set. The best prediction is in the sense of the highest preferred skill score S. Therefore, by denoting with $\theta^{(k)}$ the weights of the trained deep neural network at the k-th random initialization, we define the optimal weights as

$$\overline{\theta} := \arg\max_{(\theta^{(k)})_{k=1,\dots,M}} \mathrm{S}'(\mathbf{C}(\hat{\mathbf{Y}}^{\theta^{(k)}}_{\overline{\gamma}}, \tilde{\mathbf{Y}})), \tag{18}$$

where $\hat{\mathbf{Y}}^{\theta^{(k)}}_{\overline{\gamma}}$ is the ensemble prediction on the validation set obtained at the k-th random initialization of the training process.

In the following we show the performance of the ensemble deep learning technique when the LRCN network is used for the problem of forecasting extreme rainfall events in Liguria.

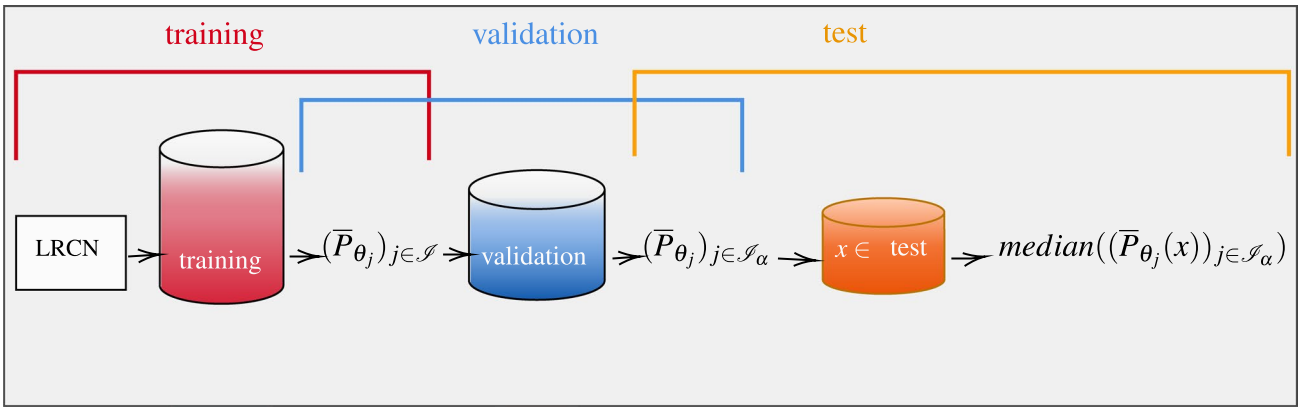


Figure 4. Pipeline diagram for the ensemble method. The first step consists in training the LRCN model (the architecture is shown in Fig. 2) over a fixed number of epochs and computing the classification thresholds defined in (10): the outputs of the training process are the estimators $(\overline{P}_{\theta_j})_{j\in\mathscr{I}}$ (see (11)) where $\mathscr{I}$ is the set of epoch indexes. The second step consists in validating the estimators $(\overline{P}_{\theta_j})_{j\in\mathscr{I}}$ by selecting the ones which provide predictions on the validation set with scores over a level α, which is determined through the procedure described in Eqs. (14)–(17). The final step consists in testing the method on a new input: the prediction of the ensemble method is given by computing the median of the estimators $(\overline{P}_{\theta_j})_{j\in\mathscr{I}}$ applied on a new input x.

Strategy	Confusion matrix		TSS	CSI	wFP	wFN	wTSS	wCSI
wTSS	TN = $1725.40_{(\pm 21.98)}$	FP = $140.60_{(\pm 21.98)}$	$0.78_{(\pm 0.04)}$	$0.17_{(\pm 0.02)}$	$243.88_{(\pm 41.34)}$	$6.79_{(\pm 1.64)}$	$0.68_{(\pm 0.04)}$	$0.10_{(\pm 0.02)}$
	FN = $4.70_{(\pm 1.25)}$	TP = $28.30_{(\pm 1.25)}$						
TSS	TN = $1727.60_{(\pm 32.42)}$	FP = $138.40_{(\pm 32.42)}$	$0.77_{(\pm 0.05)}$	$0.17_{(\pm 0.03)}$	$240.99_{(\pm 60.57)}$	$7.24_{(\pm 2.60)}$	$0.67_{(\pm 0.06)}$	$0.10_{(\pm 0.02)}$
	FN = $5.10_{(\pm 1.85)}$	TP = $27.90_{(\pm 1.85)}$						

Table 1. Results on the test set obtained by using the TSS-ensemble and wTSS-ensemble strategies. The entries are the average values of the scores over 10 runs of the network for 10 random initializations of the weights. The standard deviations are also included.

Experimental results

In order to assess the prediction reliability of our deep NN model, we considered a historical dataset of CAPPI composite reflectivity videos recorded by the Italian weather radar network in the time window ranging from 2018/07/09 at 21:30 UTC to 2019/12/31 at 12:00 UTC, each video being 90 minutes long. For the training phase, we considered the time range from 2018/07/09 at 21:30 UTC to 2019/07/16 at 10:30 UTC and label the videos with binary labels concerning the concurrent occurrence of an over-threshold rainfall event from MCM data and lightning strikes in its surroundings, as explained in "Constant altitude plan position indicator reflectivity data in Liguria" section. The training set contains 7128 samples overall, with 105 samples labeled with 1, i.e. corresponding to extreme events according to the definition given in "Constant altitude plan position indicator reflectivity data in Liguria" section. For the validation step, we considered the videos in the time range from 2019/07/19 at 14:30 UTC to 2019/09/30 at 12:30 UTC (the validation set is made of 1296 videos overall, with 48 videos labeled with 1). Eventually, the test set is made of the CAPPI videos in the time range between 2019/10/03 at 15:00 UTC and 2019/12/31 at 12:00 UTC (the test contains 1899 videos, and 33 of them are labeled with 1). The model is trained over $N = 100$ epochs using the Adam Optimizer[52] with learning rate equal to 0.001 and mini-batch size equal to 72. The class balanced cross-entropy defined in (2) is used as the loss function in the training phase, where the weights β_0 and β_1 are defined as the inverse of the number of samples labeled with 0 and with 1 in each mini-batch, respectively.

As explained in "Ensemble deep learning" section, the statistical significance of the results is guaranteed by running the network $M = 10$ times, each time with a different random initialization of the LRCN weights. We report in Fig. 5 the training and validation loss per epoch for the 10 runs. We noticed that the validating loss curves have more fluctuations for some runs especially after 60 epochs: this is most probably due to the fact that the training and validation sets have different percentages of samples labeled with 1 for the chronological splitting. Finally, we applied the ensemble strategy as described in "During the iterative optimization proc" Section, using the TSS and wTSS for choosing the epochs with best performance. For sake of clarity, for now on the two ensemble strategies will be named as TSS-ensemble and wTSS-ensemble, respectively.

These two strategies have been applied to the test set, and the results are illustrated in Table 1, where we reported the average values and the corresponding standard deviations for the entries of the quality-based and value-weighted confusion matrices, and for the TSS, CSI, wTSS, and wCSI. The table shows that the score values

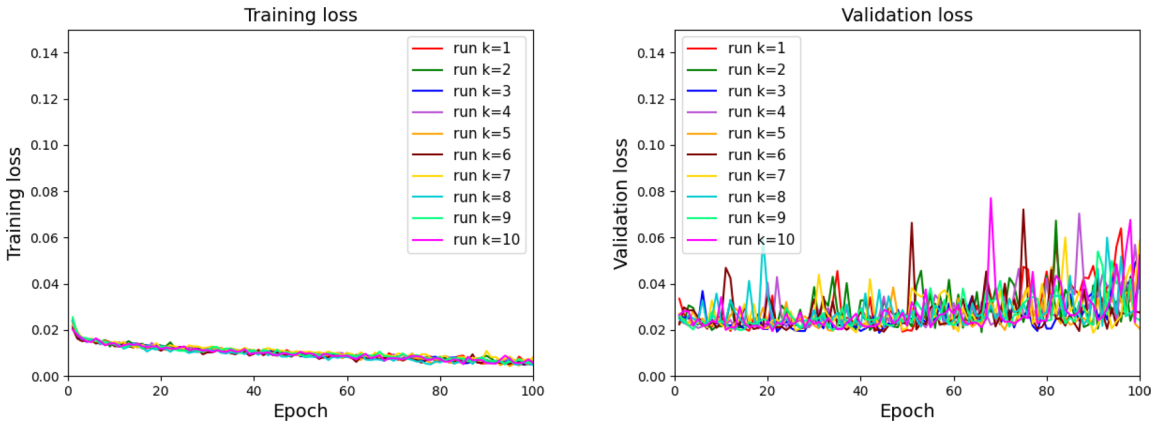


Figure 5. Learning curves showing the behaviour of the training (left panel) and validation (right panel) loss along epochs for the ten runs.

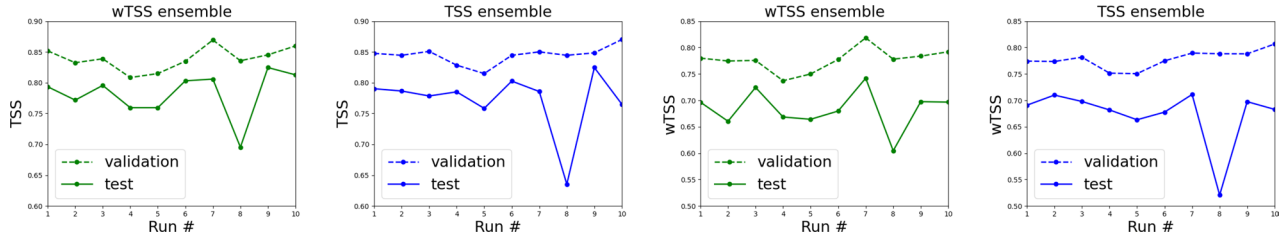


Figure 6. From right to left: TSS values on the validation set (dashed lines) and test set (continuous lines) obtained on each run by applying the wTSS-ensemble strategy (first panel) and the TSS-ensemble strategy (second panel); wTSS values on the validation set (dashed lines) and test set (continuous lines) obtained on each run by applying the wTSS-ensemble strategy (third panel) and the TSS-ensemble strategy (fourth panel).

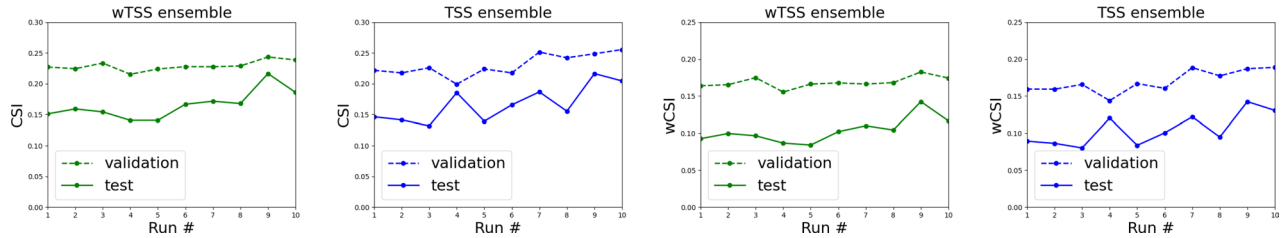


Figure 7. From left to right: CSI values on the validation set (dashed lines) and test set (continuous lines) obtained on each run by applying the wTSS-ensemble strategy (first panel) and the TSS-ensemble strategy (second panel); wCSI values on the validation set (dashed lines) and test set (continuous lines) obtained on each run by applying the wTSS-ensemble strategy (third panel) and the TSS-ensemble strategy (fourth panel).

are all rather similar, although the averaged TSS and wTSS values are slightly higher when the wTSS-ensemble strategy is adopted.

Since, according to the ensemble strategy, the prediction for a specific test set is made by using the weights corresponding to the best run in the validation set, in Fig. 6 we show the behavior of TSS and wTSS for the TSS-ensemble and wTSS-ensemble strategies, in the case of 10 runs of the network corresponding to 10 random initializations of the weights.

The results in Fig. 6 imply that, in the case of the wTSS-ensemble strategy, the best score values in validation correspond to the best score values in the test phase. Figure 7 illustrates the same analysis in the case when the scores used for assessing the prediction performance are CSI and wCSI and shows that, also in this case, the wTSS-ensemble strategy should be preferred. We pointed out that the gap between validation and test scores is most probably due to the heterogeneity of the data used in training, validation and test sets: the test set represents mainly the autumnal period whereas the validation comprises mainly data of the summer period. We think that a better practice could be using data of the autumnal period of many past years for training and validating the network in order to have a better prediction on the next autumn.

Table 2 contains the values of confusion-matrix entries and scores obtained by using the weights associated to the best runs of the network selected during the validation phase by means of the TSS-ensemble and

Score	Strategy					
	wTSS ensemble				TSS ensemble	
	S' =TSS/wTSS (run $k = 7$)		S' =CSI/wCSI (run $k = 9$)		S' =TSS/wTSS/CSI/wCSI (run $k = 10$)	
Confusion matrix	TN = 1730	FP = 136	TN = 1765	FP = 101	TN = **1767**	FP = **99**
	FN = **4**	TP = **29**	FN = **4**	TP = **29**	FN = 6	TP = 27
TSS	0.8059		**0.8247**		0.7651	
CSI	0.1716		**0.2164**		0.2045	
wFN	**4.75**		8		8	
wFP	229.83		**166.58**		171.67	
wTSS	**0.742**		0.6975		0.6829	
wCSI	0.11		**0.1425**		0.1306	

Table 2. Results on the test set obtained by using the wTSS-ensemble strategy when the run is selected with respect to the best TSS or wTSS ($k = 7$ run), the wTSS-ensemble strategy when the run is selected with respect to the best CSI or wCSI ($k = 9$ run) and the TSS-ensemble strategy when the run is selected with respect to the best TSS or wTSS or CSI or wCSI ($k = 10$ run). In bold the best results are highlighted.

wTSS-ensemble strategies. Please consider that in the case of the TSS-ensemble strategy, the best run is always the $k = 10$ one.

In order to show how the use of value-weighted scores performs in action, in Fig. 8 we enrolled over time the predictions corresponding to the test set, when the wTSS-ensemble and TSS-ensemble strategies are adopted and when wTSS, TSS, wCSI and CSI are used for selecting the best run (we point out again that using wTSS and TSS for the wTSS-ensemble strategy always leads to $k = 7$ and that using wCSI and CSI for the same ensemble strategy always leads to $k = 9$).

We remind that the labeling procedure depends on the rain rate and on the presence of lighting, as described in "Constant altitude plan position indicator reflectivity data in Liguria" Section. The blue bars represent the events labeled with 1, i.e. events which satisfy the condition on both the rain rate and the presence of lighting, whereas the green bars are events that satisfy only the condition on the rain rate.

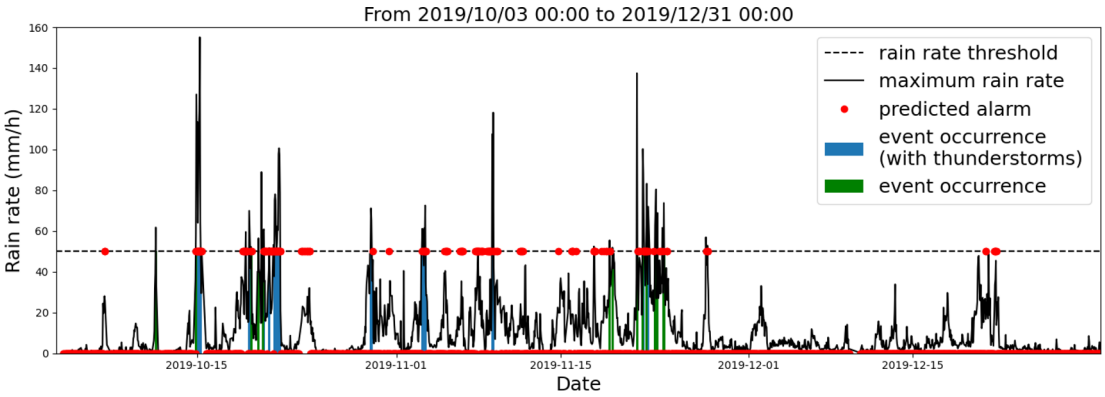
We first point out that when the wTSS-strategy is used and $k = 7$ is selected, the prediction tends to systematically anticipate the events characterized by high rain rate. Further, for sake of clarity, Fig. 9 contains a zoom around the November 22 2019 time point, when a dramatic flood caused significant damage in many areas of Liguria. This zoom shows that the wTSS-ensemble strategy for $k = 7$ is able to correctly predict the thunderstorms occurring in the time interval from 00:00 to 02:00 UTC on November 22 2019 and to anticipate the other catastrophic thunderstorm occurring between 10:00 and 11:00 UTC (this last thunderstorm is marked with a blue arrow in all panels of Fig. 9). No anticipated alarm is sounded by the other two predictions.

Conclusions and future work

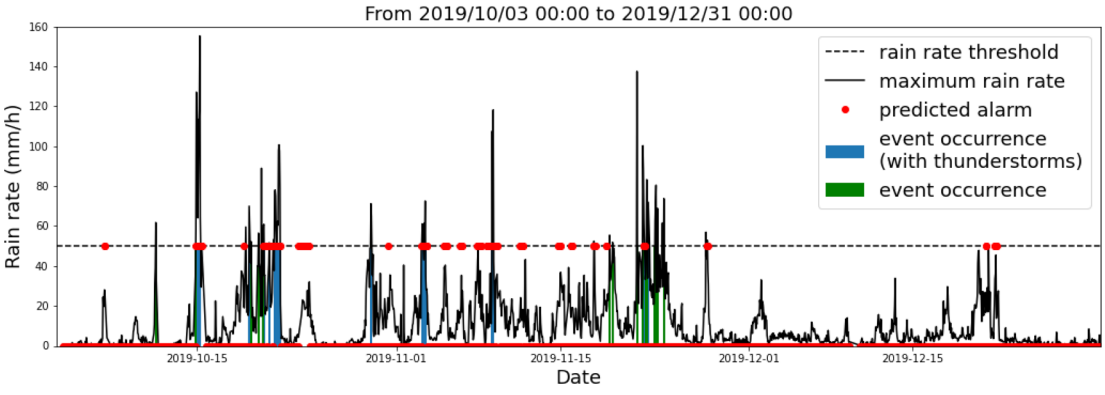
The realization of warning machines able to sound binary alarms along time is an intriguing issue in many areas of forecasting[53–56]. The present paper shows for the first time that a deep CNN exploiting radar videos as input can be used as a warning machine for predicting severe thunderstorms (in fact, previous CNNs in this field have been used to synthesize simulated radar images at time points successive to the last one in the input time series). It is worth noticing that the aim here is not the prediction of the exact location and intensity of a heavy rain event, but rather the probable occurrence of a severe thunderstorm over a reference area in the next hour.

The crucial point in our approach relies on the kind of evaluation metrics adopted. In fact, the TSS can be considered a good measure of performance in forecasting, since it is insensitive to the class-imbalance ratio. However, such a skill score, as all the ones computed on a classical quality-based confusion matrix, does not account for the temporal distribution of alarms. Therefore, we propose to focus on value-weighted skill scores, as the wTSS, which account for the distribution of the predictions over time while promoting predictions well in advance. We focused on the problem of forecasting extreme rainfall events on the Liguria region, and we showed that the performance of our ensemble technique in the case when wTSS is optimized, is significantly better than the performance when the model is trained to optimize a standard quality-based score.

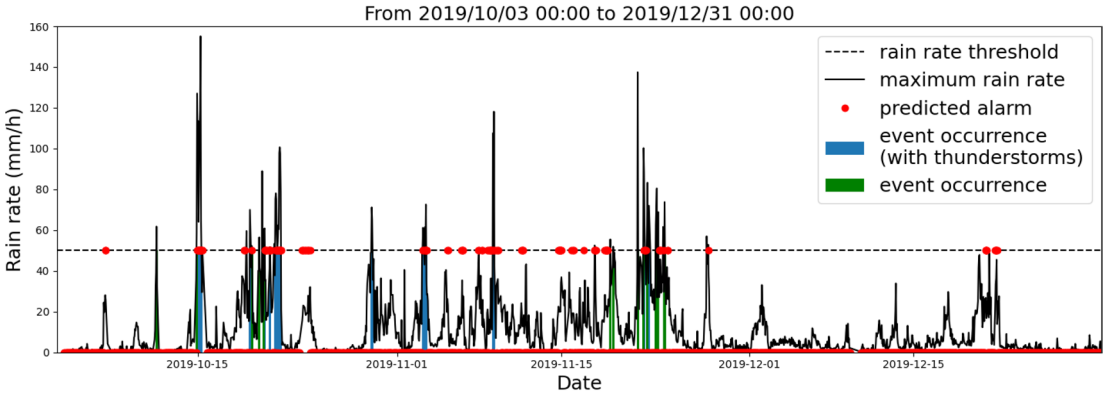
Next in line in our work will be the application of a class of score-driven loss functions[57], whose minimization in the training phase allows the automatic maximization of the corresponding skill scores. Possible future studies of this work concern (1) the investigation of other ensemble techniques as[58,59], (2) the use of feature selection methods which allow individuating the most relevant subset of features extracted by CNN models as in[60], (3) the use of dynamic graph modeling approaches to learn spatial-temporal representations in radar reflectivity videos[61]. Further, deep hashing methods[62] could be used to exploit more information for the prediction, like the density and type of lightning (such as cloud-to-cloud and cloud-to-ground strikes).



(a) wTSS-ensemble ($k = 7$)

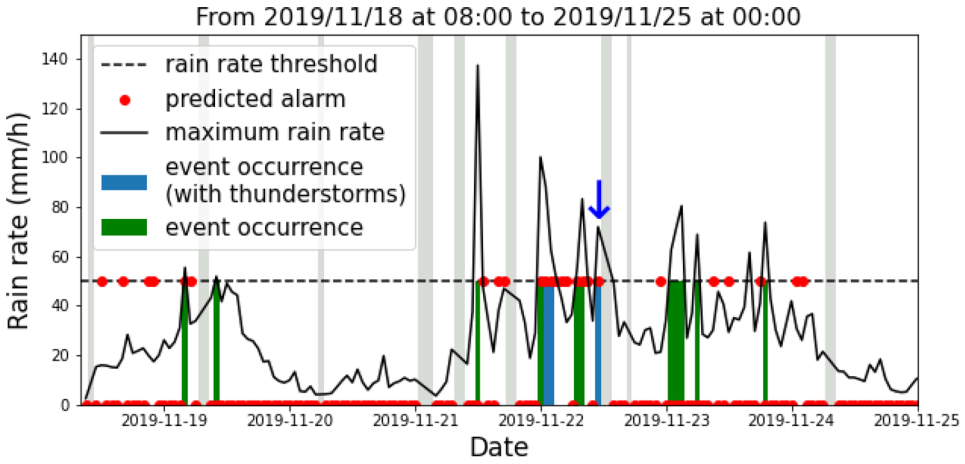


(b) wTSS-ensemble ($k = 9$)

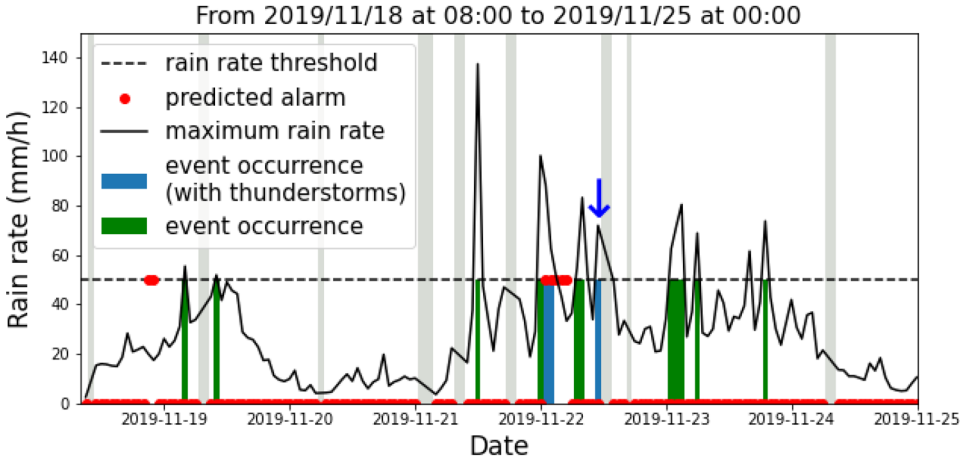


(c) TSS-ensemble ($k = 10$)

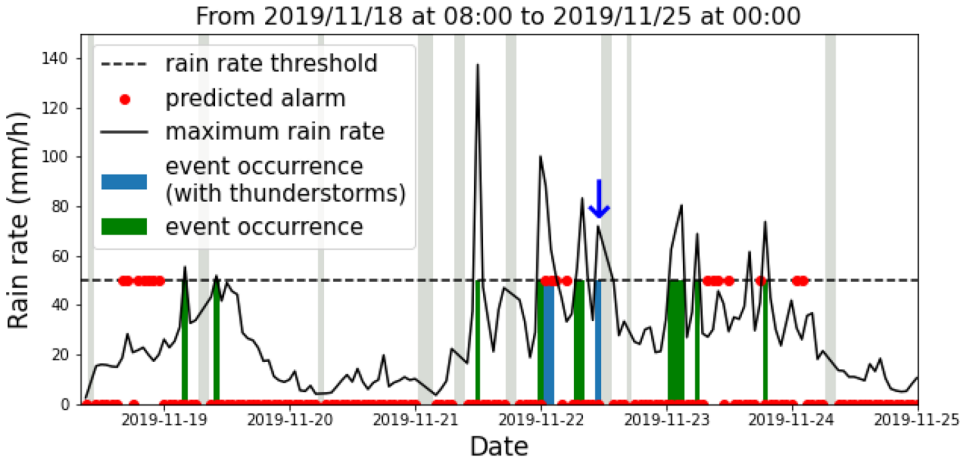
Figure 8. Predictions for the testing period obtained by applying the wTSS-ensemble strategy at $k = 7$ run (top panel), the wTSS-ensemble strategy at $k = 9$ run (central panel) and the TSS-ensemble strategy at $k = 10$ run (bottom panel).



(a) wTSS-ensemble ($k = 7$)



(b) wTSS-ensemble ($k = 9$)



(c) TSS-ensemble ($k = 10$)

Figure 9. Predictions valid from 2019/11/18 at 08:00 UTC to 2019/11/25 at 00:00 UTC obtained by applying the wTSS-ensemble strategy at $k = 7$ run (top panel), the wTSS-ensemble strategy at $k = 9$ run (central panel) and the TSS-ensemble strategy at $k = 10$ run (bottom panel). The grey boxes correspond to times when the input data are missing.

Data availability

The data that support the findings of this study are available from the Italian Civil Protection Department (radar data) and the Italian Military Aeronautic (lightnings data) but restrictions apply to the availability of these data, which were used under license for the current study, and so are not publicly available. Data are however available from the authors upon reasonable request and with permission of the Italian Civil Protection Department (radar data) and the Italian Military Aeronautic (lightnings data). However, the radar data can be downloaded from https://mappe.protezionecivile.gov.it/it/mappe-rischi/piattaforma-radar and we put at disposal the code of the deep neural network and the ensemble procedure in the github repository https://github.com/SabrinaGuastavino/Ensemble-deep-learning.

Received: 4 August 2022; Accepted: 29 October 2022
Published online: 21 November 2022

References

1. Pensieri, S., Schiano, M. E., Picco, P., Tizzi, M. & Bozzano, R. Analysis of the precipitation regime over the Ligurian Sea. *Water*-https://doi.org/10.3390/w10050566 *(2018).*
2. Ricard, D., Ducrocq, V. & Auger, V. A climatology of the mesoscale environment associated with heavily precipitating events over a Northwestern Mediterranean area. *J. Appl. Meteorol. Climatol.* **51**, 468–488 (2012).
3. Dayan, U., Nissen, K. & Ulbrich, U. Review article: Atmospheric conditions inducing extreme precipitation over the eastern and western Mediterranean. *Nat. Hazards Earth Syst. Sci.* **15**, 2525–2544 (2015).
4. Faccini, F., Luino, F., Sacchini, A. & Turconi, L. Flash flood events and urban development in Genoa (Italy): Lost in translation. In *Engineering Geology for Society and Territory*, Vol. 5 797–801 (Springer, Cham, 2015).
5. Silvestro, F. *et al.* A hydrological analysis of the 4 November 2011 event in Genoa. *Nat. Hazards Earth Syst. Sci.* **12**, 2743–2752 (2012).
6. Buzzi, A., Davolio, S., Malguzzi, P., Drofa, O. & Mastrangelo, D. Heavy rainfall episodes over Liguria of autumn 2011: Numerical forecasting experiments. *Nat. Hazards Earth Syst. Sci.* **14**, 1325–1340 (2014).
7. Fiori, E. *et al.* Analysis and hindcast simulation of an extreme rainfall event in the Mediterranean area: The Genoa 2011 case. *Atmos. Res.* **138**, 13–29 (2014).
8. Delrieu, G. *et al.* The catastrophic flash-flood event of 8–9 september 2002 in the Gard Region, France: A first case study for the Cévennes-Vivarais Mediterranean Hydrometeorological Observatory. *Nat. Hazards Earth Syst. Sci.* **6**, 34–52 (2005).
9. Rebora, N. *et al.* Extreme rainfall in the Mediterranean: What can we learn from observations?. *J. Hydrometeorol.* **14**, 906–922 (2013).
10. Cassola, F., Ferrari, F. & Mazzino, A. Numerical simulations of Mediterranean heavy precipitation events with the wrf model: A verification exercise using different approaches. *Atmos. Res.* **164–165**, 3–18 (2015).
11. Silvestro, F., Rebora, N., Giannoni, F., Cavallo, A. & Ferraris, L. The flash flood of the Bisagno Creek on 9th October 2014: An "unfortunate" combination of spatial and temporal scales. *J. Hydrol.* **541**, 50–62. https://doi.org/10.1016/j.jhydrol.2015.08.004 (2016). Flash floods, hydro-geomorphic response and risk management.
12. Davolio, S., Silvestro, F. & Gastaldo, T. Impact of rainfall assimilation on high-resolution hydrometeorological forecasts over Liguria, Italy. *J. Hydrometeorol.* **18**, 2659–2680 (2017).
13. Fiori, E. *et al.* Triggering and evolution of a deep convective system in the Mediterranean Sea: Modelling and observations at a very fine scale. *Q. J. R. Meteorol. Soc.* **143**, 927–941. https://doi.org/10.1002/qj.2977 (2017).
14. Meroni, A. N., Parodi, A. & Pasquero, C. Role of sst patterns on surface wind modulation of a heavy midlatitude precipitation event. *J. Geophys. Res. Atmos.* **123**, 9081–9096 (2018).
15. Lagasio, M., Silvestro, F., Campo, L. & Parodi, A. Predictive capability of a high-resolution hydrometeorological forecasting framework coupling WRF cycling 3DVAR and continuum. *J. Hydrometeorol.* **20**, 1307–1337. https://doi.org/10.1175/JHM-D-18-0219.1 (2019).
16. Davolio, S., Silvestro, F. & Malguzzi, P. Effects of increasing horizontal resolution in a convection permitting model on flood forecasting: The 2011 dramatic events in Liguria (Italy). *J. Hydrometeorol.* **16**, 1843–1856 (2015).
17. Ferrari, F. *et al.* Impact of model resolution and initial/boundary conditions in forecasting flood-causing precipitations. *Atmosphere* **11**, 592 (2020).
18. Cassola, F., Ferrari, F., Mazzino, A. & Miglietta, M. M. The role of the sea on the flash floods events over Liguria (northwestern Italy). *Geophys. Res. Lett.* **43**, 3534–3542 (2016).
19. Ferrari, F., Cassola, F., Tuju, P. & Mazzino, A. Rans and les face to face for forecasting extreme precipitation events in the liguria region (northwestern Italy). *Atmos. Res.* **259**, 105654. https://doi.org/10.1016/j.atmosres.2021.105654 (2021).
20. Han, S. & Coulibaly, P. Bayesian flood forecasting methods: A review. *J. Hydrol.* **551**, 340–351 (2017).
21. Blöschl, G., Reszler, C. & Komma, J. A spatially distributed flash flood forecasting model. *Environ. Model. Softw.* **23**, 464–478 (2008).
22. Kauffeldt, A., Wetterhall, F., Pappenberger, F., Salamon, P. & Thielen, J. Technical review of large-scale hydrological models for implementation in operational flood forecasting schemes on continental level. *Environ. Model. Softw.* **75**, 68–76 (2016).
23. Hering, A., Morel, C., Galli, G., Ambrosetti, P. & Boscacci, M. Nowcasting thunderstorms in the alpine region using a radar based adaptive thresholding scheme. In *Proceedings, Third ERAD Conference, Visby, Sweden* 206–211 (2004).
24. Silvestro, F. & Rebora, N. Operational verification of a framework for the probabilistic nowcasting of river discharge in small and medium size basins. *Nat. Hazard.* **12**, 763–776. https://doi.org/10.5194/nhess-12-763-2012 (2012).
25. Ayzel, G. *et al. Proceedings of the 13th International Symposium "Intelligent Systems 2018" (INTELS'18), 22–24 October, 2018, St. Petersburg, Russia*. https://doi.org/10.1016/j.procs.2019.02.036 (2019).
26. Ayzel, G., Scheffer, T. & Heistermann, M. Rainnet v1.0: A convolutional neural network for radar-based precipitation nowcasting. *Geosci. Model Dev.* **13**, 2631–2644. https://doi.org/10.5194/gmd-13-2631-2020 (2020).
27. Samsi, S., Mattioli, C. J. & Veillette, M. S. Distributed deep learning for precipitation nowcasting. In *2019 IEEE High Performance Extreme Computing Conference (HPEC)* 1–7 (IEEE, 2019).
28. Shi, X. *et al.* Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Proceedings of the 28th International Conference on Neural Information Processing Systems, Volume 1, NIPS'15* 802–810 (MIT Press, 2015).
29. Heye, A., Venkatesan, K. & Cain, J. E. Precipitation nowcasting: Leveraging deep recurrent convolutional neural networks (2017).
30. Tran, Q.-K. & Song, S.-K. Computer vision in precipitation nowcasting: Applying image quality assessment metrics for training deep neural networks. *Atmosphere*https://doi.org/10.3390/atmos10050244 *(2019).*
31. Bonnet, S. M., Evsukoff, A. & Morales Rodriguez, C. A. Precipitation nowcasting with weather radar images and deep learning in São Paulo, Brasil. *Atmosphere*https://doi.org/10.3390/atmos11111157 *(2020).*
32. Czibula, G., Mihai, A. & Mihuleţ, E. Nowdeepn: An ensemble of deep learning models for weather nowcasting based on radar products' values prediction. *Appl. Sci.*https://doi.org/10.3390/app11010125 *(2021).*

33. Shi, X. *et al.* Deep learning for precipitation nowcasting: A benchmark and a new model. In *Advances in Neural Information Processing Systems* Vol. 30 (eds Guyon, I. *et al.*) (Curran Associates Inc, 2017).
34. Franch, G. *et al.* Precipitation nowcasting with orographic enhanced stacked generalization: Improving deep learning predictions on extreme events. *Atmosphere*https://doi.org/10.3390/atmos11030267 *(2020).*
35. Choi, S. & Kim, Y. Rad-cgan v1.0: Radar-based precipitation nowcasting model with conditional generative adversarial networks for multiple dam domains. *Geosci. Model Dev* **15**, 5967–5985 (2022).
36. Ravuri, S. *et al.* Skilful precipitation nowcasting using deep generative models of radar. *Nature* **597**, 672–677 (2021).
37. Poletti, M. L., Silvestro, F., Davolio, S., Pignone, F. & Rebora, N. Using nowcasting technique and data assimilation in a meteorological model to improve very short range hydrological forecasts. *Hydrol. Earth Syst. Sci.* **23**, 3823–3841. https://doi.org/10.5194/hess-23-3823-2019 (2019).
38. Le, X.-H., Ho, H. V., Lee, G. & Jung, S. Application of long short-term memory (lstm) neural network for flood forecasting. *Water* **11**, 1387 (2019).
39. Van Houdt, G., Mosquera, C. & Nápoles, G. A review on the long short-term memory model. *Artif. Intell. Rev.* **53**, 5929–5955 (2020).
40. Donahue, J. *et al. Long-Term Recurrent Convolutional Networks for Visual Recognition and Description*. Retrieved 30 August 2019 (2019).
41. Guastavino, S., Piana, M. & Benvenuto, F. Bad and good errors: Value-weighted skill scores in deep ensemble learning. *IEEE Trans. Neural Netw. Learn. Syst.*https://doi.org/10.1109/TNNLS.2022.3186068 *(2022).*
42. Atlas, D. *Radar in Meteorology: Battan Memorial and 40th Anniversary Radar Meteorology Conference* (Springer, 2015).
43. Bruno, G. *et al.* Performing hydrological monitoring at a national scale by exploiting rain-gauge and radar networks: The Italian case. *Atmosphere*https://doi.org/10.3390/atmos12060771 *(2021).*
44. Biron, D. Lampinet-lightning detection in Italy. In *Lightning: Principles, Instruments and Applications* 141–159 (Springer, Dordrecht, 2009).
45. Donahue, J. *et al.* Long-term recurrent convolutional networks for visual recognition and description. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**, 677–691. https://doi.org/10.1109/TPAMI.2016.2599174 (2017).
46. Guastavino, S., Marchetti, F., Benvenuto, F., Campi, C. & Piana, M. Implementation paradigm for supervised flare forecasting studies: A deep learning application with video data. *Astron. Astrophys.* **662**, A105 (2022).
47. Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016).
48. Glorot, X. & Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, vol. 9 of Proceedings of Machine Learning Research* (eds Teh, Y. W. & Titterington, M.) 249–256 (PMLR, Chia Laguna Resort, 2010).
49. Chollet, F. Keras, GitHub. https://github.com/fchollet/keras (2015).
50. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).
51. Bloomfield, D. S., Higgins, P. A., McAteer, R. J. & Gallagher, P. T. Toward reliable benchmarking of solar flare forecasting methods. *Astrophys. J. Lett.* **747**, L41 (2012).
52. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. In *Proceedings of 3rd International Conference on Learning Representations* (2015).
53. Chang, M.-J. *et al.* A support vector machine forecasting model for typhoon flood inundation mapping and early flood warning systems. *Water* **10**, 1734 (2018).
54. Benvenuto, F., Campi, C., Massone, A. M. & Piana, M. Machine learning as a flaring storm warning machine: Was a warning machine for the 2017 September solar flaring storm possible?. *Astrophys. J. Lett.* **904**, L7 (2020).
55. Zhang, Z. & Chen, Y. Tail risk early warning system for capital markets based on machine learning algorithms. *Comput. Econ.* **60**, 901–923 (2021).
56. Li, H., Li, C. & Liu, Y. Machine learning-based frequency security early warning considering uncertainty of renewable generation. *Int. J. Electr. Power Energy Syst.* **134**, 107403 (2022).
57. Marchetti, F., Guastavino, S., Piana, M. & Campi, C. Score-oriented loss (sol) functions. *Pattern Recogn.*https://doi.org/10.1016/j.patcog.2022.108913 *(2022).*
58. Pramanik, R. *et al.* A fuzzy distance-based ensemble of deep models for cervical cancer detection. *Comput. Methods Programs Biomed.* **219**, 106776. https://doi.org/10.1016/j.cmpb.2022.106776 (2022).
59. Paul, A., Pramanik, R., Malakar, S. & Sarkar, R. An ensemble of deep transfer learning models for handwritten music symbol recognition. *Neural Comput. Appl.* **34**, 10409–10427 (2022).
60. Pramanik, R., Sarkar, S. & Sarkar, R. An adaptive and altruistic pso-based deep feature selection method for pneumonia detection from chest x-rays. *Appl. Soft Comput.* **128**, 109464. https://doi.org/10.1016/j.asoc.2022.109464 (2022).
61. Zhu, W., Han, Y., Lu, J. & Zhou, J. Relational reasoning over spatial-temporal graphs for video summarization. *IEEE Trans. Image Process.* **31**, 3017–3031. https://doi.org/10.1109/TIP.2022.3163855 (2022).
62. Ma, C., Lu, J. & Zhou, J. Rank-consistency deep hashing for scalable multi-label image search. *IEEE Trans. Multimed.* **23**, 3943–3956 (2020).

Acknowledgements

SG was financially supported by a regional grant of the 'Fondo Sociale Europeo', Regione Liguria. MP and FB acknowledge the financial contribution from the agreement ASI-INAF n.2018-16-HH.0. We acknowledge the Italian Civil Protection Department, CIMA Research Foundation and the Italian Military Aeronautic for providing CAPPI radar data, MCM rainfall estimates and lightning data. We also acknowledge the support of a scientific agreement between ARPAL and the Dipartimento di Matematica, Università di Genova. This work has been supported by the EU Programme Interreg V-A France-Italie ALCOTRA 2014-2020 through the PITEM RISK GEST project and by the EU Programme Interreg Marittimo IT-FR 2014-2020 through the GIAS project.

Author contributions

S.G., F.B. and M.P. conceived the study, the ensemble method and the value-weighted skill scores. S.G. implemented the codes. M.T., F.C., A.I, D.S. and E.S. provided the data. S.G., F.B. and M.P. wrote the manuscript. M.T., F.C., A.I, D.S. and E.S. contributed to the introduction, the data description and they provided Fig. 1 of the manuscript. S.G. made all the other graphical representations. All authors discussed and analyzed the results.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to S.G.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.