



## Prediction Interval of Interface Regions: Machine Learning Nowcasting Approach

**Special Section:**  
Machine Learning in  
Heliophysics

Khaled Alielden<sup>1</sup> , Enrico Camporeale<sup>2,3</sup> , Marianna B. Korsós<sup>1,4,5</sup>, and Youra Taroyan<sup>1</sup>

<sup>1</sup>Department of Physics, Aberystwyth University, Ceredigion, UK, <sup>2</sup>CIRES, University of Colorado, Boulder, Boulder, CO, USA, <sup>3</sup>NOAA Space Weather Prediction Center, Boulder, CO, USA, <sup>4</sup>Dipartimento di Fisica e Astronomia “Ettore Majorana”, Università di Catania, I 95123 Catania, Italy, <sup>5</sup>Department of Astronomy, Eötvös Loránd University, Budapest, Hungary

**Key Points:**

- Deep neural networks are used for time series classification and prediction
- The adapted machine learning ensemble model is tailored for the Space Weather community, but applicable to many other communities
- The adopted recalibration technique seems reliable for robust classification

**Correspondence to:**

K. Alielden,  
khd2@aber.ac.uk

**Citation:**

Alielden, K., Camporeale, E., Korsós, M. B., & Taroyan, Y. (2023). Prediction interval of interface regions: Machine learning nowcasting approach. *Space Weather*, 21, e2022SW003326. <https://doi.org/10.1029/2022SW003326>

Received 19 OCT 2022  
Accepted 28 FEB 2023

**Author Contributions:**

**Conceptualization:** Khaled Alielden  
**Formal analysis:** Khaled Alielden  
**Funding acquisition:** Marianna B. Korsós  
**Investigation:** Khaled Alielden  
**Methodology:** Khaled Alielden, Enrico Camporeale  
**Resources:** Khaled Alielden  
**Software:** Khaled Alielden  
**Supervision:** Enrico Camporeale, Youra Taroyan  
**Validation:** Khaled Alielden  
**Visualization:** Khaled Alielden  
**Writing – original draft:** Khaled Alielden  
**Writing – review & editing:** Khaled Alielden, Enrico Camporeale, Marianna B. Korsós, Youra Taroyan

© 2023. The Authors.

This is an open access article under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

**Abstract** Stream interaction region (SIR) is one of the space weather phenomena that accelerates the upstream particles of the interface region in interplanetary space and causes geomagnetic storms. SIRs are large-scale structures that vary temporally and spatially, both in latitudinal and radial directions. Predicting the arrival times of interface regions (IRs) is crucial to protect our navigation and communication systems. In this work, a 1D ensemble system comprised of a Long-short-term memory (LSTM) model and a Convolution Neural Network (CNN) model—LCNN is introduced to classify the observed IR time series and give the prediction interval nowcast of its transit time to the observer. The outcomes of the two models are combined in a way to boost the accuracy of the predictor and prevent error propagation between them. The implemented technique is time series classification on datasets from STEREO A and B spacecrafts. The LCNN prediction system of IRs provides advanced Notice Time (NT) interval between [20, 160] minutes with sensitivity around 93% and geometric mean score  $g_{mean}$  of 91.7%, and the skills decrease with increasing the prediction time. The LCNN demonstrates an enhancement in the prediction with respect to using only either the CNN or LSTM models. The predicted probabilities are recalibrated so that the predicted frequency of IRs becomes on average consistent with the observed frequency. Application of the method is useful to provide a classification of IRs by inputting a time series and estimating the likelihood of occurrence of an IR and its arrival time on the observer.

**Plain Language Summary** Stream interaction regions (SIRs) accelerate the upstream particles of the interaction/interface regions (IRs) in interplanetary space and cause geomagnetic storms. That raises the hazards on satellites' components and Earth's communication and navigation systems. Predicting the arrival time of IRs using machine learning regression techniques is challenging, particularly for such a complex problem of multi-dimensions time series of different time scales. In this work, a 1D ensemble system comprised of a Long-short-term memory (LSTM) model and a Convolution Neural Network (CNN) model—LCNN is introduced to classify the observed IR time series and give the prediction interval nowcast of its transit time to the observer. We used a nowcasting data set from STEREO-A and B spacecrafts. The LCNN provides advance imminent IR at least 20 min and not longer than 160 min with a sensitivity of around 93% and the latter decreases with increasing the prediction time. The predicted probabilities are recalibrated so that the predicted frequency of IRs becomes on average consistent with the observed frequency. Application of the method is useful to provide a classification of IRs by inputting a time series and estimating the likelihood of occurrence of an IR and its arrival time on the observer.

### 1. Introduction

Stream interaction regions (SIRs) form in interplanetary space where the fast solar wind stream overtakes the preceding slow solar wind stream (Allen et al., 2021; Pizzo, 1980). The fast wind piles up the upstream density and forms a compressed plasma along the leading edge of the stream interface region (IR), which is typically identified as the peak in total pressure between the two solar wind populations, and is followed by a rarefaction region in the fast solar wind (Richardson, 2018). The hydrodynamic models of solar wind describe that as the fast streams flow away from the Sun, the solar wind velocity transits steeply from slow to fast and the pile-up density of the IR upstream grows (Alielden & Taroyan, 2022; Hundhausen, 1973). The fast solar wind is from coronal holes and meets the slow wind at a far distance from the Sun. If the fast wind travels significantly faster than the slow wind, a shock wave forms in the interface region which in turn leads to a discontinuity of the total pressure and an increase of the plasma speed upstream of the interface region. Sometimes two pairs of shock waves

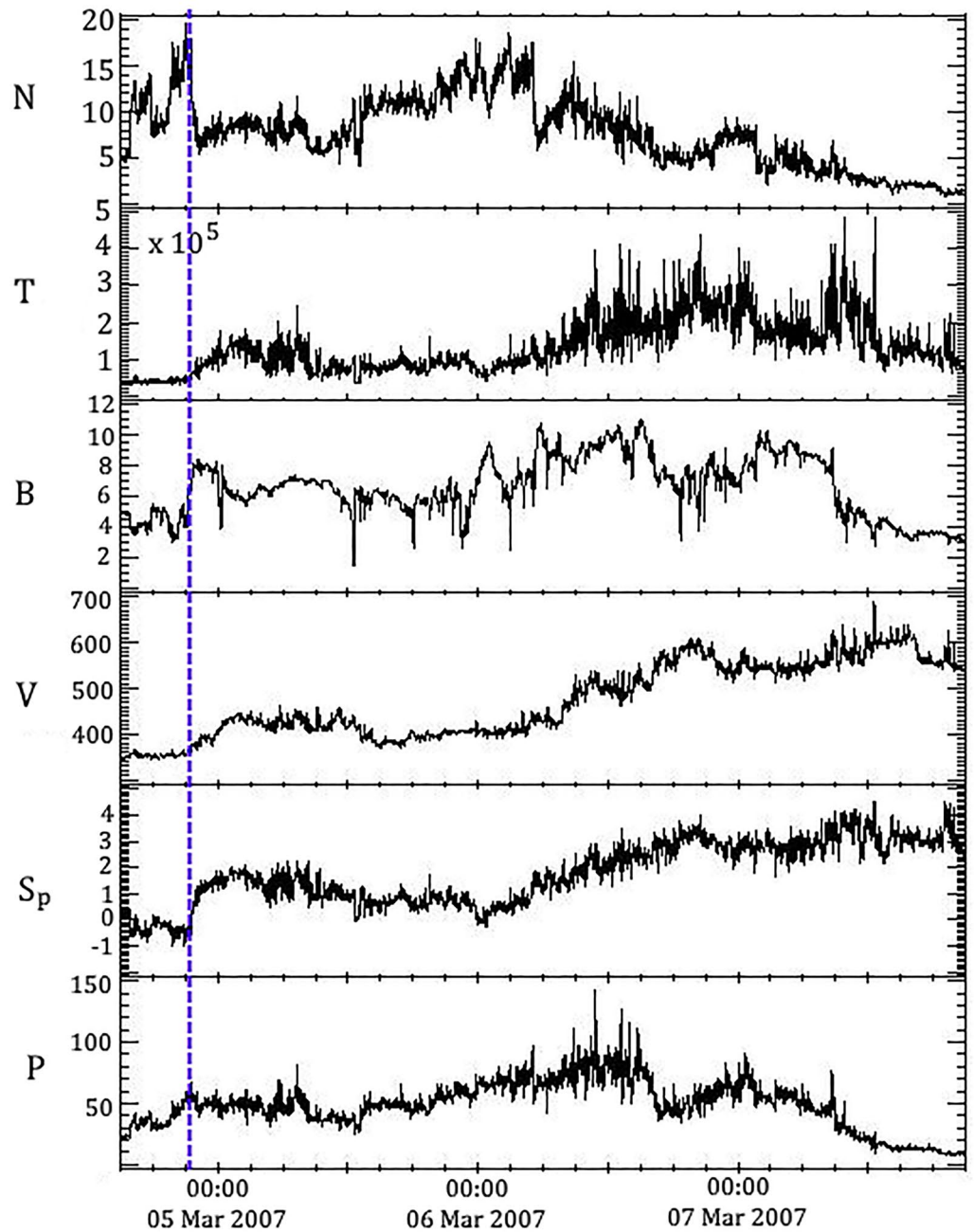
are formed: forward and reverse. The forward shock can efficiently accelerate the particles in interplanetary space and often forms beyond 1 AU. The study of L. K. Jian et al. (2008) on different slices of the solar activity cycle between 1992 and 2005 shows that about 91% of SIRs are associated with shocks, with 47% of such shocks being forward–reverse shock pairs. Since the Sun's equatorial rotation has a period of  $\sim 27$  days as observed from the Earth, the IR region sometimes sweeps in interplanetary space with the same rotation frequency as the Sun. The SIR that co-rotates or completes at least one solar rotation is called a co-rotating interaction region (CIR).

Variations of solar wind parameters triggered by the SIRs have been studied in several works (L. Jian et al., 2006; Kilpua et al., 2012; Grandin et al., 2019; Allen et al., 2021). Observations show that, during the passage of SIRs, the solar wind density and pressure increase. They are usually accompanied by enhancements in temperature, magnetic field strength and speed which often exceeds  $500 \text{ km s}^{-1}$  and sometimes peaks at above  $800 \text{ km s}^{-1}$  (Kavanagh et al., 2012). These variations could last for several days before the plasma returns back to the slow mode state (Tsurutani et al., 2006). L. K. Jian et al. (2019) describe the features of SIRs on solar wind data: (a) the peak of the total pressure  $P$  is usually accompanied by a drop of the density  $N$  and an increase in the speed  $V$ . (b) The compression region causes an increase of the total magnetic field  $B$  and a jump of the entropy  $S_p$ . Figure 1 presents an example of SIR started on 04 March 2007 and lasted for more than 3 days. The blue vertical line refers to the passage of the stream interface where the drop density is coincidental with the pressure peak and the jump in entropy. Since the jump appears only in the  $S_p$  variable, there is no certainty of existing shock wave (L. Jian et al., 2006).

SIRs can drive compression and shocks that accelerate energetic particles in interplanetary space and trigger geomagnetic storms and disturb the ionosphere of the Earth (Chen et al., 2014; Tsurutani et al., 1997; Yu et al., 2021). The intensity of geomagnetic storms on the Earth is often measured by the disturbance storm time (Dst) index which estimates the H-component perturbation by the equatorial magnetometers on the Earth (Mayaud, 1980). Chi et al. (2018) studied the geoeffectiveness of SIRs from 1995 to 2016 and found 52% of the SIRs can cause geomagnetic storms with  $\text{Dst}_{\min} \leq -30 \text{ nT}$ , and only 3% of them cause intense geomagnetic storms with  $\text{Dst}_{\min} \leq -100 \text{ nT}$ . That indicates the SIR could convey a strong magnetic field that disturbs the Earth's magnetic field and can potentially cause malfunctions in the communication systems, electric power transmission systems, and GPS-based navigation systems (Lucas et al., 2018; Thomson et al., 2005). Moreover, SIRs increase the high energetic particles in the space environment around the Earth which in turn raises the risk of ionizing the insulating layers of satellite's components and causes single event effects (Alielden, 2020; Camporeale et al., 2018; Koons & Fennell, 2006). In many cases, software and duplicate circuits are used to correct these effects, but such high-speed solar wind (HSSW) causes a challenging environment with many malfunctions in a short period of time (Horne et al., 2013). The influence of geomagnetic storms sometimes extends to the ground-based power systems on the Earth and induces geomagnetic currents (GICs) that disturb the high-frequency navigation systems (Hapgood, 2011; Marqué et al., 2018). The prediction of SIRs also are studied by Owens and Forsyth (2013) using the 27 days persistence model, Miyake et al. (2005) using the Nozomi spacecraft in-situ observations around L5 point, and Chi et al. (2022) using the STEREO-B and Wind in-situ observations.

Predicting the arrival time of SIRs on the Earth is crucial to avoid hazards and protect the Earth's daily systems. Davis et al. (2012) tried to predict the arrival time of HSSW by superposing epoch observations of the Heliospheric Imager (HI) instrument aboard the ahead NASA Solar Terrestrial Relations Observatory STEREO spacecraft, that is, STEREO A/HI, and matching them with the solar wind data from ACE spacecraft near the Earth. The HI instrument was for observing the enhanced plasma regions in the Sun's atmosphere that are associated with CIRs. Prediction based on Sun observation is restricted by satellite epoch data gap, projection effect, limited field of view, and the location of the satellite with respect to the Earth as well as the position of the event on the Sun. Previous studies found that the properties of SIRs/CIRs have a strong latitudinal dependence (L. K. Jian et al., 2019; Thomas et al., 2018). Allen et al. (2020) investigated the predictive capabilities of in-situ SIR/CIR observations at various heliospheric longitudinal and latitudinal separations. The study suggested that the primary limitation of forecasting the arrival time of SIR is related to the difficulty in predicting the rotational speed of a SIR/CIR, likely due to the evolution of coronal holes and their boundaries and the variations of solar wind speeds.

Predicting the arrival time of SIRs using regular regression techniques may not be a straightforward way as SIRs move in a such highly dynamic environment and change their properties during the travel. Additionally, their time



**Figure 1.** Solar wind variables:  $N(\text{cm}^{-3})$ ,  $T(\text{K})$ ,  $B$  (nT),  $V(\text{km s}^{-2})$ ,  $S_p$ ,  $P(\text{pPa})$ , triggered by passage of a SIR on 4 March 2007 at 15:50 UT (vertical blue line) as recorded by STEREO A. The SIR ends on 7 March 2007 at 20:00 UT.

series are long (predominantly a few days) and often times they are superimposed with coronal mass ejection (CME) or other events. In this work, we focus on predicting the arrival time of the interface region IR portion instead of the whole SIR due to the accompanying hazards of IRs such as shock waves, energetic particles, and significant increase of plasma pressure. The fast solar wind, that typically originates from a coronal hole, impacts the ahead slow wind by pushing it into interplanetary space which in turn leads to density pile-up and pressure increase upstream. The raised pressure gradient, which is maximum at the IR, enhances the HSSW expansion in the interplanetary space. Based on that, we create a nowcasting data set, collected from STEREO spacecrafts, that comprises IR time series intervals labeled before real-time and use a machine learning approach to recognize the features of upstream plasma and provide advanced notice of arriving IRs. The approach is based on time series classification technique and is proposed to be a prediction interval nowcast type. That means the predictor uses

a time range for arriving IR rather than a deterministic time. To increase the certainty of the predicted probabilities, an ensemble prediction system (predictor) is built to combine the outcomes from convolution and recurrent neural network models. The data description and the ensemble prediction system are introduced in Sections 2.1 and 3, respectively. The results and predictor validity are discussed in Section 5. The recalibration method of the predicted probabilities and more discussion are, respectively, in Section 5.2 and 6.

## 2. Data Preparation

### 2.1. Data Description

We use time-series data of solar wind as observed by the instruments of In-situ Measurements of Particles and CME Transients (IMPACT) and Plasma and Suprathermal Ion Composition investigation (PLASTIC) aboard the twin spacecraft: STEREO Ahead (STA) and STEREO Behind (STB) of the STEREO mission (Galvin et al., 2008; Luhmann et al., 2008). Both STA and STB orbit the Sun at a heliocentric distance of  $\sim 1$  AU. The angular separation between STA and STB increases gradually with a drift rate of  $20^\circ/\text{year}$  for STA (leading spacecraft) and  $-28^\circ/\text{year}$  for STB (lagging spacecraft) with respect to the Earth (Kaiser et al., 2008). The two spacecrafts take measurements of the solar wind plasma in interplanetary space at two different points simultaneously. We collected seven solar wind features used by L. Jian et al. (2006) to recognize the IRs such as proton density  $N$  ( $\text{m}^{-3}$ ), solar wind speed  $V$  ( $\text{m s}^{-1}$ ), total magnetic field strength  $B$  (T), temperature  $T_p$  (K), plasma beta ( $\beta$ ), entropy ( $S_p$ ), and pressure  $p$  (pPa), where the latter is the sum of thermal pressure and magnetic pressure of the plasma. The data set spans the time range from 01 January 2007 to 31 December 2016 with a one-minute resolution. Thus the data set includes the solar wind variation during the ascending and descending phases of the solar cycle 24. That is proposed to train the model, later on, for classifying the IRs from complex events, where some of the SIRs are interfaced with interplanetary coronal mass ejections (ICME) or other events. Because of the losing connection to the STB in 2014, the data for the years 2015 and 2016 are collected from the STA only.

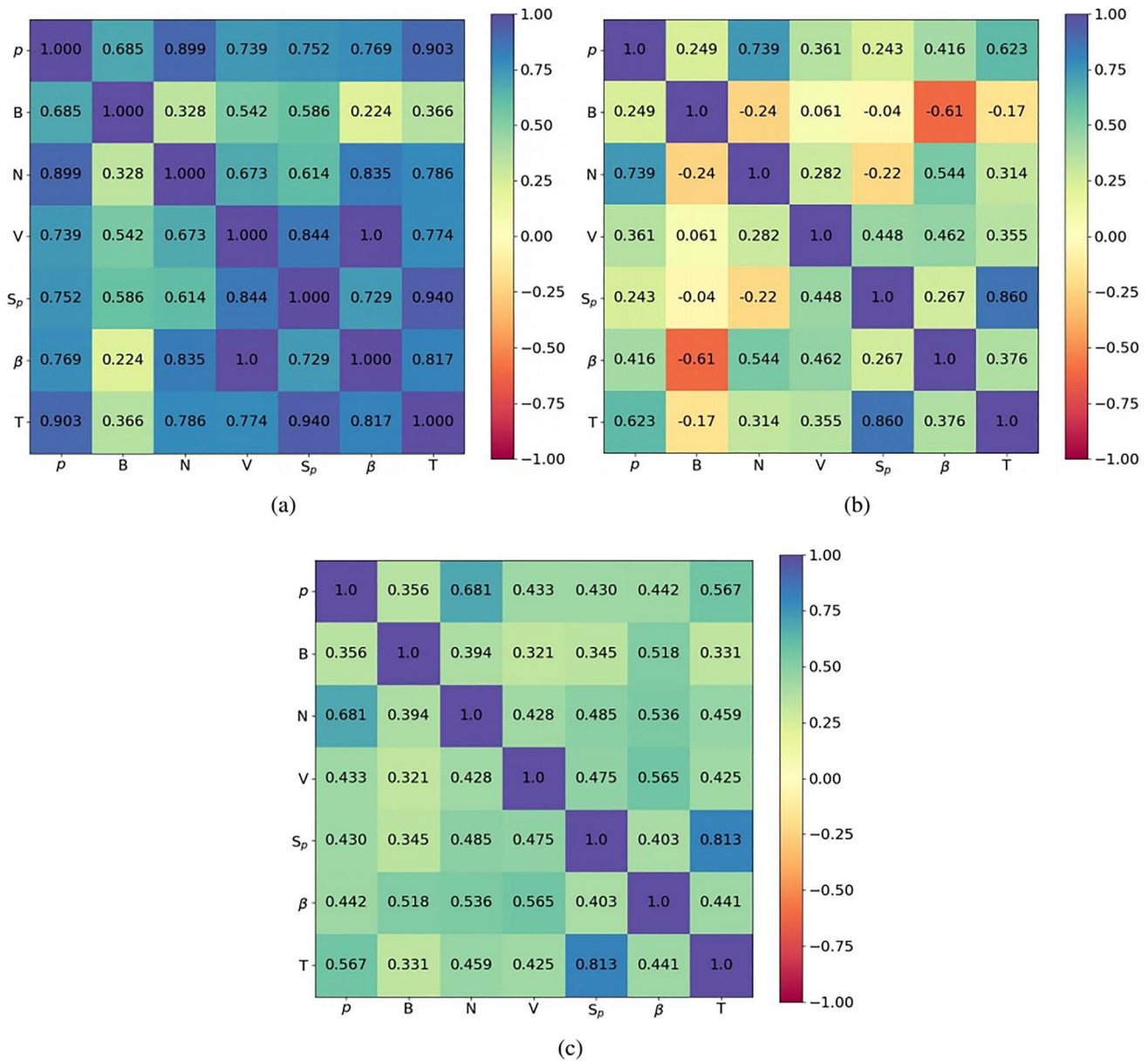
### 2.2. Feature Selection

The aim is to train a model for classifying the time series of IRs using a supervised machine learning technique. We started with general features of SIR intervals to extract the main features for the IR portions. We used SIR catalog that is published by L. Jian et al. (2006) and L. K. Jian et al. (2013) to label the data set, where SIR intervals are labeled as positive. The criteria for identifying SIRs in the catalog are described by L. Jian et al. (2006). Accordingly, we have collected 561 positive intervals altogether from STA and STB in the time range from 2007 to 2016, and each interval has a dimension of  $l_s \times \mathcal{N}_f$  where  $l_s$  is the number of timestamps in the interval and  $\mathcal{N}_f$  is the number of features. Selecting the most informative inputs used in a machine learning algorithm is done by understanding the importance of the features and the correlation between them. We want to make sure that the number of features is not unnecessarily too large to prevent overfitting and for optimizing computational efficiency. We create a function that calculates the vector  ${}_{ij}\mathbf{C}_N = [\text{Corr}_g(i, j)]_g^{N=561}$  which represents the Pearson's correlations between a feature  $i$  and a feature  $j$  for all  $N$  SIR intervals. Repeating this function for all  $\mathcal{N}_f$  features, we get a tensor  $\mathbf{F}_{\mathcal{N}_f \mathcal{N}_f N}$  that for all the correlations between the features

$$\mathbf{F}_{\mathcal{N}_f \mathcal{N}_f N} = \begin{bmatrix} {}_{i1}\mathbf{C}_N & {}_{(i+1)1}\mathbf{C}_N & \dots & {}_{\mathcal{N}_f 1}\mathbf{C}_N \\ \vdots & \vdots & \ddots & \vdots \\ {}_{i\mathcal{N}_f}\mathbf{C}_N & {}_{(i+1)\mathcal{N}_f}\mathbf{C}_N & \dots & {}_{\mathcal{N}_f \mathcal{N}_f}\mathbf{C}_N \end{bmatrix},$$

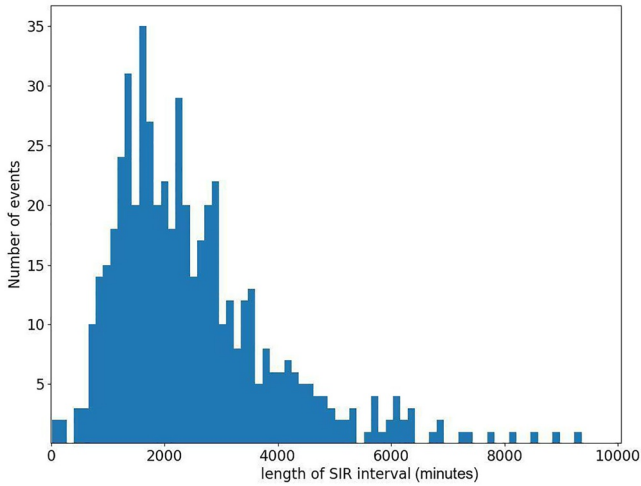
$$\mathbf{M}_{\mathcal{N}_f \mathcal{N}_f} = \begin{bmatrix} Q_2({}_{i1}\mathbf{C}_N) & \dots & Q_2({}_{\mathcal{N}_f 1}\mathbf{C}_N) \\ \vdots & \ddots & \vdots \\ Q_2({}_{i\mathcal{N}_f}\mathbf{C}_N) & \dots & Q_2({}_{\mathcal{N}_f \mathcal{N}_f}\mathbf{C}_N) \end{bmatrix} = [Q_2({}_{ij}\mathbf{C}_N)]_g^N,$$

The value of  $Q_2$  is the median/second quartile of the correlations between features  $i$  and  $j$  of vector  ${}_{ij}\mathbf{C}_N$  of  $N$  dimension. Similarly, the 90th percentile of the vectors is calculated as  $\mathbf{P}_{\mathcal{N}_f \mathcal{N}_f} = [p_{90}({}_{ij}\mathbf{C}_N)]_g^{N=561}$ . Figure 2a demonstrates the obtained correlation quantities of matrix  $\mathbf{P}_{\mathcal{N}_f \mathcal{N}_f}$ . It reflects that 90% of the quantities in the distribution are equal to or lower than the appeared value. The correlation distributions of pressure  $p$  with the



**Figure 2.** Correlation maps between the solar wind features of SIRs over the time span 2007–2016 as obtained from  $\mathbf{F}_{N_f, N_f}$ . (a) The 90th percentile matrix  $\mathbf{P}_{N_f, N_f}$  of the SIRs correlation quantities. (b)  $\mathbf{M}_{N_f, N_f}$  matrix for the median values of the correlations. (c)  $\mathbf{A}_{N_f, N_f}$  matrix for the mean of absolute values of the correlations.

other features are dominantly high. There are features that represent high correlation with some features (e.g.,  $N$ - $\beta$ ,  $V$ - $S_p$ ,  $T$ - $p$ ) and low correlation with others (e.g.,  $B$ - $N$ ,  $B$ - $T$ ). Some features have a dominant distribution up to 100% such as between  $V$  and  $\beta$ . Thus, as generally appeared, the dominant distributions of the observed correlations between the features seem high. Figure 2b shows the median values of  $\mathbf{M}_{N_f, N_f}$ . The correlation such as between  $B$  and  $\beta$  is negative because  $B$  is inverse proportional to  $\beta$  (i.e.,  $\beta = p/B$  is a calculated parameter) and it is positive such as between  $N$  and  $p$  because  $N$  is proportional to  $p$ . The median values clarify that not all the features are highly correlated with each other but with some of them (e.g.,  $N$ - $p$ ,  $T$ - $S_p$ ). The point is that the median presents the value at 50% of the observed distribution in the range between  $[-1, 1]$  but, that does not reflect the exact value of the correlations. Our concern is to determine the importance of the feature by its correlation with the other features regardless of its sign. This is useful to eliminate the non-important features and reduce the dimension of the data set. Figure 2c presents the quantities of  $\mathbf{A}_{N_f, N_f} = \left[ \overline{|\mathbf{C}|_N} \right]_g^{N=561}$  which represents the



**Figure 3.** Distribution of the SIR interval lengths as obtained from the STEREO SIR catalog.

means of absolute values of  ${}_i\mathbf{C}_N$ . It demonstrates the correlations, for example, between  $V$  and  $\beta$  and between  $N$  and  $S_p$  that are not clear in Figure 2b. Taking the mean or the median of the absolute values does not change the distribution significantly.

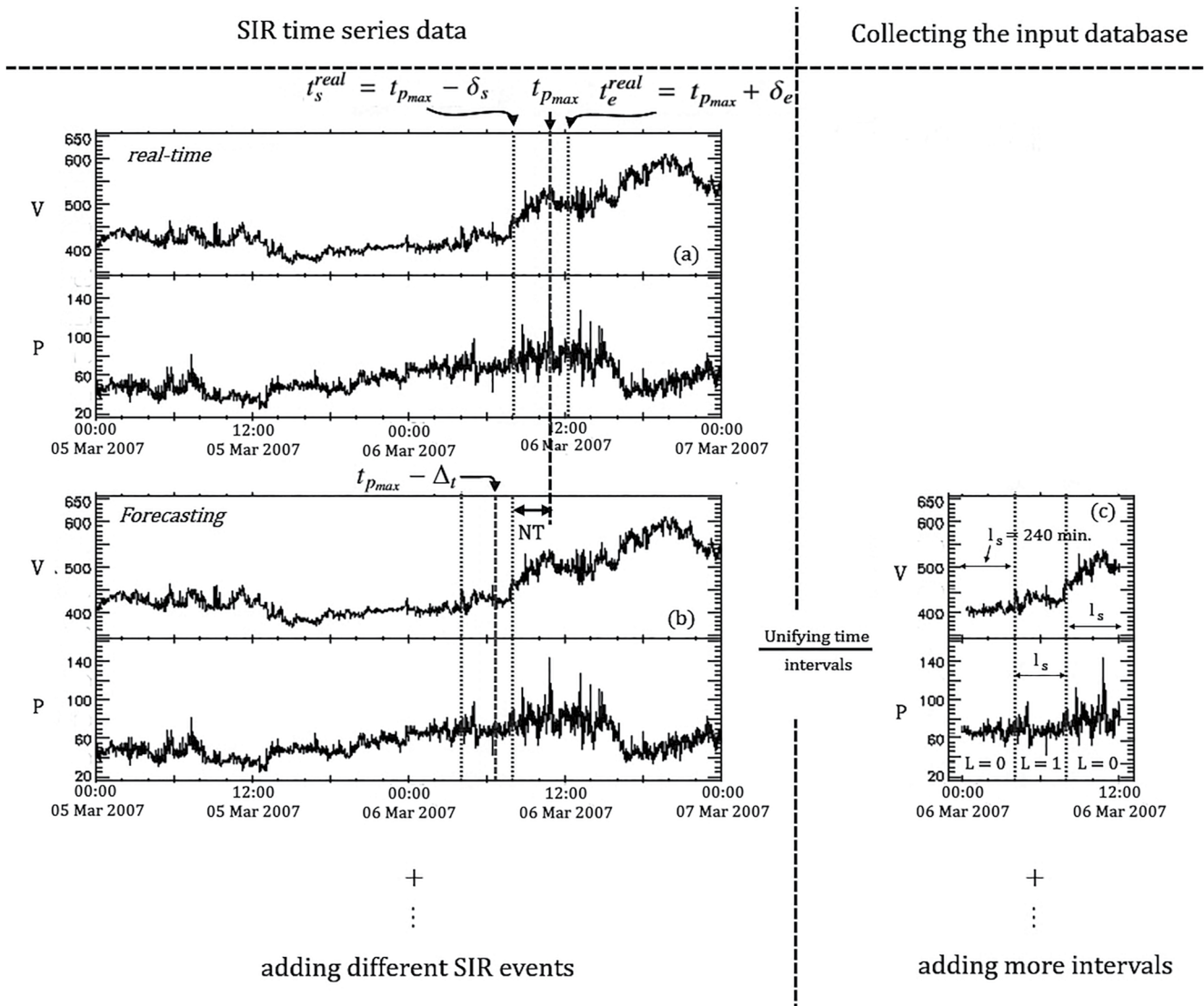
The analysis demonstrates that all the features have at least a good correlation with another feature and that expresses the importance of all features in the computations therefore we use these features (i.e.,  $\mathcal{N}_f = 7$ ) as inputs for the classifier.

The obtained SIR intervals have various time durations. Figure 3 shows a histogram of the duration of SIR intervals as obtained from the catalog. The shortest duration is 19 min and the dominant is larger than 1,000 min. We focus on feeding the model with time series intervals of identical duration. That is because in the time series classification method, the classifier learns the patterns of IRs and there should be an identity length of the input pattern for the classifier to be accurate, particularly for such data that include random noise. Otherwise, training the classifier on different lengths of IRs intervals needs larger training set, which is not available here, and is vague for use in real-world applications. Also, as will be explained in Sections 2.3 and 3, the idea of a unified length is vital for the nowcasting purpose to know the

required length of the input interval for the classifier to recognize IRs. Hence, the two main tasks that we should tackle are: first, how to limit the IRs interval to a fixed length  $l_s$  of timestamps without missing the core information, particularly for the long SIR intervals. Second, find the suitable values of  $l_s$  to get a robust performance of the classifier.

First, we need to find a manner of identifying the start and end times of the intervals. We consider that the most hazardous part of the SIRs is the interface regions IRs (see Section 1). Hence, in each SIR event, we focus on the IR part in the time series data instead of the whole SIR interval. L. Jian et al. (2006) demonstrates that the time of maximum solar wind pressure  $\max(p)/p_{\max}$  mainly refers to the timing of interface region passage. Based on the time of  $p_{\max}$  (herein  $t_{p_{\max}}$ ) in the catalog, we inspected each SIR interval and limited each positive interval referenced to  $t_{p_{\max}}$  to start at  $t_s^{real} = t_{p_{\max}} - \delta_s$  and end at  $t_e^{real} = t_{p_{\max}} + \delta_e$  where  $\delta_s$  and  $\delta_e$  are, respectively, the number of timestamps before and after  $t_{p_{\max}}$ . We stress that every single SIR has only one positive interval in the data set. The time limits for each interval satisfy the boundary condition that is  $l_s = t_e^{real} - t_s^{real}$  where  $l_s$  is the fixed length of the intervals. Figure 4a shows an example of determining  $t_s^{real}$  and  $t_e^{real}$  of a positive interval in the real-time data set. The superscript of *real* refers to the real-time data. That means we relabeled the data set to be positive for the limited IR intervals and negative otherwise. If  $\delta_s$  and  $\delta_e$  are set to constants, the bias of the classifier model will be high. For instance, if  $\delta_s = l_s/2$  and  $\delta_e = l_s/2$  and  $t_{p_{\max}}$  is always in the middle of the limited IR intervals, the signature of shock waves, form in the interaction region, will be always located in the middle of the intervals which resulting in a flaw in the computer memory during training of the classifier. To avoid that, the values of  $\delta_s$  and  $\delta_e$  are taken randomly at each time of calculating  $t_s^{real}$  and  $t_e^{real}$  for different positive intervals. Meanwhile, we apply the condition of  $\delta_s(\delta_e)$  being in the range of [160 (80), 20 (220)].

Second, we utilized the classification model that will be described in Section 3 to find out the optimal  $l_s$  that could be unified for all input time intervals and get a robust classification. The timestamps in the data set (DS) are divided into time series intervals of a fixed length  $l_s$ . Denote a time series interval  $I$  of length  $l_s$  as  $I = [t_i, t_{i+1}, \dots, t_{i+l_s}]$  where  $t_i$  is the value at time stamp  $i \in \mathbf{R}$ . The DS then contains  $N$  time series intervals that are labeled as  $\text{DS} = \{(I_j, L_j)\}_{j=1}^N$  where  $I_j$  is the  $j$ th interval and its label is  $L_j$  which is a single value of positive (1) or negative (0). That means the classifier will tackle each input interval as a positive or negative signal. The intervals in the data set are normalized individually using the z-score of formula  $(x - \mu)/d$  where  $\mu$  and  $d$  are the mean and standard deviation, respectively. The intervals are distributed randomly in the input DS, of  $\dim(\text{DS}) = N \times l_s \times \mathcal{N}_v$ , for unbiased training of the classifier. The aim is to train the model for classifying the positive intervals. We run the classifier several times and each time the value of  $l_s$  changes. After validating and testing the classifier with different values of  $l_s$  in the range from 90 to 400 min with an increment of 20 min, we found experimentally that



**Figure 4.** Procedure of labeling an IR interval around  $t_{p_{max}}$  on 06 March 2007 in the data set (DS). The time series of  $V$  and  $P$  are presented as a simple example but, each interval in the DS comprises seven features. The real time of  $p_{max}$  is  $t_{p_{max}}$ . (a) Real-time IR interval starts at  $t_s^{real}$  and ends at  $t_e^{real}$  (two vertical dashed lines) and has a length of  $l_s = 240$  time stamps. (b) In the nowcasting data set NDS, the  $t_{p_{max}}$  of the positive IR interval precedes the real-time one by  $\Delta_t = 4$  hr. The advanced notice time NT is the time difference between the end of predicting positive interval and  $t_{p_{max}}$  in real-time. (c) The binary labeled intervals in the NDS with a unified  $l_s$ .

- when  $l_s$  is shorter than 240 min, we are exposed to losing information about the IR attributes which in turn leads to declining both performance and certainty,
- when  $l_s$  is longer, we increase the computational expenses and the accuracy of the model does not improve though because the big amount of noise that is included in the data of long time series intervals sometimes misleads the model.

Thus, the suitable unified length of  $l_s$  is selected to equal 240 min. That means the positive and negative intervals have  $\dim(I) = l_s \times \mathcal{N}_v = 240 \times 7$  in the data set and that will be used for the nowcasting of IRs.

### 2.3. Nowcasting Data Set

We intend to use the classifier to recognize the IR and approach their transit time to the observer in advance. The classification process is based on training the model for classifying the IR intervals. While approaching the arrival time of IRs is based on using a nowcasting data set (NDS) as a training set, where  $t_{p_{max}}$  in the real-time (see sec. 2.2)

is shifted back by  $\Delta_t$  and so that each positive interval in the NDS starts at  $t_s^{fore} = (t_{p_{max}} - \Delta_t) - \delta_s$  and ends at  $t_e^{fore} = (t_{p_{max}} - \Delta_t) + \delta_e$ . For instance, if  $t_{p_{max}} = 14:00$  in real-time and  $\delta_s = 150$  min and  $\delta_e = 90$  min, the  $t_{p_{max}}$  in the NDS for  $\Delta_t = 4$  hr will be at 10:00 and the positive interval will start at  $t_s^{fore} = 07:30$  and end at  $t_e^{fore} = 11:30$ .

Since the time difference between  $t_{p_{max}}$  in the real-time and  $t_e^{fore}$  in the NDS is  $14:00 - 11:30 = 150$  min, we consider the model function, in this case, would provide 150 min advanced notice of an IR before it reaches the observer. Generally, it means that the advanced notice time NT of reaching IRs is basically  $NT = \Delta - \delta_e > 0$  where  $\delta_e$  is the end point of the nowcasting time-series interval that is needed to predict, and as long as  $\Delta_t$  is larger than  $\delta_e$  the model function will keep providing nowcasting. The model's outcome NT interval is  $[\Delta_t - \max(\delta_e), \Delta_t - \min(\delta_e)]$  that is determined based on the  $\delta_e$  range in the training set. That indicates the imminent reaching IR (in hours) is at least  $\Delta_t - \max(\delta_e)$  and not longer than  $\Delta_t - \min(\delta_e)$ . Thus, when  $\Delta_t$  is large and  $\delta_e$  is small the nowcasting/advanced NT increases and when  $\delta_e$  range is narrow the prediction becomes more precise and needs a large data set for training, however. Figure 4b demonstrates the way of determining the time boundaries of a positive interval in the NDS for the example of real-time IR interval in Figure 4a.

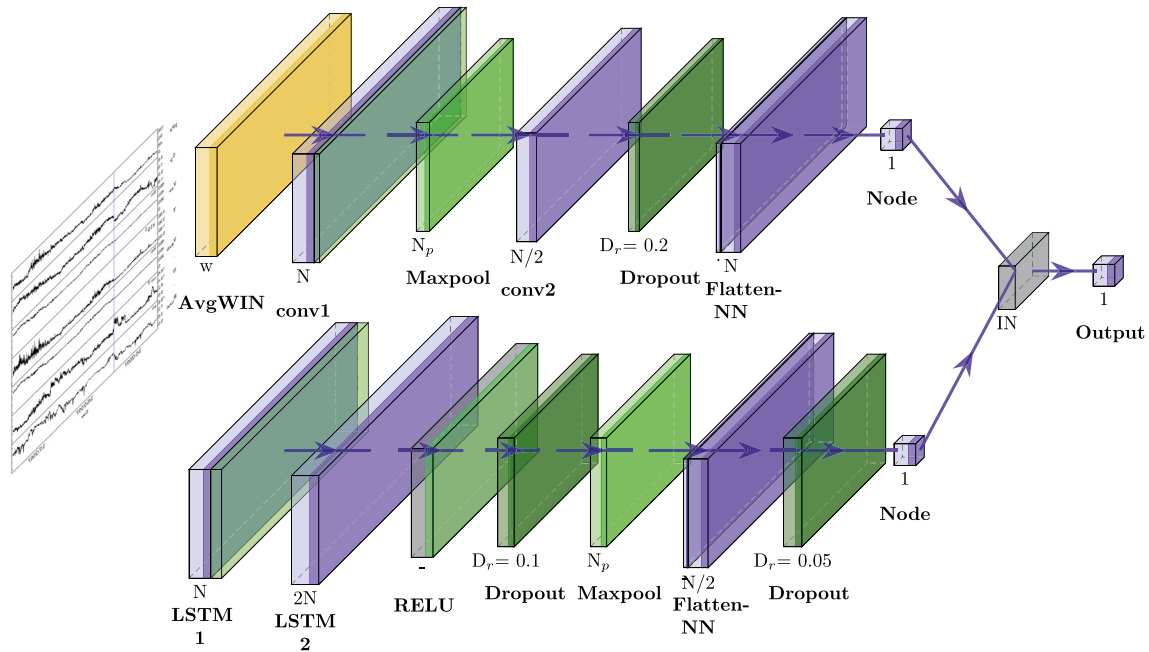
The restriction of  $\Delta_t \geq l_s$  should be applied to avoid overlapping between the nowcasting interval and the real-time interval. Also,  $\Delta_t$  is preferred to be not very large because when the nowcasting interval is very far from the real-time interval, the signature of the IRs vanishes and their patterns become not clear for the classifier. An additional thing that influences the performance of the predictor is that the number of positive intervals in the NDS decreases with increasing  $\Delta_t$  due to the missing data. For instance, the number of positive intervals reduces from 561 to 558 for  $\Delta_t = 4$  hr and it further reduces if  $\Delta_t$  is larger. Therefore, improving the performance of the classifier/predictor for large  $\Delta_t$  requires a larger training set to enhance the sensitivity and specificity of the system. The input NDS for the classifier/predictor is structured in the same way as DS (see Section 2.2) and it covers the same spanning time. We checked if the optimal  $l_s$  is different for the nowcasting data by carrying out the same test as explained above but, there was no significant change in the conclusion mentioned above. Hence, the positive and negative intervals kept unified to  $l_s = 240$  min Figure 4c shows a slot of three consecutive intervals (one positive between two negatives) as included in the NDS.

### 3. Ensemble Model

The sequence of real-valued data points of SIR with timestamps is analyzed as time series (TS) with machine learning classification approaches. Time series classification (TSC) is a method to predict the class label L of a given input time series. The complexity of the TSC method, using neural networks, increases when the data pattern is represented in high dimensions. The additional key factors that contribute to the difficulty of the TSC method for the real-world time series data are: first, representation of different features of the time series requires different time scales (Drewry et al., 2013). Second, discriminative patterns of the time series are often distorted by high-frequency perturbations and random noises. To make the trend of the time series clear, automatic smoothing and de-noising procedures are needed. The existing TSC techniques rarely adapt the time series features to the right scales. Moreover, one of the notorious problems of the TSC using machine learning algorithms is the vanishing of the gradient during the training process of the classifier. That is because sometimes the change of the values with time is not obvious to the algorithm (Bengio et al., 1994; Glorot & Bengio, 2010). Our work is not only for binary classification of IRs but for exceeding the sensitivity and probability of the TS classifier. In this section, we introduce an ensemble classification system called LCNN that is structured to increase the ability of the classifier to learn different features of the TS without losing the major information.

The LCNN abbreviation stands for Long-short term memory - Convolution Neural Network. The LCNN system basically combines the convolutional and long-short term memory (LSTM) layers/models in parallel where each model works on a different scale of the input time series and consequently learns different features. Figure 5 shows the structure of the LCNN system. The top model is the convolution neural network model (model 1) and the bottom model is the LSTM neural network model (model 2). The two models in the LCNN are combined through the integrated layer (IN) to exceed the probability of predicting label L of the input time series. Model 1 works on low-frequency (small-scale) time series data while model 2 works on the original time series which carries information about the combination of low and high frequencies (Cui et al., 2016). Table 1 shows the parameters of model 1 and model 2 that are combined in the LCNN system for classifying the IRs. These parameters are set up after running the models several times with the same input and different parameters until we get a robust classification. Both model 1 and model 2 are employed for the TSC task in one dimension and work independently as explained in the following.





**Figure 5.** Procedures of classifying IR intervals by the LCNN ensemble system. The different colors of the layers refer to different functions. The consecutive layers of model 1 (top) are: AvgWIN → conv1 → Maxpool → conv2 → linear → single Node (output). The consecutive layers of model 2 (bottom) are: LSTM 1 → LSTM 2 → ReLU → Maxpool → linear → single Node (output). LCNN is designed to combine the outputs of model 1 and model 2 through the integrated layer (IN) that returns a single value of probability for binary classification.

### 3.1. Model 1 - Convolution Neural Network

To get low-frequency time series from the original input time series, we created an average-window layer (AvgWIN) that generates the low-frequency time series before passing the data to the convolutional layers. The AvgWIN layer basically generates a smooth time series using an average window of size  $w$  moving with a stride of  $S$ . Suppose the original input time series to AvgWIN layer is  $I$ , the layer returns a new time series as

$$I^w = [I_1^w, I_2^w, \dots, I_{l_s-w+1}^w]$$

$$I_k^w = \frac{x_k + x_{k+1} + \dots + x_{k+w-1}}{w},$$

and  $k = 1, 2, \dots, l_s - w + 1$ . The new time series interval  $I^w$  is shorter than the input interval  $I$  by  $w - 1$ . The interval  $I^w$  is used as input to the first consecutive convolutional layer (conv1) to extract the features of the low-frequency time series. The output from conv1 is normalized and then passes through a max pooling procedure to extract high-level features and reduce the size of the features map which in turn leads to decreasing the number of parameters in model 1 and preventing the vanishing of gradient and over-fitting. The obtained information afterward passes through the second convolutional layer (conv2). A dropout layer is employed with a rate  $D_r$  of 0.2 before the linear/flatten-neural network (NN) layer. Dropout is a technique that ignores randomly selected neurons to avoid overfitting during the training process and enhances the classifier performance. Both conv1 and conv2 apply the Rectified Linear Unit (ReLU) filters that move with a stride of  $S = 1$ . The ReLU function,  $f(x) = \max(0, x)$ , returns the value of input  $x$  if it is positive and returns 0 if  $x$  is negative. The output layer of model 1 comprises a single node with a logistic function  $\sigma(x) = 1 / (1 + e^{-x})$  to train and test the performance of model 1 separated from model 2. The latter function returns the likelihood of being the time series positive. Finding the suitable value of  $w$  is carried out by running model 1 with different values of  $w$  in the range from 30 to 120 with an increment of 20. When  $w$  is large, the computation turns out to be fast but, the amount of information

**Table 1**  
Parameters of the LCNN System That Is Presented in Figure 5

Model 1		Model 2	
Layer	Parameters	Layer	Parameters
AvgWIN	( $w = 60, S = 1$ )	lstm 1	( $N = 256$ )
conv1	( $N = 256, f_z = 8$ )	lstm 1	( $N = 512$ )
maxpool	( $N_p = 60, S = 20$ )	maxpool 1	( $N_p = 60, S = 30$ )
conv2	( $N = 128, f_z = 4$ )	linear	( $N = 128$ )
linear	( $N = 256$ )		

*Note.*  $w$  is the window size of AvgWIN layer and  $S$  refers to the stride of a window in the layer.  $N$  and  $f_z$  are the number of nodes and the filter size in the layer, respectively.  $N_p$  is the window size of taking the maximum.

in  $I_w$  becomes not enough to learn the model properly (lost data). When  $w$  is relatively small, the amount of data exceeds and does not discriminate from the original data. Therefore, the user should be careful in selecting the value of  $w$ , and in our case based on the experiment, the suitable value of  $w$  is around 60.

### 3.2. Model 2 - LSTM Neural Network

Model 2 uses long-short-term memory (LSTM) network to extract the sequential features of the original time series. LSTM layer is a recurrent neural network (RNN) that is modified to avoid the gradient problem in the RNN, particularly for the long-term dependency problems (Bengio et al., 1994). LSTM layer generates the output through the interaction between the past stored computation and the given input data. For more information about the structure of LSTM layers, we refer to the references, for example, Graves (2012); Hochreiter and Schmidhuber (1997). LSTM layers have been employed for the time series predictions (Li et al., 2019; Wang et al., 2021; Zhang et al., 2019). In this work, we applied *tanh* and *sigmoid* functions in the two LSTM layers in model 2. The parameters of model 2 are presented in Table 1. The output of the LSTM 1 layer is normalized before passing the data to LSTM 2 layer. The roles of ReLU function, dropout layer of  $D_r = 0.1$ , and Maxpool layer are the same as explained in model 1. Unlike the convolutional layer, LSTM is sensitive to sequential information, and the max pooling operation in model 2 is expected to not vary this information. That makes model 2 learns different features from model 1 which in turn would improve the predicting capability of the LCNN system. That reflects the purpose of combining two models in the LCNN system, to feed the system more information about the SIR features. The output of the Maxpool layer passes to the linear layer. Like model 1, the output layer comprises a single node with the same logistic function.

The Adaptive Moment (AdaM) optimization algorithm is utilized with a learning rate of  $2 \times 10^{-5}$  to train both model 1 and model 2 (Kingma & Ba, 2014). AdaM is an adaptation of a stochastic gradient descent algorithm that combines the two optimization methods of the Adaptive Gradient (AdaGrad) and the Root Mean Square Propagation (RMSprop). The former uses the average of sparse gradients to approach the best direction of minima while the latter adapts the learning rate per parameter based on how quickly the weight changes and takes the average of recent gradients. Adam demonstrates a decent performance with non-stationary and noise problems (Kingma & Ba, 2014). During the training process, the performance of the classifier is evaluated by the cross-entropy/log loss function  $F_p$

$$F_p = -\frac{1}{N} \sum_{j=1}^N L_j \log P(L_j) + (1 - L_j) \log(1 - P(L_j))$$

where  $L_j$  is the truth label of the input interval and  $P(L_j)$  is the predicted probability of the interval being positive for all  $N$  intervals in the data set. The output of the LCNN system is given from the combination of model 1 ( $m_1$ ) and model 2 ( $m_2$ ) through the Integrated layer (IN) as

$$\text{Int}(x) = \sum_{n=1}^S \sigma^*(P_n(x)) \quad (1)$$

where  $S$  is the number of combined models in the LCNN system, which in our case equals 2. The function  $\sigma^*(y) = -\log_e(y^{-1} - 1)$  is used to get the exact output from each model instead of their predicted probabilities. The variable  $P_n(x)$  is the predicted probability of  $n$ th model for an input  $x$ . We will discuss later the effect of the IN layer on the accuracy of the LCNN system. The output layer of the LCNN (far right in Figure 5) predicts the probability of the time series as  $P(x) = \sigma(\text{Int}(x))$ . We have used the Keras and TensorFlow libraries publicly available at [https://keras.io/getting\\_started/faq/](https://keras.io/getting_started/faq/) and <https://www.tensorflow.org/>.

## 4. Training and Validation

The NDS of a specific  $\Delta_i$  is utilized to train the LCNN model. That means the LCNN should return the probability of being the input interval of an IR that will arrive within time NT to the observer. Since  $\Delta_i$  is within a short period, several hours, we consider the model function to be nowcasting of IRs. Nowcasting is basically forecasting in a short term. We found that the sensitivity of the model  $\sim 66\%$  when using the STA data only as a training set and the STB data only as a test set. That is because the SIRs change shape during their travel time in interplanetary

space which in turn causes changes in the SIR patterns (Dorrian et al., 2010). That means the SIR patterns that are observed by STA could be different than the SIR patterns that are observed by STB. This is consistent with the analysis of Allen et al. (2020) who studied the predictive limitations of in-situ SIR observations at various longitudinal and latitudinal separations. They found that the probability of predicting the occurrence of SIRs at another spacecraft decreases with increasing longitudinal separation between the spacecrafts. They mentioned that the primary source of error in predicting the arrival time of a SIR/CIR is uncertainty in the rotational speed of the SIR structure, likely due to the evolution of coronal holes and the variations in solar wind velocities.

To enhance the model flexibility, the NDS is mixed with different positive intervals from STA and STB. Then, the data set is divided into 70% for training, 10% validation, and 20% for testing the model. Since the number of negative intervals in the NDS is quite larger than the number of positive intervals, for instance for  $\Delta_t = 4$  hr the number of negative intervals in NDS is  $\sim 29,177$  and positive intervals are  $\sim 558$ , the number of negative intervals in the training and testing sets exceeds the positive ones by 400 intervals. That is purposed to cover a good variety of simple and complex patterns of positive and negative classes in the data set during training the model. Also, to satisfy an unbiased condition when testing the sensitivity of the model to recognize IRs. When we equalize the positive and negative intervals in the datasets, the model does not perform well because the model trains on a small portion of the data and it falsely classifies the complex patterns in the test set.

## 5. Results

### 5.1. IR Nowcasting Approach

The model performance is evaluated by matching the binary predicted map with the truth map. The predicted map forms based on a threshold (TH) that determine the label \*L of an interval I as

$$*L = \begin{cases} 1, & \text{if } P(I) \geq \text{TH} \\ 0, & \text{otherwise} \end{cases}$$

where  $P(I)$  is the predicted probability. The main quantities of the evaluating matrices are: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The validity of the classifiers is estimated through the following statistical quantities:

- Precision, also known as Positive Predictive Value (PPV), is an indicator of the accuracy of the model. In the context of our study, precision expresses the ability of the classifier to identify the IRs.
- Precision =  $TP/(TP + FP)$ ,
- True Positive Rate (TPR), also known as sensitivity or recall, is a measure of likelihood of a positive result. In our case, it expresses the ability of the classifier to find all the IRs in the database.
- TPR =  $TP/(TP + FN)$ ,
- False Positive Rate (FPR), which is a measure of the likelihood of a false alarm. It measures the returned IRs by the classifier that do not exist.
- FPR =  $FP/(TN + FP)$ ,
- True Negative Rate (TNR), also known as specificity, is a measure of likelihood of a negative test. In our study, this quantifies the ability of the model to find all the intervals without IRs correctly.
- TNR =  $TN/(TN + FP) = 1 - \text{FPR}$
- The  $F_\beta$ -score, also known as  $F_\beta$ -measure, is the weighted harmonic mean of the precision and TPR and quantifies the accuracy of a test. It is given by the expression:

$$F_\beta = (1 + \beta^2) (\text{Precision} \cdot \text{TPR}) / (\beta^2 \cdot \text{Precision} + \text{TPR}).$$

In this work, the importance of both precision and TPR are considered to be balanced on the positive class, that is,  $\beta = 1$ , and therefore the harmonic mean will be measured by the so-called  $F_1$  score value which is obtained via this equation:

$$F_1 = 2 (\text{Precision} \cdot \text{TPR}) / (\text{Precision} + \text{TPR}).$$

We implemented the receive characteristic curve (ROC) strategy to map the TPR and the FPR together as a function of the threshold TH in the range from 0 to 1 (e.g., see Figure 8). A robust classification model is obtained

**Table 2**  
*Statistic Measurements of Three Independent Runs R of Model 1, Model 2, and LCNN System for 240 min Nowcast*

	Model 1			Model 2			LCNN		
	R1	R2	R3	R1	R2	R3	R1	R2	R3
TH*	0.39	0.83	0.54	0.73	0.68	0.67	0.66	0.94	0.61
Measurements %:									
max( $g_{mean}$ )	89.06	88.67	88.09	90.97	90.04	90.21	91.13	91.59	91.79
TPR	92.72	88.18	88.18	93.63	93.63	93.63	92.72	91.81	94.54
FPR	14.44	10.82	11.99	11.60	13.41	13.07	10.43	8.61	10.87
$F_1$ score	40.23	45.32	42.92	45.77	42.29	42.91	48.0	52.16	47.70
$F_2$ score	60.92	63.98	62.02	66.02	63.06	63.57	67.54	70.40	67.88
AUC	62.61	64.89	63.82	64.95	63.46	63.72	65.97	67.99	65.78
Corr. C.	85.92	89.12	88.01	88.66	86.94	87.26	89.73	91.40	89.40
Statistics:									
Mean TPR		89.69	$\pm 2.14$		93.63			93.02	$\pm 1.13$
Mean FPR		12.41			12.69			9.96	
Skills %:									
TSS	78.28	77.35	76.18	82.03	80.21	80.56	82.29	83.19	83.67
HSS	35.03	40.83	38.13	41.23	37.34	38.03	43.73	48.42	43.37

when TPR and TNR are close to 1 and FPR is close to 0. So that, we calculate the geometric mean  $g_{mean}$  (see in sec. Appendix A) at the different values of TH to pick the highest scores of both TPR and TNR together and ensure the balance between them (Kubat & Matwin, 1997). The best TH (TH\*) is considered the one that gives the maximum  $g_{mean}$  score. We run the LCNN system 30 iterations with internal neural network changes to make each integration independent from the others. The performance of the system does not change significantly with iterations. The slight changes are due to the problematic nature of probabilities and the changing of initial weights and input intervals at each iteration.

Table 2 shows the output statistics at TH\* of three iterations  $R_1$ – $R_3$  out of 30 independent iterations for model 1, model 2, and LCNN system. The results describe the score ranges and the capability of the LCNN/predictor to provide advanced notice of arriving an IR with NT at least 20 min and not longer than 160 min, for  $\Delta_t = 4$  hr (see sec. 2.3). The test set in this case contains  $\sim 110$  positive intervals out of 558 in the NDS. The detected TH\* and the corresponding max( $g_{mean}$ ) are presented in row 1 and row 2, respectively. The max( $g_{mean}$ ) of model 1 and model 2 is more or less 90% while combining their outcomes using formula 1 exceeds the certainty to  $>91\%$ . The sensitivity in row 3 also enhances with the LCNN however, a few times it is lower than in model 2. Looking at row 4, the LCNN often decreases FPR which indicates to increase in specificity. That means the quantities of FN and FP for model 1 and model 2 are more likely adjusted in the LCNN and the certainty of the latter would be higher. This explains the reason for the value of TH\* for the LCNN system is different from the TH\* values of the two models. The  $F_1$  score in row 5 reflects how well the positive intervals are classified. As anticipated,  $F_1$  score for the LCNN system is higher than the other two models even in the case of decreased TPR. This confirms that the performance of the classifier/predictor is enhanced with the LCNN system. The confidence of  $F_1$  score of around 49% is acceptable for our imbalanced data. This is mainly because there are too many negative intervals in the input NDS and the classifier returns some of them as false positives. That increases the quantity of FP and decreases the precision. If the false negatives are considered worse than the false positives, which means the extra attention we receive is not harmful like missing the IRs, the  $F_2$  score with  $\beta = 2$  would be a good measure for that case. The  $F_2$  scores in row six indicate also a good performance of the LCNN system to classify the positive intervals.

The area under the ROC curve AUC is insensitive to the changes in class distribution in the test set, that is, changes in the class distribution do not have an influence on calculating the AUC, but it is sensitive to a class imbalance in the test set. When the positive class is the minority class it will have a strong impact on the AUC

**Table 3**  
Statistic Measurements of Three Independent Runs R of Model 1, Model 2, and LCNN System for 16 and 24 hr Nowcasts

	16 hr			24 hr		
	Model 1	Model 2	LCNN	Model 1	Model 2	LCNN
TH*	0.61	0.34	0.16	0.52	0.87	0.35
Measurements %:						
max( $g_{mean}$ )	86.12	83.94	86.37	82.77	85.83	87.89
TPR	88.18	85.45	90.90	83.63	81.81	90.0
FPR	15.88	17.54	17.93	18.07	09.94	14.15
$F_1$ score	36.39	33.33	34.60	32.22	44.66	39.75
AUC	61.09	59.88	60.38	59.44	64.82	62.44
C. Corr	84.32	82.60	82.51	82.01	89.63	86.05
Skills %:						
TSS	72.29	67.91	72.97	65.56	71.87	75.84
HSS	30.81	27.38	28.72	26.12	40.22	34.54

value (Ahmadzadeh et al., 2019; Bloomfield et al., 2012). In other words, ROC is biased toward the larger population when it comes to classification or prediction as discussed by Jeni et al. (2013). Hence, the AUC, in row 7, of values >62% reflects a good performance of the LCNN to discriminate IRs, when considering the TPR values. The AUC could be enhanced by increasing the number of positive intervals in the data set. Row 8 shows the correct classified score Corr. C. (see section Appendix A) that measures the accuracy of classifying the positive and negative classes. The scores of model 2 are usually higher than the scores of model 1, while the FPR scores for model 1 are lower. We stress that model 1 and model 2 do not give the same prediction map because each of them sees the input intervals differently and combining their outcomes gives a better prediction map to the true one. Despite there is no big difference between the sensitivity of the LCNN and model 2, the generic measured quantities reveal the performance enhancement with the LCNN system. The values are close to each other which reflects the stability of the LCNN system for nowcasting.

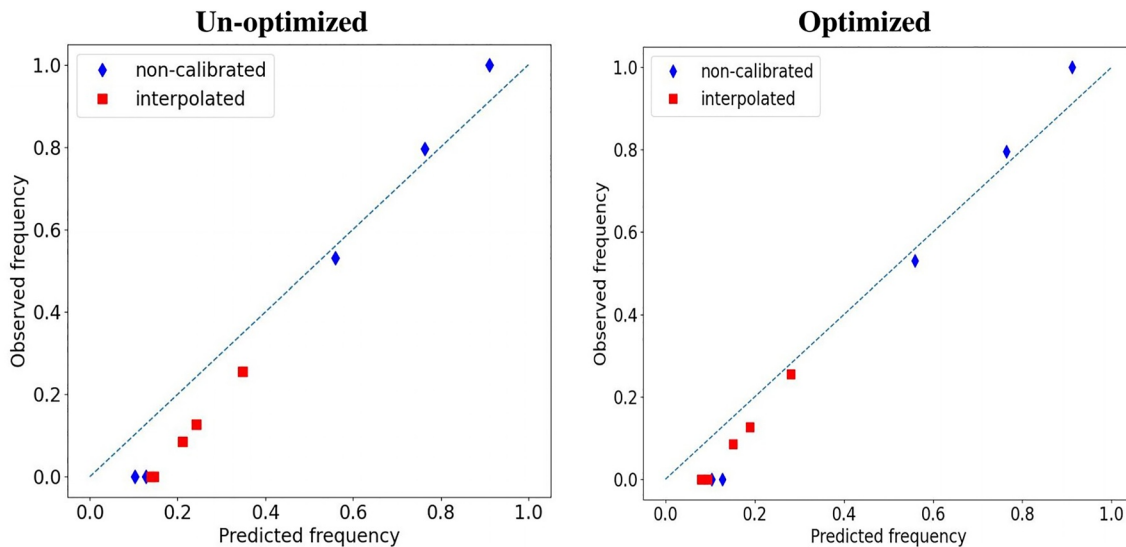
The true skill statistic (TSS) and the Heidke skill score (HSS) are calculated to evaluate the LCNN system as a predictor of IRs. TSS was introduced by Peirce (1884) as Peirce's skill score to evaluate the meteorology predictions. It is a special case of Kappa statistic, which measures the similarity between the predictive maps derived from modeling and the actual maps (Allouche

et al., 2006; Piri Sahragard et al., 2018). The similarity takes a value between  $[-1, 1]$  and it is good when the TSS value is between 0.55 and 0.7 and it is high when the value is between 0.7 and 0.85, and it is excellent when the value is greater than 0.85 - indicating a perfect match between the actual and predicted maps. In the ROC curve, TSS combines TPR and FPR (see sec. Appendix A) and measures the distance to the diagonal (no-skill) line (Krzanowski & Hand, 2009). Despite TSS considering the proportions of positives and negatives in the validation set are equal (Allouche et al., 2006), we calculated the TSS to see how well the IRs are predicted. According to the scale above, the TSS in row 11 demonstrates a high capability particularly for the LCNN system to nowcast the arrival of IRs 240 min in advance. HSS measures the quality of the prediction over a baseline represented by random chance (Florios et al., 2018). The term *skill* refers to the normalization of the predictor score by the total range of possible values over the standard (reference values) (Hyvärinen, 2014). Often, the standard used is the long-term average prediction of the parameters. HSS takes a value in the range from  $-\infty$  to 1 (see section Appendix A) and it has been used in several works such as Camporeale et al. (2020) and Ji et al. (2021). A score of 0 means no skill and the prediction is likely to be expected by chance. A score of 1 indicates a perfect prediction and a score of  $-0.5$  indicates a not good prediction. Here, we used HSS for evaluating the matching between the prediction and observation within the selected time span. Row 12 demonstrates that the skills of LCNN are higher than model 1 and model 2. Overall, the LCNN has a good HSS score of  $83.05 \pm 0.57$  and it seems stable for  $\Delta_t = 4$  hr. We conclude from the experiments' outputs that combining the outcomes from model 1 and model 2 does not drop the performance of the classifier/predictor but, it exceeds the certainty of prediction. For instance, if model 1 and model 2 have different sensitivities of the positive intervals, the sensitivity of the LCNN system will not be lower than one of the models and often with lower FPR.

The limitation of the predictor is representing in answering the question, of how early can the predictor recognize the IRs before their arrival to the observer. This could be described through Table 3 that shows the measured scores of the predictors that are trained on two separated NDSs with  $\Delta_t = 16$  hr (960 min) and  $\Delta_t = 24$  hr. We kept the same features and  $l_s = 240$  for the intervals as described in Section 2.3. Comparing the results of Table 2 with Table 3 demonstrate that the certainty and reliability of the predictor decrease with increasing the nowcasting time  $\Delta_t$ . That appears in the reduced scores of  $\max(g_{mean})$ , TSS, and HSS and also in the raising values of the FPR. The values of  $H^*$  indicate that the predicted probabilities for long  $\Delta_t$  become highly sparse and need to be re-calibrated. That will be discussed in Section 5.2. The performance of LCNN still works well for exceeding the probabilities of prediction.

## 5.2. Recalibration

The observed occurrence frequency of the IRs should be consistent with the predicted frequency. In a binary classification setting, we calculate the predicted occurrence frequency  $\gamma$  by dividing the data set into  $n$  parts and



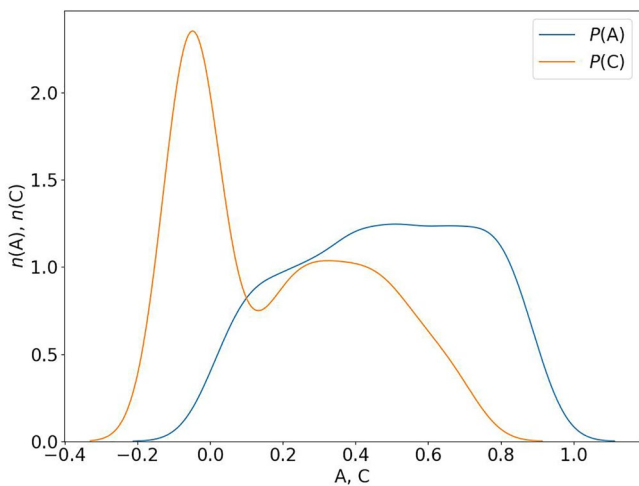
**Figure 6.** Reliability diagram of the observed frequencies of the IRs versus the predicted frequencies by the LCNN. The blue diamonds represent the prediction using the training set, while the red squares represent the recalibrated prediction using the test set. (a) Without the optimized calibration function. (b) With the optimized calibration function.

for each part we calculate the ratio of collected predicted positive PP (predicted probabilities exceeding a threshold) to the total number of instances in the  $n$ th part  $N_{nth}$ , that is,  $\gamma = PP/N_{nth}$ . The classifier is well-calibrated if on average  $\gamma$  is actually observed (the actual frequency is calculated over all those instances in the same way as  $\gamma$ ) (Guo et al., 2017). A reliability diagram is used to visualize the relationship between the predicted and observed frequencies/probabilities (DeGroot & Fienberg, 1983). To construct such a diagram for binary classification, the predicted probabilities are discretized in bins. For each bin, the average predicted frequency (horizontal axis) is plotted against the true fraction of positive cases in that bin (vertical axis). Figure 6 shows the reliability diagrams as obtained from the LCNN. The diagonal straight line represents a perfect calibration. One can clearly see that the predictions are miscalibrated. For recalibration, we linearly interpolate the blue diamonds to derive a map between old and new probabilities (Camporeale et al., 2020). The derived recalibration map might change slightly the outcome predicted probabilities and consequently the  $TH^*$ . For a valid recalibration process, the recalibration map has been derived from the training set, and then it is applied over the test set. The red squares in Figure 6a

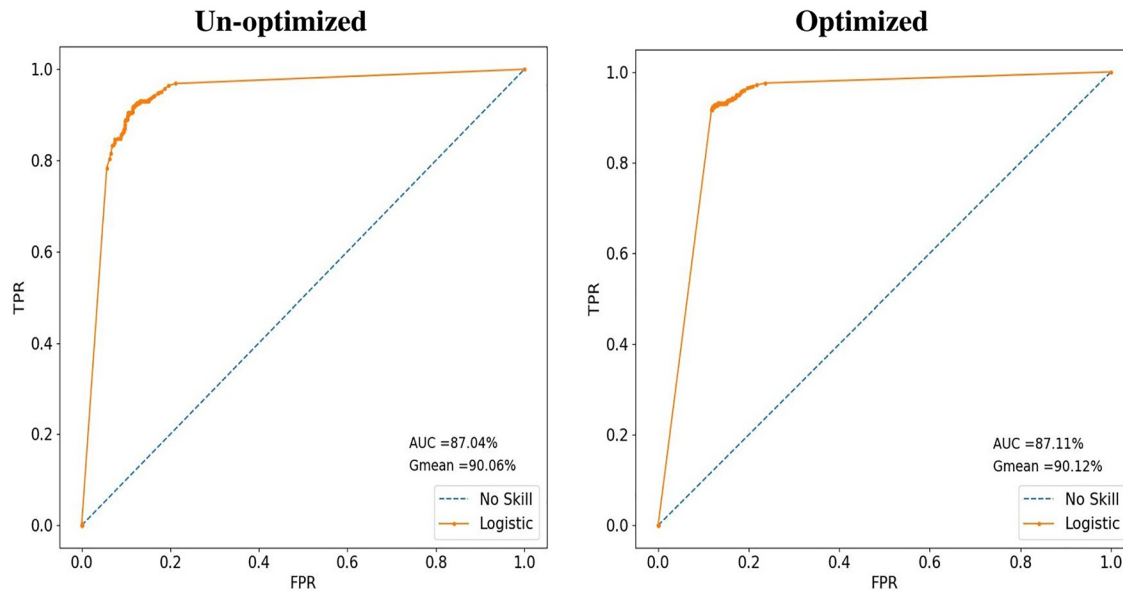
represent the recalibrated reliability diagram, which shows the predicted  $\gamma$  is slightly higher than the observation. To optimize the recalibration map, we run a simulation that recalibrates the predicted  $\gamma$  first before deriving the linear interpolation map. The simulator optimizes the parameters A and B of the calibration function  $\gamma_{cal} = A\gamma + C$  to satisfy the condition

$$\gamma_{cal} - \epsilon \leq \gamma_{ob} \leq \gamma_{cal} + \epsilon$$

where  $\gamma_{ob}$  is the observed frequency. The simulator basically moves the interpolation map to the position of where the distance of  $(\gamma_{cal} - \gamma_{ob})$  is within a tolerance range of  $[-\epsilon, \epsilon]$ . Figure 7 shows the density distributions of  $n(A)$  and  $n(C)$  at different values of A and C for  $\epsilon = 10^{-6}$ . The picked optimum values of A and C are the average maximum values of the obtained distributions. The optimized recalibration map is derived by linearly interpolating the reliability diagram between the optimized  $\gamma_{cal}$  and  $\gamma_{ob}$ . Figure 6b shows the shifting of red squares toward the diagonal line which clearly suggests that the LCNN system has been properly recalibrated. One takes a bit more precautions if the points are close to the diagonal line on the right side, that would be less risky than missing an IR. The interpolation is done by considering the corresponding  $TH^*$  of the recalibrated probabilities. The linear formula of  $\gamma_{cal}$  looks fit for the low frequency predictions ( $<0.5$ ),



**Figure 7.** Density distributions of the parameters A and C as obtained from the simulator. The optimum value of A (C) is considered at the point of the highest peak of the distribution, that is, maximum probability of  $P(A)$  ( $P(C)$ ).



**Figure 8.** ROC curve of the LCNN for different thresholds. The left panel presents the curve for the obtained probabilities. The right panel presents the curve for the re-calibrated probabilities.

but due to the nonlinearly attribute of solar wind features, the linear formula might not fit well for the high frequencies ( $>0.5$ ), that requires more data to enhance the interpolation as well as the recalibration map. The recalibration process is expected to not change the statistical quantities that are calculated in the previous section. Figure 8 shows the ROC curves for the non-calibrated probabilities (left panel) and the optimized recalibrated probabilities (right panel) of the LCNN. The ideal ROC curve is near the upper left corner. For the recalibrated ROC curve, some scores such as AUC, HSS,  $g_{mean}$ , and TSS apparently might increase in the order of  $10^{-1}\%$  and the sensitivity is almost the same. That indicates the recalibrated values should be used to increase the confidence of a probabilistic prediction.

## 6. Discussion and Conclusion

A time series classification method is applied to nowcast IRs from the solar wind data. This work concerns to explain a possible way to exceed the probability of predicting a class by combining different models. We propose a new structure of Long short term memory - Convolution Neural Network (LCNN) classification/prediction system. This LCNN is an ensemble system to predict the probability of arriving an IR on the observer within time NT. The system comprises two neural network classification models in one dimension. In this LCNN, Model 1 uses convolution neural network (CNN) techniques while model 2 uses recurrent neural network RNN-LSTM techniques. The two models work on different scales of the input nowcasting time series and therefore each model learns different features of IRs. The convolutional layer processes the sequential data by exploiting the size of the space between the data points and computes the spatial correlation in data (Wibawa et al., 2022). That allows the convolutional layer to understand the relationships between the different observations in the time series. The LSTM layer can understand long-term dependencies between data points sequence.

Cui et al. (2016) introduced the multi-scale convolution neural network (MCNN) framework for TSC. The MCNN basically generates multi-scale time series from the original time series in order to extract the features of the time series by the convolutional layers. The LCNN system is also designed for TSC, but it combines models of different learning techniques that extract information in different ways. Also, for our case of binary classification, combining convolutional and LSTM models gives a robust classification better than combining two convolutional models with different time scales ( $g_{mean} \sim 86\%$ ). That is because the moving average window method sometimes is not suitable for the short time series or if it contains high outliers noise. In that case, the user should be careful in choosing the width of the moving window. Our concern here is not to compare LCNN with other classifiers, but to compare the LCNN system with the two combined models individually to describe the properties of LCNN

system and its capability to predict the IRs. Prediction of IRs utilizing our system is restricted to the data availability and it is planned to provide immediate nowcasting of IRs based on satellites data.

Both model 1 and model 2 are designed to work in parallel and not combine their layers. Since the models are separated and work independently, the errors do not propagate in the LCNN system during computations and the lost information in one model might be compensated by the other model. Optimizing the LCNN parameters is crucial for not losing information of the original time series. The working method of the filters in the convolutional layers and features reduction are the main causes of losing information (Wei et al., 2015; Zhang & Wang, 2015). Experiments demonstrate the following notices:

- z-score normalization works better than the min-max normalization of formula  $x - x_m / (x_{mx} - x_m)$ , where  $x_{mx}$  and  $x_m$  are the maximum and minimum values of a time series data, for our case of classification. That is because the latter formula scales the data quantities between [0, 1] and when there is an outlier (an abnormally high peak in the solar wind data due to interactions in space), the min-max scale becomes wide, making most of the scaled quantities small comparing to the outlier which in turn makes the difference between the quantities not clear for the algorithm, except for the outliers.
- The probability of the sum of the models' outputs, using formula 1, enhances the outcome predicted probabilities of the LCNN system better than summing the probabilities of the models.
- Increasing the number of CNN and/or RNN models in the LCNN system (with changes in the layers) does not improve the performance significantly and it seems that the saturation of the classifier system is satisfied with the two combined models. The saturation might change with a different case study or for multi-classification tasks.
- Replacing the inputs in such a way run model 1 on the original time series and model 2 on the low-frequency time series does not improve the performance of LCNN but reduces the scores by  $\leq 4\%$  from the obtained scores in Table 2 and increases the instability of the LCNN classifier. That is because the convolution layer learns the relation between the data points but does not learn the long-term trend of the time series. LSTM could learn the trend of the time series however, sometimes it is not sensitive to the fine features of the input data due to the vanishing of the gradient/attractor. This gradient problem is overcome by using the Maxpool layer in model 2 however it does not guarantee preserving the fine features information. That explains the usefulness of combining the two models.
- The normalization layer in the models plays an important role to keep the scale of the feature quantities as represented in the input time series to get a successful training and a small regularization. Batch normalization layer performs not well and that is because it normalizes the data within the input batch intervals and sometimes it vanishes the IRs' quantities of some intervals when there is high variance in the quantities of other intervals (Andreeva & Tsyganenko, 2019; Heinemann et al., 2019). That mostly happens when the input batch contains a mix of IR intervals during solar minimum and solar maximum. The batch normalization depends on the batch size during training the model and the suitable batch size for our problem is 50 to get a good performance.

Analysis shows that the sensitivity and skills of the LCNN system reduce by increasing the nowcasting time NT. To increase the reliability of the nowcasting probabilities, we recalibrated the output predicted probabilities of the LCNN to approach the predicted occurrence frequency of IRs to the observed frequency. LCNN system provides relatively high stability and sensitivity for nowcasting IRs and could be used for predicting and classifying other events. It also paves a way to increase the certainty of a classifier or a predictor using real-time series data in a simple and effective way.

## Appendix A: Formulas of Statistical Quantities

Geometric mean

$$g_{mean} = \sqrt{TPR \cdot TNR}$$

Correct classification

$$C. Corr = \frac{TP + TN}{TP + TN + FP + FN}$$



### True Skill Static

$$TSS = TPR + TNR - 1 = TPR - FPR$$

### Heidke Skill Score

$$HSS = \frac{S_M - S_{\text{standard prediction}}}{S_{\text{perfect}} - S_{\text{standard prediction}}},$$

where  $S_M$  and  $S_{\text{perfect}}$  are the correct score of the model and the perfect score, respectively.  $S_{\text{standard prediction}}$  is the correct score of the standard prediction. It could be calculated from the quantities of classifier metrics as

$$HSS = \frac{2(TP \cdot TN - FP \cdot FN)}{(TP + FN)(FN + TN) + (TP + FP)(FP + TN)}$$

### Data Availability Statement

Data from STEREO A& B are publicly available from the CDAW Data Center at <https://omniweb.gsfc.nasa.gov/hw.html>. The Keras and TensorFlow libraries are publicly available at <https://github.com/fchollet/keras> and <https://www.tensorflow.org/>.

### Acknowledgments

K.A. acknowledges the support from Aberystwyth University through the Aberdoc scheme. E.C. is partially supported by NASA Grants 80NSSC20K1580, 80NSSC20K1275, 80NSSC21K1555. MBK is grateful to ST/S000518/1, PIA.CE.RI. 2020–2022 Linea 2, CESAR 2020-35-HH.0, and UNKP-22-4-II-ELTE-186 grants. The authors acknowledge the following organizations and people for data provision: the teams of STEREO A& B at NASA/CDAW Data Center and the Catholic University of America in cooperation with the Naval Research Laboratory, and both J.H. King and N. Papatashvili for the use of NASA/GSFC space physics data in the NASA-CDAWeb website: <https://cdaweb.gsfc.nasa.gov>.

### References

- Ahmadzadeh, A., Hostetter, M., Aydin, B., Georgoulis, M. K., Kempton, D. J., Mahajan, S. S., & Angryk, R. A. (2019). Challenges with extreme class-imbalance and temporal coherence: A study on solar flare data. arXiv e-prints, arXiv:1911.09061.
- Alielden, K. (2020). Machine learning in satellites monitoring and risk challenges (1st ed.). In *Machine learning and data mining in aerospace technology*: Springer. [https://doi.org/10.1007/978-3-030-20212-5\\_2](https://doi.org/10.1007/978-3-030-20212-5_2)
- Alielden, K., & Taroyan, Y. (2022). Evolution of Alfvén waves in the solar wind. *Monochromatic Driver*, 935(2), 66. <https://doi.org/10.3847/1538-4357/ac7f41>
- Allen, R. C., Ho, G. C., Jian, L. K., Mason, G. M., Vines, S. K., & Lario, D. (2020). Predictive capabilities and limitations of stream interaction region observations at different solar longitudes. *Space Weather*, 18(4), e02437. <https://doi.org/10.1029/2019SW002437>
- Allen, R. C., Ho, G. C., Jian, L. K., Vines, S. K., Bale, S. D., Case, A. W., et al. (2021). A living catalog of stream interaction regions in the Parker Solar Probe ERA. *A&A*, 650(A25). <https://doi.org/10.1051/0004-6361/202039833>
- Allouche, O., Tsoar, A., & Kadmon, R. (2006). Assessing the accuracy of species distribution models: Prevalence, kappa and the true skill statistic (TSS). *Journal of Applied Ecology*, 43(6), 1223–1232. <https://doi.org/10.1029/2019JA027684>
- Andreeva, V. A., & Tsyganenko, N. A. (2019). Empirical modeling of the geomagnetosphere for SIR and CME-driven magnetic storms. *Journal of Geophysical Research: Space Physics*, 124(7), 5641–5662. <https://doi.org/10.1029/2018JA026008>
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2), 157–166. <https://doi.org/10.1109/72.279181>
- Bloomfield, D. S., Higgins, P. A., McAteer, R. T. J., & Gallagher, P. T. (2012). Toward Reliable Benchmarking of Solar Flare Forecasting Methods. *The Astrophysical Journal Letters*, 747(2), L41. <https://doi.org/10.1088/2041-8205/747/2/L41>
- Camporeale, E., Cash, M. D., Singer, H. J., Balch, C. C., Huang, Z., & Toth, G. (2020). A gray-box model for a probabilistic estimate of regional ground magnetic perturbations: Enhancing the NOAA operational geospace model with machine learning. *Journal of Geophysical Research: Space Physics*, 125(11), e2019JA027684. <https://doi.org/10.1029/2019JA027684>
- Camporeale, E., Wing, S., & Johnson, J. (2018). *Machine learning techniques for space weather*. Elsevier.
- Chen, G. M., Xu, J., Wang, W., & Burns, A. G. (2014). A comparison of the effects of CIR-and CME-induced geomagnetic activity on thermospheric densities and spacecraft orbits: Statistical studies. *Journal of Geophysical Research: Space Physics*, 119(9), 7928–7939. <https://doi.org/10.1002/2014JA019831>
- Chi, Y., Shen, C., Luo, B., Wang, Y., & Xu, M. (2018). Geoeffectiveness of stream interaction regions from 1995 to 2016. *Space Weather*, 16(12), 1960–1971. <https://doi.org/10.1029/2018SW001894>
- Chi, Y., Shen, C., Scott, C., Xu, M., Owens, M., Wang, Y., & Lockwood, M. (2022). Predictive capabilities of corotating interaction regions using STEREO and wind in-situ observations. *Space Weather*, 20(7), e2022SW003112. <https://doi.org/10.1029/2022SW003112>
- Cui, Z., Chen, W., & Chen, Y. (2016). Multi-scale convolutional neural networks for time series classification. arXiv preprint arXiv:1603.06995.
- Davis, C. J., Davies, J. A., Owens, M. J., & Lockwood, M. (2012). Predicting the arrival of high-speed solar wind streams at Earth using the stereo heliospheric imagers. *Space Weather*, 10(2), S02003. <https://doi.org/10.1029/2011SW000737>
- DeGroot, M. H., & Fienberg, S. E. (1983). The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1–2), 12–22. <https://doi.org/10.2307/2987588>
- Dorrian, G. D., Breen, A. R., Davies, J. A., Rouillard, A. P., Fallows, R. A., Whittaker, I. C., et al. (2010). Transient structures and stream interaction regions in the solar wind: Results from EISCAT interplanetary scintillation, STEREO HI and Venus express. *ASPERS-4 Measurements*, 265(1–2), 207–231. <https://doi.org/10.1007/s11207-010-9599-z>
- Drewry, A. M., Fuller, B. M., Bailey, T. C., & Hotchkiss, R. S. (2013). Body temperature patterns as a predictor of hospital-acquired sepsis in afebrile adult intensive care unit patients: A case-control study. *Critical Care*, 17(5), 1–11. <https://doi.org/10.1186/cc12894>
- Floriou, K., Kontogiannis, I., Park, S.-H., Guerra, J. A., Benvenuto, F., Bloomfield, D. S., & Georgoulis, M. K. (2018). Forecasting solar flares using magnetogram-based predictors and machine learning. *Solar Physics*, 293(2), 28. <https://doi.org/10.1007/s11207-018-1250-4>
- Galvin, A. B., Kistler, L. M., Popecki, M. A., Farrugia, C. J., Simunac, K. D. C., Ellis, L., et al. (2008). The plasma and suprathermal ion composition (plastic) investigation on the stereo observatories. *Space Science Reviews*, 136(1–4), 437–486. <https://doi.org/10.1007/s11214-007-9296-x>
- Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *JMLR workshop and conference proceedings* (pp. 249–256). Retrieved from <http://proceedings.mlr.press/v9/glorot10a>

- Grandin, M., Aikio, A. T., & Kozlovsky, A. (2019). Properties and geoeffectiveness of solar wind high-speed streams and stream interaction regions during solar cycles 23 and 24. *Journal of Geophysical Research: Space Physics*, *124*(6), 3871–3892. <https://doi.org/10.1029/2018JA026396>
- Graves, A. (2012). Long short-term memory. Supervised sequence labelling with recurrent neural networks.
- Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. arXiv e-prints, arXiv:1706.04599.
- Hapgood, M. A. (2011). Towards a scientific understanding of the risk from extreme space weather. *Advances in Space Research*, *47*(12), 2059–2072. <https://doi.org/10.1016/j.asr.2010.02.007>
- Heinemann, S. G., Temmer, M., Farrugia, C. J., Dissauer, K., Kay, C., Wiegmann, T., et al. (2019). CME-HSS interaction and characteristics tracked from. *Sun to Earth*, *294*(9), 121. <https://doi.org/10.1007/s11207-019-1515-6>
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Horne, R. B., Glauert, S. A., Meredith, N. P., Boscher, D., Maget, V., Heynderickx, D., & Pitchford, D. (2013). Space weather impacts on satellites and forecasting the Earth's electron radiation belts with SPACECAST. *Space Weather*, *11*(4), 169–186. <https://doi.org/10.1002/swe.20023>
- Hundhausen, A. J. (1973). Nonlinear model of high-speed solar wind streams. *Journal of Geophysical Research*, *78*(10), 1528–1542. <https://doi.org/10.1029/JA078i010p01528>
- Hyvärinen, O. (2014). A probabilistic derivation of Heidke skill score. *Weather and Forecasting*, *29*(1), 177–181. <https://doi.org/10.1175/WAF-D-13-00103.1>
- Jeni, L. A., Cohn, J. F., & De La Torre, F. (2013). Imbalanced data recommendations for the use of performance metrics. In *2013 Humaine association conference on affective computing and intelligent interaction* (pp. 245–251). <https://doi.org/10.1109/ACII.2013.47>
- Ji, A., Aydin, B., Georgoulis, M. K., & Angryk, R. (2021). All-clear flare prediction using interval-based time series classifiers. arXiv e-prints, arXiv:2105.01202.
- Jian, L., Russell, C. T., Luhmann, J. G., & Skoug, R. (2006). Properties of stream interactions at one au during 1995–2004. *Solar Physics*, *239*(1), 337–392. <https://doi.org/10.1007/s11207-006-0132-3>
- Jian, L. K., Luhmann, J. G., Russell, C. T., & Galvin, A. B. (2019). *Solar terrestrial relations observatory* (STEREO) observations of stream interaction regions in 2007–2016: Relationship with heliospheric current sheets, solar cycle variations, and dual observations. *Solar Physics*, *294*(3), 1–27. <https://doi.org/10.1007/s11207-019-1416-8>
- Jian, L. K., Russell, C. T., Luhmann, J. G., Galvin, A. B., & Simunac, K. D. C. (2013). Solar wind observations at STEREO: 2007 - 2011. In *Solar wind 13* (Vol. 1539, pp. 191–194). <https://doi.org/10.1063/1.4811020>
- Jian, L. K., Russell, C. T., Luhmann, J. G., Skoug, R. M., & Steinberg, J. T. (2008). Stream interactions and interplanetary coronal mass ejections at 5.3 au near the solar ecliptic plane. *Solar Physics*, *250*(2), 375–402. <https://doi.org/10.1007/s11207-008-9204-x>
- Kaiser, M. L., Kucera, T. A., Davila, J. M., Cyr, O. S., Guhathakurta, M., & Christian, E. (2008). The stereo mission: An introduction. *Space Science Reviews*, *136*(1), 5–16. <https://doi.org/10.1007/s11214-007-9277-0>
- Kavanagh, A. J., Honary, F., Donovan, E. F., Ulich, T., & Denton, M. H. (2012). Key features of >30 keV electron precipitation during high speed solar wind streams: A superposed epoch analysis. *Journal of Geophysical Research*, *117*(A9), A00L09. <https://doi.org/10.1029/2011ja017320>
- Kilpua, E. K. J., Jian, L. K., Li, Y., Luhmann, J. G., & Russell, C. T. (2012). Observations of ICMEs and ICME-like solar wind structures from 2007 - 2010 using near-Earth and STEREO observations. *Solar Physics*, *281*(1), 391–409. <https://doi.org/10.1007/s11207-012-9957-0>
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv, 1412.6980 Retrieved from <https://arxiv.org/abs/1412.6980>
- Koons, H. C., & Fennell, J. F. (2006). Space weather effects on communications satellites. *URSI Radio Science Bulletin*, *2006*(316), 27–41. <https://ieeexplore.ieee.org/document/7909358>
- Krzanowski, W. J., & Hand, D. J. (2009). *Roc curves for continuous data*. Chapman and Hall/CRC. <https://doi.org/10.1201/9781439800225>
- Kubat, M., & Matwin, S. (1997). Addressing the curse of imbalanced training sets: One-sided selection. Retrieved from <https://sci2s.ugr.es/keel/pdf/specific/congreso/kubat97addressing.pdf>
- Li, Y., Zhu, Z., Kong, D., Han, H., & Zhao, Y. (2019). EA-LSTM: Evolutionary attention-based LSTM for time series prediction. *Knowledge-Based Systems*, *181*, 104785. <https://doi.org/10.1016/j.knsys.2019.05.028>
- Lucas, G. M., Love, J. J., & Kelbert, A. (2018). Calculation of voltages in electric power transmission lines during historic geomagnetic storms: An investigation using realistic Earth impedance. *Space Weather*, *16*(2), 185–195. <https://doi.org/10.1002/2017SW001779>
- Luhmann, J. G., Curtis, D. W., Schroeder, P., McCauley, J., Lin, R. P., Larson, D. E., et al. (2008). Stereo impact investigation goals, measurements, and data products overview. *Space Science Reviews*, *136*(1–4), 117–184. <https://doi.org/10.1007/s11214-007-9170-x>
- Marqué, C., Klein, K. L., Monstein, C., Oppenorth, H., Pulkkinen, A., Buchert, S., et al. (2018). Solar radio emission as a disturbance of aeronautical radionavigation. *Journal of Space Weather and Space Climate*, *8*, A42. <https://doi.org/10.1051/swsc/2018029>
- Mayaud, P. N. (1980). *Derivation, meaning, and use of geomagnetic indices*. Washington DC American Geophysical Union Geophysical Monograph Series. <https://doi.org/10.1029/GM022>
- Miyake, W., Saito, Y., Hayakawa, H., & Matsuoka, A. (2005). On the correlation of the solar wind observed at the L5 point and at the Earth. *Advances in Space Research*, *36*(12), 2328–2332. <https://doi.org/10.1016/j.asr.2004.06.019>
- Owens, M. J., & Forsyth, R. J. (2013). The heliospheric magnetic field. *Living Reviews in Solar Physics*, *10*(1), 5. <https://doi.org/10.12942/lrsp-2013-5>
- Peirce, C. S. (1884). The numerical measure of the success of predictions. *Science*, *4*(93), 453–454. <https://doi.org/10.1126/science.ns-4.93.453>
- Piri Sahragard, H., Ajarlo, M., & Karami, P. (2018). Modeling habitat suitability of range plant species using random forest method in arid mountainous rangelands. *Journal of Mountain Science*, *15*(10), 2159–2171. <https://doi.org/10.1007/s11629-018-4898-1>
- Pizzo, V. J. (1980). A three-dimensional model of corotating streams in the solar wind 2. *Hydrodynamic streams*, *85*(A2), 727–743. <https://doi.org/10.1029/JA085iA02p00727>
- Richardson, I. G. (2018). Solar wind stream interaction regions throughout the heliosphere. *Living Reviews in Solar Physics*, *15*(1), 1. <https://doi.org/10.1007/s41116-017-0011-z>
- Thomas, S. R., Fazakerley, A., Wicks, R. T., & Green, L. (2018). Evaluating the skill of forecasts of the near-Earth solar wind using a space weather monitor at L5. *Space Weather*, *16*(7), 814–828. <https://doi.org/10.1029/2018SW001821>
- Thomson, A. W., McKay, A. J., Clarke, E., & Reay, S. J. (2005). Surface electric fields and geomagnetically induced currents in the Scottish power grid during the 30 October 2003 geomagnetic storm. *Space Weather*, *13*(11), S11002. <https://doi.org/10.1029/2005SW000156>
- Tsurutani, B. T., Gonzalez, W. D., Gonzalez, A. L. C., Guarnieri, F. L., Gopalswamy, N., Grande, M., et al. (2006). Corotating solar wind streams and recurrent geomagnetic activity: A review. *Journal of Geophysical Research: Space Physics*, *111*(A7), A07S01. <https://doi.org/10.1029/2005JA011273>
- Tsurutani, B. T., Gonzalez, W. D., & Kamide, Y. (1997). Magnetic storms. *Surveys in Geophysics*, *18*(4), 363–383. <https://doi.org/10.1023/A:1006555215463>

- Wang, J., Jiang, W., Li, Z., & Lu, Y. (2021). A new multi-scale sliding window LSTM framework (MSSW-LSTM): A case study for GNSS time-series prediction. *Remote Sensing*, *13*(16), 3328. <https://doi.org/10.3390/rs13163328>
- Wei, D., Zhou, B., Torralba, A., & Freeman, W. (2015). Understanding intra-class knowledge inside CNN. arXiv e-prints, arXiv:1507.02379.
- Wibawa, A. P., Utama, A. B. P., Elmunsyah, H., Pujianto, U., Dwiyanto, F. A., & Hernandez, L. (2022). Time-series analysis with smoothed convolutional neural network. *Journal of Big Data*, *9*(1), 1–18. <https://doi.org/10.1186/s40537-022-00599-y>
- Yu, B., Scott, C. J., Xue, X., Yue, X., Chi, Y., Dou, X., & Lockwood, M. (2021). A signature of 27 day solar rotation in the concentration of metallic ions within the terrestrial ionosphere. *The Astrophysical Journal*, *916*(2), 106. <https://doi.org/10.3847/1538-4357/ac0886>
- Zhang, D., & Wang, D. (2015). Relation classification via recurrent neural network. arXiv e-prints, arXiv:1508.01006.
- Zhang, X., Liang, X., Zhiyuli, A., Zhang, S., Xu, R., & Wu, B. (2019). At-LSTM: An attention-based LSTM model for financial time series prediction. *IOP Conference Series: Materials Science and Engineering*, *569*(5), 052037. <https://doi.org/10.1088/1757-899x/569/5/052037>