Acta Astronautica 193 (2022) 595–606

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

A deep learning approach to solar radio flux forecasting

Emma Stevenson [a,b,*], Victor Rodriguez-Fernandez [a], Edmondo Minisci [c], David Camacho [a]

[a] School of Computer Systems Engineering, Universidad Politécnica de Madrid, Calle de Alan Turing, 28038 Madrid, Spain
[b] School of Aeronautical and Space Engineering, Universidad Politécnica de Madrid, Plaza del Cardenal Cisneros, 3, 28040 Madrid, Spain
[c] Department of Mechanical and Aerospace Engineering, University of Strathclyde, 75 Montrose Street, Glasgow, G1 1XJ, United Kingdom

ARTICLE INFO

Keywords:
Solar radio flux
Space weather
Deep learning
Time series forecasting
Ensemble

ABSTRACT

The effect of atmospheric drag on spacecraft dynamics is considered one of the predominant sources of uncertainty in Low Earth Orbit. These effects are characterised in part by the atmospheric density, a quantity highly correlated to space weather. Current atmosphere models typically account for this through proxy indices such as the F10.7, but with variations in solar radio flux forecasts leading to significant orbit differences over just a few days, prediction of these quantities is a limiting factor in the accurate estimation of future drag conditions, and consequently orbital prediction. In this work, a novel deep residual architecture for univariate time series forecasting, N-BEATS, is employed for the prediction of the F10.7 solar proxy on the days-ahead timescales relevant to space operations. This untailored, pure deep learning approach has recently achieved state-of-the-art performance in time series forecasting competitions, outperforming well-established statistical, as well as statistical hybrid models, across a range of domains. The approach was found to be effective in single point forecasting up to 27-days ahead, and was additionally extended to produce forecast uncertainty estimates using deep ensembles. These forecasts were then compared to a persistence baseline and two operationally available forecasts: one statistical (provided by BGS, ESA), and one multi-flux neural network (by CLS, CNES). It was found that the N-BEATS model systematically outperformed the baseline and statistical approaches, and achieved an improved or similar performance to the multi-flux neural network approach despite only learning from a single variable.

1. Introduction

The dynamics of space objects orbiting in Low Earth Orbit (LEO) strongly depend on the characterisation of the uncertainties on the initial state, physical properties of the objects themselves (such as mass and shape) and properties of the atmosphere, chiefly the density. These atmospheric properties are strongly influenced by both solar and geomagnetic activities, whose forecasting is therefore of paramount importance for space operations, and whose forecast uncertainties are fundamental to properly characterise the uncertainties on the orbital states of spacecraft and space debris. As such, the prediction of these quantities has fundamental implications both in the day-to-day management of operational spacecraft such as collision avoidance [1], and also in the longer term, in re-entry prediction [2].

Typical atmospheric density models, which are used to model the dynamics of space objects, capture the space weather conditions using two types of proxies, one for the solar activity and one for the geomagnetic activity. The atmospheric density is predominantly influenced by the solar activity, or the so-called Extreme Ultra Violet (EUV) irradiance [3]. The solar EUV is highly energetic and is absorbed by the upper atmosphere, which is subsequently heated up and ionised (creating the ionosphere), driving a change in the atmospheric density. However, as direct measurements of the solar EUV cannot be made on ground, such models rely on correlated proxy measures such as the F10.7 radio flux, which is a measurement of the intensity of solar radio emissions with a wavelength of 10.7 cm (a frequency of 2800 MHz) [4]. This quantity has a very long time series history, with data covering many decades, and as such is still the most common solar proxy for typical atmosphere models [5].

As a consequence, there have been a number of studies that have investigated and developed empirical time series forecasting methods and services for predicting the F10.7. Of these, there have been a variety of efforts using both statistical [6–10], and Machine Learning (ML) [11,12] approaches. The popularity of machine learning in the field of space weather forecasting as a whole has grown significantly in recent years [13], owing to its ability to exploit large amounts of available data and capture non-linearity. However, unlike geomagnetic proxy forecasting, the use of these techniques is not yet the universal standard in solar proxy forecasting, with many operationally available

* Corresponding author at: School of Computer Systems Engineering, Universidad Politécnica de Madrid, Calle de Alan Turing, 28038 Madrid, Spain.
E-mail address: emma.stevenson@upm.es (E. Stevenson).

https://doi.org/10.1016/j.actaastro.2021.08.004
Received 14 March 2021; Received in revised form 19 June 2021; Accepted 2 August 2021
Available online 8 August 2021

forecasts still relying on statistical techniques. Moreover, many of the considered approaches on the machine learning side focus on the application of classical approaches such as Support Vector Regression (SVR) [11], or single layer feedforward neural networks [12].

However, within the last year there have been significant advancements in the field of time series forecasting by way of deep learning. More specifically, in [14], Oreshkin et al. presented N-BEATS, a deep residual architecture for univariate time series forecasting, which, for the first time, succeeded in outperforming winning approaches of recent forecasting competitions across a range of domains, which were all previously based on either statistical or hybrid (statistical + ML) methods. Its success is due to a unique architecture that combines a deep stack of fully-connected layers, backward and forward residual links, aggregation of the partial forecasts in a hierarchical fashion, and ensembling. Being a pure deep learning approach implies that, unlike statistical approaches, there is no expert knowledge, or ad-hoc feature engineering, required on the data itself in order to train the model.

Given the promising performance of this state-of-the-art architecture across a range of domains, in this work we apply N-BEATS to the daily prediction of the F10.7 solar proxy and examine its feasibility over forecast horizons relevant to space operations, from 3 days for activities such as collision avoidance, up to 27 days for activities such as re-entry campaigns. To the best of our knowledge, this is the first time deep residual networks have been applied to the forecasting of solar proxies. Furthermore, we extend this approach with non-intrusive uncertainty quantification using deep ensembles [15,16]. Finally, we perform a systematic comparison of the forecasts generated using this pure deep learning approach to those generated using other data-driven approaches, both statistical and ML, and show that it can produce improved single point forecasts, as well as useful forecast uncertainty estimates.

The main contributions of this work can be briefly summarised as follows:

- The use of a state-of-the-art deep neural network (N-BEATS) to forecast future values of the F10.7 using only its past history, with no additional variables and no requirement for domain-specific knowledge of the data.
- The use of deep ensembles to improve the accuracy of the forecasts and to provide a measure of model uncertainty alongside the single point predictions.
- A detailed systematic comparison with operationally available forecasts, which emphasises the strengths and weaknesses of this approach and paves the way for future work. To this end, the forecasts provided by our approach, along with the code to reproduce the experiments of this paper, are publicly available in a Github repository,[1] to enable further research and comparisons.

The paper is structured as follows. In Section 2 we provide backgrounds on the state of the field of time series forecasting, with a particular emphasis on its use in relation to space weather activities. In Section 3, the proposed approach is described, which includes not only the explanation of the deep learning architecture employed, N-BEATS, but also the way the data is extracted and passed to the model, the training and evaluation procedures, and the estimation of the prediction intervals through an ensemble of trained models. Section 4 makes a detailed comparison of the proposed approach with respect to current operationally available forecasts, comparing both the values of the predicted data points in the future, and the uncertainty intervals. Finally, in Section 5 we discuss the conclusions of the paper, and outline avenues for future research.

[1] https://github.com/stardust-r/deep-learning-space-weather-forecasting.

2. Backgrounds on time series forecasting

The goal of time series forecasting is to predict the values of a set of future data points given a set of past observations. There are multiple types of forecasting, depending on different criteria:

- The *number of series to predict*. The term univariate time series forecasting refers to making predictions on one single series, regardless the number of input variables used. On the other hand, multivariate time series forecasting refers to the prediction of several related series at once.
- The *number of time steps to predict*, also known as the horizon (H). In contrast to one-step-ahead predictions, multi-horizon forecasting predicts the variables of interest at multiple future time steps, thus providing decision makers with an estimate that can be used to optimise their course of action across an entire path of predictions. The number of time steps used to create the prediction is then known as the lookback.
- The *uncertainty estimation* provided by the forecasting model. Single point models focus on estimating, as precisely as possible, the future point values. However, in many scenarios [17], the provision of uncertainty intervals can be useful, if not critical, for risk management, by giving decision makers an indication of likely best and worst-base values that the target can take.

Although there have been recent attempts to create a meaningful distinction between forecasting methods [18], these can be roughly classified as being either of a statistical or machine learning nature. Statistical methods make use of statistics based on historical data to predict what will happen in the future. They are normally computationally efficient as they rely on linear processes to minimise the prediction error, and require expert knowledge about the trend and the seasonality of the data to model. Traditional and popular examples of these methods include ARIMA [19] and Exponential smoothing (ETS) [20] models. On the other hand, ML methods tackle the problem of forecasting as a supervised learning (auto)regression task, where the model is trained on pairs of past/future values from different slices of the time series. They are computationally more demanding and rely, in many cases, on non-linear training algorithms. In a "pure" ML method, the main advantage is that no time series specific engineering is needed to train the model.

Among the several ML methods that can be used for time series forecasting, neural networks, and more specifically, deep neural networks, are one of the most popular alternatives in the recent literature, due to the latest breakthroughs in Deep Learning (DL) [21]. A neural network is an artificial model that emulates how the human brain works, using an abstract (or simplified) mathematical model of a neuron. It consists of a series of such neurons connected to each other with a series of weights. These weights are learned from the training data, using a learning algorithm that updates them in order to minimise the loss (or error) of the network predictions summed over all training cases. The most common type of neural network is the feedforward neural network, where the information enters into the input units, and flows in one direction through the hidden layers until it reaches the output units. Although the universal approximation theorem [22] shows that any function can be well-approximated using a feedforward neural network with just one hidden layer of non-linear neurons, in practice, deeper architectures (with more than one hidden layer) have smaller matrices, making it possible to split the derivative of the loss function into pieces, meaning that the model can be trained more quickly and will take up less memory [23]. In addition, the layered structure of a deep neural network enables the automatic extraction of features from the data at different levels of abstraction, the later layers being the most specialised for the task at hand. In the field of time series forecasting, the most common deep learning architectures are those based on Recurrent Neural Networks (RNNs), whose units contain an

internal memory state which acts as a compact summary of past information [24]. In recent years, the development of attention mechanisms and the Transformer architecture [25] has also lead to improvements in temporal dependency learning, from which time series forecasting has benefited [26].

Despite all of this, the use of ML and DL methods are far from being the standard for the task of time series forecasting. For instance, in the 2018 forecasting competition M4,[2] which challenges researchers to forecast time series data over multiple domains, 12 out of the 17 most accurate solutions were ensembles of classical statistical methods [27], and only six of the submissions were pure ML. Only in the last few months, when the 2020 M5 competitions[3,4] were concluded, could it finally be seen that ML methods were part of the top solutions of the leaderboard, representing a significant step forward in the implantation of ML for time series forecasting.

2.1. Forecasting the F10.7 proxy

Given the importance of the F10.7 proxy to atmospheric density modelling, many research works have been carried out to derive and test forecasting methods and approaches. Support Vector Regression (SVR) was used for short-term forecasting of F10.7 [11], with the authors showing that "*the proposed approach can perform well by using fewer training data points than the traditional neural network*". A simple linear forecasting model for the F10.7 proxy has been proposed by Warren et al. [8]. In this paper the authors also compared the linear forecasting approach of the F10.7 to forecasting using artificial neural networks, and preliminarily concluded that "*forecasting via sophisticate artificial neural networks is not any better than a simple linear forecasting approach*". Various empirical time series prediction techniques were compared in [12]. The authors selected a multi-wavelength, non-recursive, analogue neural network, and found that "*the prediction of the 30 cm flux, and to a lesser extent that of the 10.7 cm flux, performs better than NOAA's present prediction of the 10.7 cm flux, especially during periods of high solar activity*". A linear multi-step forecasting model based on the correlation between different forecasting steps and the characteristic of heteroscedasticity is proposed in [9]. In the same paper, a variational Bayesian procedure to optimise the model is also introduced, and it is claimed that the proposed model improves the performance of multi-step F10.7 forecasting by considering correlation and heteroscedasticity. More recently, a thorough analysis of the power of statistical ARIMA models for the forecasting of this proxy was carried out in [10], proving that, as long as the order p of the ARIMA model is optimally chosen, the model is not inferior to other techniques. The importance of benchmarking these models, especially with a view to use in atmospheric density models and operational applications such as collision avoidance, is now beginning to be recognised in this field [28].

To the best of our knowledge, the reception of deep neural networks to forecast the solar flux is limited, especially in terms of the F10.7 proxy. Most of the work in the intersection of solar activity and deep learning is focused on the early classification of solar flares and geomagnetic storms. As an example, Long Short Term Memory (LSTM) architectures (a subclass of RNNs) have been employed for the detection of geomagnetic storms based on the K_p index in [29], and in [30], Convolutional Neural Networks (CNNs) are trained to classify flaring and nonflaring active regions using line-of-sight magnetograms. Only in the last few months, the task of forecasting future values of the GOES X-ray flux has been studied with different deep learning architectures [31], including N-BEATS, the architecture used as a basis for this work.

3. Applying deep learning to F10.7 forecasting

In this section, we present our approach to provide daily, univariate, multi-horizon forecasts with prediction intervals of the F10.7, using only past information of the proxy, with no additional input variables. It is an end-to-end deep learning approach based on the novel architecture N-BEATS. For the sake of reproducibility, the implementation of this approach is publicly available on Github.[5]

3.1. Model architecture: N-BEATS

N-BEATS (Neural Basis Expansion Analysis for interpretable Time Series forecasting) [14] is a novel deep learning architecture for single point, univariate, multi-horizon forecasting that has been gaining traction in the field since it was proved to be the first pure deep learning method that outperforms the winning approaches of recent forecasting competitions. It does not need any specific expert knowledge on the data, and is thus applicable to a wide array of target domains without any feature engineering. An open source implementation of N-BEATS,[6] written with the deep learning library PyTorch, has been employed in this work.

N-BEATS belongs to the family of deep residual networks, which were introduced for computer vision tasks as a way to train very deep networks effectively [32]. More specifically, its topology is described as doubly residual stacking (see Fig. 1), where each stack consists of multiple residual blocks that produce two outputs: the block's estimation of the input (or lookback) data, called backcast, and the estimation of the future values across the desired horizon, known as forecast. The backcast is subtracted from the current's block input, forming a residual which then serves as input to the next block in the stack. The output of the network is the result of a hierarchical aggregation of the forecasts across stacks, i.e., first the partial forecasts of each block are aggregated at the stack level and then at the overall network level, providing the final global output. The iterative and residual nature of this architecture aims to encourage gradual signal reconstruction and forecasting.

Internally, each basic residual block within a stack consists of a multi-layer fully connected network with non-linear (ReLu) activation functions between each layer, although there are some extensions of N-BEATS that replace this basic building block with temporal aware structures such as RNNs [33]. The fully connected network outputs two vectors of basis expansion coefficients, normally referred to as θ, which are then accepted by two learnable basis functions to generate the final backcast and forecast of the block, respectively. The parameters of the basis functions can be constrained so that only a family of functions can be learnt (e.g., low-degree polynomials or Fourier series), forcing the model to decompose the forecast into distinct human interpretable outputs such as the trend and the seasonal components of the data. On the other hand, the parameters of these functions can be left unconstrained (or generic, as it is known in the N-BEATS paper), with the aim of using no domain knowledge in the modelling.

3.2. Data and model inputs

The European Space Agency (ESA) Space Weather Service Network[7] maintains a database containing both past, and forecast, values of solar and geomagnetic indices which are relevant to drag calculations. This data is compiled from a number of independent providers in one place, for the convenience of end users and space operators, as a part of its Space Surveillance and Tracking Service.

We use this service to extract the time series of the F10.7, measured in solar flux units (sfu), which has been measured continuously since

[2] https://www.kaggle.com/yogesh94/m4-forecasting-competition-dataset.
[3] https://www.kaggle.com/c/m5-forecasting-accuracy.
[4] https://www.kaggle.com/c/m5-forecasting-uncertainty.
[5] See footnote 1.
[6] https://github.com/philipperemy/n-beats.
[7] http://swe.ssa.esa.int/.

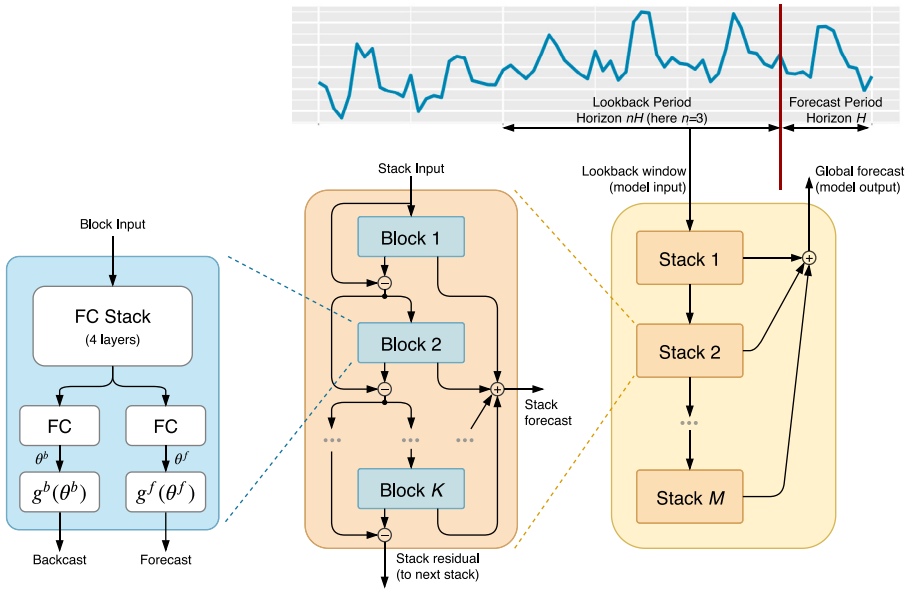

Fig. 1. N-BEATS architecture diagram, as shown in the original paper by Oreshkin et al. [14].

1947 by the Ottawa, and then Penticton Radio Observatories [4]. The F10.7 is available in either *observed* values, which vary throughout the year with the Sun–Earth separation, or *adjusted* values, where the observations are adjusted to 1 AU (Astronomical Unit). In this work, we train our forecasting model using the *observed* data, which is used by typical thermosphere models [5,12]. Daily values of the F10.7 were taken to be those measured at 20:00 where appropriate, and missing values, more prevalent early in the time series, linearly interpolated.

The data is then split into training and validation subsets. However, due to the correlated nature of time series data, the typical ML strategy of random splitting, that ensures that the underlying distribution in these subsets is the same, cannot be used. We must therefore ensure that the validation set contains a full solar cycle so that it is representative of the training data. To achieve this, we use an approximate 80% to 20% splitting strategy, with the training set covering the period from 01/01/1950 to 20/10/2006, and the validation set covering 20/10/2006 to 01/10/2020, covering Solar Cycle 24, as shown in Fig. 2.

As we are using deep learning, for which successive feature extraction through the layers of the architecture is implicit, this data does not require extensive pre-processing in order for the model to perform well. As such, the input data needs only to be normalised, to prevent exploding gradients and improve the numerical stability of the model, and subdivided into lookback-horizon windows. No explicit knowledge or analysis of time series features such as trend and seasonality is required in advance.

3.3. Model training and evaluation

The underlying architecture of a deep learning model is defined by a series of hyperparameters which can be pre-set by the user, and are not learnt by the model during training. These parameters constrain the complexity of the model and should be optimised, or tuned, to find the optimal configuration for a specific problem such that the model does not under or overfit the training data.

In the case of the N-BEATS architecture, as can be seen in Fig. 1, there are a large number of potential hyperparameters that can be tuned. However, one of the distinguishing aspects of this architecture is that it was specifically designed to be generally applicable across a wide variety of horizons and datasets, and should perform well without the need for extensive tuning.

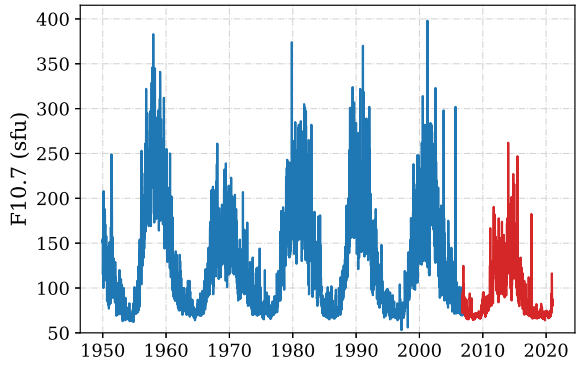

Fig. 2. Splitting of the F10.7 time series into training (blue) and validation (red) datasets. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
Hyperparameter settings for the N-BEATS architecture and training procedure.

Parameter	Value
Basis type	Generic
Number of stacks	30
Dimension of basis coefficients (θ)	7,8
Share weights in stack	False
Number of blocks per stack	1
Number of layers per block	4
Number of hidden units per layer	64
Activation function	ReLu
Optimiser	RAdam
Learning rate	1e−3
Weight decay	0
Batch size	Full batch

As such, we investigated architectural hyperparameters (for example the number of stacks, number of blocks per stack, number of layers etc.) similar to those recommended by the authors [14], and also focused on basis choice and optimisation hyperparameters such as learning rate. Tuning was performed through a grid search, using the experiment-tracking tool Weights & Biases [34], and the chosen parameters are given in Table 1.

Notably, we found that constraining the bases to trend (polynomial) and seasonality (Fourier) components (in an interpretable approach,

as described in Section 3.1) significantly hindered model performance compared to the generic approach, where the model has free reign to learn the preferred basis. By not constraining the *Basis Type*, the final model therefore does not rely on domain specific knowledge.

This analysis was performed using the Mean Squared Error (MSE), the nominal metric used to evaluate the performance of regression problems in machine learning, which is defined as follows,

$$\text{MSE} = \frac{1}{H}\sum_{i=1}^{H}(\hat{y}_{T+i} - y_{T+i})^2, \tag{1}$$

in which $\vec{\hat{y}}$ are the set of predicted future values of a time series of length T over a forecast horizon of length H, and $\vec{y}$ are the set of true observed values over the horizon, $\vec{y} = [y_{T+1}, y_{T+2}, \cdots, y_{T+H}]$.

For a more robust and systematic approach to model evaluation, several additional metrics will also be considered in this work, which capture different aspects of the model performance.

Firstly, we include the Mean Absolute Percentage Error (MAPE), and Mean Absolute Scaled Error (MASE), which are standard scale-free metrics used more specifically in the field of time series forecasting, and used by [14] to enable performance comparison over a range of different datasets,

$$\text{MAPE} = \frac{100}{H}\sum_{i=1}^{H}\frac{\left|\hat{y}_{T+i} - y_{T+i}\right|}{\left|y_{T+i}\right|}, \tag{2}$$

$$\text{MASE} = \frac{1}{H}\sum_{i=1}^{H}\frac{\left|\hat{y}_{T+i} - y_{T+i}\right|}{\frac{1}{T+H-m}\sum_{j=m+1}^{T+H}\left|y_{j-m} - y_j\right|}. \tag{3}$$

These are linear metrics, which means that unlike the squared MSE, they do not give as much weighting or importance to larger errors, which are typically associated with higher levels of solar activity.

The MASE is equivalent to the Mean Absolute Error (MAE), scaled by the average error of a naive baseline model whose forecast is simply a previously observed value m periods in the past. If there is no prior knowledge of the seasonality of the time series, m can be set to 1 and the naive model is that of the persistence. As we do not want our analysis to depend on any pre-existing domain knowledge, and as the solar flux encompasses multiple seasonalities, we consider the persistence as our baseline model here in the definition of the MASE, and later in Section 4.1.

Next, we consider the recommendations of [35], who proposed a standardised set of comparison metrics for benchmarking geomagnetic index prediction models. In this way, we hope to enable and encourage more transparent and systematic comparisons between pre-existing models by future authors.

We therefore also include the Pearson linear correlation coefficient (R),

$$R = \frac{\text{cov}(\hat{y}, y)}{\sigma_{\hat{y}}\sigma_y}, \tag{4}$$

and the Mean Error (ME), or bias,

$$\text{ME} = \frac{1}{H}\sum_{i=1}^{H}(\hat{y}_{T+i} - y_{T+i}), \tag{5}$$

which gives an indication as to whether the model, on average, over-predicts (positive bias) or underpredicts (negative bias) the observed data. We also include the MAE and Root Mean Squared Error (RMSE), the square root of Eq. (1), to be consistent with the recommended guidelines, and to be comparable to other authors who may choose these metrics.

Finally, we introduce the concept of the *Relative* metric, which is an extension of a metric suggested by Yaya et al. in [12]. In the case that the model is trained separately for each horizon, using the above metrics will result in a set of performance metrics for different horizons. However, in order to obtain a single metric over all horizons, the performances must be scaled to prevent higher horizons, with higher errors, dominating the final value. We therefore define the relative metric as the average, over all horizons, of the ratio of the model performance to that of the persistence,

$$\text{Relative } X = \frac{1}{H_{\max} - H_{\min} + 1}\sum_{h=H_{\min}}^{H_{\max}}\frac{X_{\text{model},h}}{X_{\text{persistence},h}}, \tag{6}$$

where X can be any of the above metrics, and $H_{\min}, H_{\max}$ are the minimum and maximum horizons of interest, which in our case are 3 and 27 days respectively.

Table 2
Ensemble parameters per horizon, H. For each horizon, individual models are trained on lookback windows of different lengths, with different loss functions (as defined in Eqs. (1)–(3)) and random initialisations, resulting in 90 individual models which are then aggregated to form the ensemble prediction.

Parameter	#	Values
Lookback period	6	$[H, 2H, \ldots, 6H]$
Loss function	3	MSE, MAPE, MASE
Initialisation	5	Random

3.4. Ensemble forecasting and uncertainty quantification

Ensemble forecasting is a technique that has long been used in terrestrial weather forecasting [36], and is also employed in all leading submissions in time series forecasting competitions [14], owing to its ability to not only improve accuracy, but also to improve the reliability of such forecasts by providing an inherent measure of model uncertainty [37].

This is achieved by averaging the predictions over a diverse set of models to create a single more-accurate model, with an associated uncertainty, that has several additional advantages. For example, by providing a range of possible outcomes, this approach can yield a better understanding of extreme events, such as solar storms. It can also be used to account for both uncertainty in the model inputs (aleatoric uncertainty), and the propagation of uncertainty inherent in the model itself (epistemic uncertainty, as considered in this work) in the resultant forecast uncertainty [36].

However, one of its greatest benefits when employing deep learning techniques, is its ability to improve the out-of-distribution robustness of the model [16]. Due to their large network complexity, deep learning models are particularly susceptible to overfitting, from which the model does not generalise well to new data. Regularisation techniques can be used to overcome this, and in the case of the N-BEATS architecture, ensembling was found to be more powerful than typical techniques such as drop out, or weight penalties [14].

We adopt an approach similar to that recommended in [14], building the ensemble from a set of models that have the same underlying architecture (which is chosen through hyperparameter tuning, see Section 3.3), but different higher level training parameters. For this we use three main sources of diversity: length of input window (lookback), choice of loss function (the error used internally during training, see Section 3.3), and choice of weight initialisation, as described in Table 2.

In this way, the resulting ensemble can account for trends in the data over different time scales, and account for model bias and variance arising from the training procedure. This procedure is iterative and stochastic, and therefore the initial values of the weights strongly determine which local optima is found. Varying the initialisations, and the search space itself by varying the loss function, therefore improves the performance of the ensemble as a whole by averaging out weaker solutions. Injecting randomness in the initialisation is also particularly important when using machine learning techniques for time series forecasting, as the sequential nature of the data prevents the usual practice of shuffling the training dataset prior to each epoch in order to provoke changes in the gradient estimate of the optimiser.

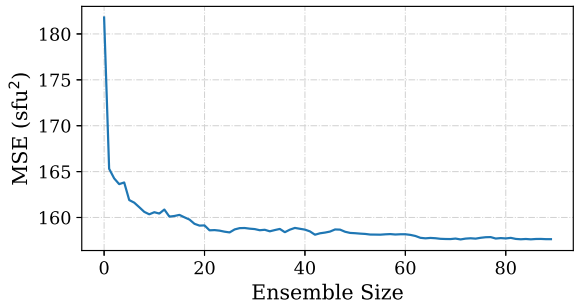

Fig. 3. Performance of N-BEATS 27-day forecast as a function of ensemble size.

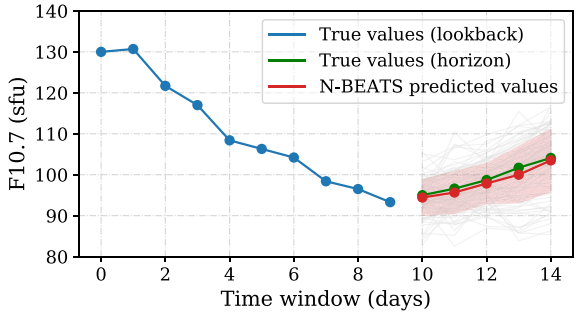

Fig. 4. An example of a 5-day forecast of the F10.7 generated by N-BEATS (true values over the forecast horizon shown in green, N-BEATS prediction in red) with an example 10 day lookback window (shown in blue). The associated 1-sigma uncertainty of the forecast generated using the ensemble approach is also shown in red, and the individual ensemble members in grey. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

As shown in Table 2, for every forecast horizon, we generate 90 individual models, which are then combined using the mean as the ensemble aggregation function, to generate the final forecast. In this approach, we use the same random initialisations over all horizons, and do not use bootstrapping,[8] in order to ensure that each model is trained with as large a dataset as possible. Such an approach has been shown to perform well in practice compared to traditional bagging procedures [15]. It can be seen from Fig. 3, that using this ensemble approach improves the performance of the overall model, and that the number of individual models we consider (90) is sufficient.

An example of a 5-day N-BEATS ensemble forecast generated using the approach described in this section, which will simply be denoted as N-BEATS for the remainder of this paper, can then be seen in Fig. 4. Here, we show an example of the lookback period used for the prediction (10 days, $2H$, as one of the lookbacks used in the ensemble), and the N-BEATS ensemble prediction over the forecast horizon. The 1σ uncertainty band naturally arises from the distribution of forecasts over the ensemble (each of which is shown here in grey to illustrate the concept), and can be seen to encompass the true values of the F10.7 over the horizon.

4. Comparison with operationally available forecasts

In this section, the performance of the N-BEATS ensemble approach described in Section 3, is compared to operationally available forecasts that comprise both statistical and machine learning approaches. The models themselves are described in Section 4.1, with the results of the comparisons in terms of single point forecasting and uncertainty estimation discussed in Sections 4.2 and 4.3 respectively.

4.1. Forecast model descriptions

Here, we describe the F10.7 forecasts used for the comparison with N-BEATS. They are publicly available, and updated on a daily basis.

4.1.1. Persistence (Baseline)

The persistence model forecasts always the last observed value, i.e., $\hat{y}_{T+i} = y_{T+i-1} \forall i \in \{1, \dots, H\}$. It is a simple model that performs reasonably well, especially for low horizons, and thus is a common baseline to use for comparison in multiple related works [12]. Setting $m = 1$ in the MASE metric (See Eq. (3)) can be thought of as comparing a certain set of predictions against the performance of the persistence model.

4.1.2. BGS (ESA)

In 1993, the Geomagnetism Group of the British Geological Survey (BGS) carried out work under contract to ESA to investigate forecasting techniques for predicting solar and geomagnetic activity [6]. As part of this work, they constructed a software for the forecasting of the F10.7 proxy up to 27 days ahead. This software uses an ARIMA model [19] with 60 coefficients, which are recalculated daily to reflect changing solar and geomagnetic conditions, using the preceding two years of data.

4.1.3. CLS (CNES)

The Collecte Localisation Satellites (CLS), a subsidiary of the French Space Agency (CNES), provides forecasts of the F10.7 using a model developed from their research, published in [12]. Although their method is similar to the approach proposed in this work in the sense that both are based on neural networks, there are two main differences:

1. Unlike the approach presented here, CLS use additional input variables aside from the F10.7 to compute the forecasts. More specifically, multiple wavelengths (8.2 cm, 10.7 cm, 15 cm and 30 cm) of the solar radio flux are included.
2. The architecture employed cannot be considered as a deep neural network, since it only has one hidden layer. This layer then relies on a logistic activation function, while N-BEATS relies on ReLu. Additionally, the architecture is based on a feedforward approach, which differs to the deep residual approach used by N-BEATS.

4.2. Comparison of single point forecasting

In this section, we present a comparison of the model performances in terms of single point forecasting. The analysis is comprised of two subsections.

First, we consider forecasts provided by the ESA Space Weather Service Network.[9] These forecasts, which include BGS, are only available since late 2016, and therefore this analysis is performed on a reduced validation set covering the period 26/11/2016 to 01/10/2020.

The second section then contains an extended comparison, covering a full solar cycle, between N-BEATS and CLS, whose complete forecast archive is publicly available.[10] This covers the full validation set shown in Fig. 2, from 20/10/2006 to 01/10/2020.

In these sections, the N-BEATS, persistence and BGS single point forecasts are evaluated against the observation dataset described in Section 3.2. The CLS forecasts are evaluated against their own archive of observations, in order to be consistent with the data that they used during model training.

[8] Bootstrapping is a resampling technique that draws samples N times uniformly with replacement from a dataset with N items.

[9] http://swe.ssa.esa.int/.

[10] https://spaceweather.cls.fr/.

Table 3
Relative metric comparison of N-BEATS with operationally available forecasts for 2016–2020. Metrics are scaled against the persistence baseline, and averaged over forecast horizons. Lower error metrics and higher correlation metrics are preferred, with a value of 1 exhibiting the same performance as the persistence baseline. The best performing values in each metric are highlighted in bold.

Model	Relative metric				
	MSE	RMSE	MAPE	MAE	R
BGS	0.852	0.920	0.926	0.923	1.193
CLS	0.801	0.890	1.123	1.125	1.032
N-BEATS	**0.657**	**0.808**	**0.911**	**0.901**	**1.354**

4.2.1. 2016–2020 validation period

This analysis covers a period of relatively low solar activity, as seen in Fig. 2, but over which we can compare N-BEATS to all the models described in Section 4.1. During this period, a small number of BGS forecasts are missing, and thus these windows are also removed from the other models for a fair comparison.

In Fig. 5a, we show the evolution of different performance metrics, defined in Section 3.3, for different N-BEATS models with these available operational forecasts. RMSE and MAPE are error metrics, and so better performing models have lower errors. As expected, the errors increase with horizon, as we forecast further forward in time. The opposite is true of the Pearson correlation coefficient, R, where higher values are desirable, with a value of 1 indicating a perfect linear correlation between the observed and predicted values of the models.

It can be seen that for the reduced validation set in Fig. 5a, the N-BEATS ensemble model gives consistently good results up to a forecast horizon of 27 days, outperforming the baseline persistence model, statistical BGS approach, and neural network CLS model in RMSE and R, and showing a similar or improved performance over all models in all displayed metrics.

To infer the overall best performing model over this period, we use relative metrics, as defined in Eq. (6), to obtain a set of single performance metrics which are averaged over all forecast horizons. As described in Section 3.3, these are scaled against the persistence at each horizon before they are averaged to ensure that the final metrics are not weighted too heavily towards larger, more error prone horizons, and therefore a relative metric value of 1 means that the model exhibits the same performance as the persistence baseline. As such, models exhibiting better performance than the persistence in error metrics will have a relative metric less than 1, but those exhibiting better performance in R will have a score greater than 1. The performance of the models using these metrics are given in Table 3.

Again, it can be concluded that N-BEATS consistently outperforms the persistence, as its relative error metrics are below, and correlation metric is above 1 respectively. However, more significantly, it can be seen that it systematically outperforms both the BGS statistical approach and neural network CLS model[11] in all metrics over this reduced validation period.

One perhaps surprising result, is the relative poor performance of the CLS model in MAPE when it outperforms the persistence and BGS in RMSE. One possible explanation for this, is that the CLS model used a variation of the RMSE as their loss function during training [12]. In this way, the model learns to minimise this specific metric and, as a result, forecasts generated with this model may have a bias towards it. This illustrates the importance of considering multiple metrics for model performance, and multiple loss functions in the N-BEATS ensemble. It should be noted, however, that it is difficult to definitively conclude the relative performance of these models in any metric using only this reduced validation set. For this, the comparison should be performed over a full solar cycle, as models that are deficient for low solar activity, may perform better during periods of high solar activity.

[11] At the time of submission, the following warning was given by CLS for their solar radio flux service: *"Since May 1st, 2018, F10.7 data on the solar archive are estimated data and not observed data, due to the update interruption of the time series at the NCEI"*. As the CLS forecasts were evaluated against their own archive of observations in order to be consistent with the data that they used during model training, it is feasible that this factor could contribute to decreased performance during this period, although the nature of this estimated data should be investigated further.

Table 4
Relative metric comparison of N-BEATS with operationally available forecasts for 2006–2020. Metrics are scaled against the persistence baseline, and averaged over forecast horizons. The best performing values in each metric are highlighted in bold.

Model	Relative metric				
	MSE	RMSE	MAPE	MAE	R
CLS	**0.345**	**0.585**	0.851	0.828	**1.175**
N-BEATS	0.349	0.589	**0.815**	**0.804**	**1.175**

4.2.2. 2006–2020 validation period

The same analysis was therefore performed over the full validation set for available models, with the metric evolution and relative metrics shown in Fig. 5b and Table 4 respectively. It can be seen that, over a full solar cycle, the performance of N-BEATS is significantly closer to that of CLS in all metrics. As expected, both models outperform the persistence model, but in RMSE related metrics, N-BEATS fractionally underperforms compared to CLS, whereas it outperforms CLS in MAPE related metrics. This supports the hypothesis discussed in Section 4.2.1, that the CLS model may have a bias towards RMSE as it was used as its loss function during training. On the other hand, we use both the MSE and MAPE as loss functions during the ensemble approach (Section 3.4), which works to minimise bias in our final model. This leaves the correlation coefficient, R, as the only independent comparison metric that was not used during training for either approach and, as can be seen in Table 4, the final metric is identical for both models.

This similarity in performance is an encouraging result, given that the CLS approach uses 4 different flux wavelengths during training, whereas N-BEATS learns only from a single one. This suggests that N-BEATS is a more powerful architecture, able to infer hidden patterns in the data that aid in its forecasting capabilities.

Note that, on comparing Table 4 to that obtained over the reduced validation set, Table 3, it can be seen that in general the relative metrics improve when considering a longer timescale with different levels of solar activity. This may be suggestive that the models improve their performance during periods of high solar activity, or indeed rather that the persistence baseline which is used as a scaling factor, has relatively worse performance during these periods. However, the same is not true of the relative R value for N-BEATS, whose performance decreased, which again serves to justify the use of different metrics that capture different aspects of the model performance, but is also an indicator that N-BEATS may have a stronger performance during low solar activity.

To further investigate the strengths and deficiencies of the models over the course of the solar cycle, during different levels of solar activity, we consider the breakdown of RMSE over the validation set. This is shown in Fig. 6, alongside the observed F10.7 data during this period, to illustrate the high level of correlation between the model error and the level of solar activity itself.

From this breakdown of single point model error we can draw two main conclusions. First, that the performance of the two models is fairly comparable throughout the solar cycle, with CLS showing a slight tendency to perform better during increasing activity and for lower horizons, and N-BEATS performing better for lower activity and longer horizons. Second, that all models perform poorly in predicting the peak event in late 2017.

To understand this first point, we consider the Mean Error (ME), or bias, of the models over four year-long periods which are characteristic of increasing, high, decreasing and low levels of solar activity over solar

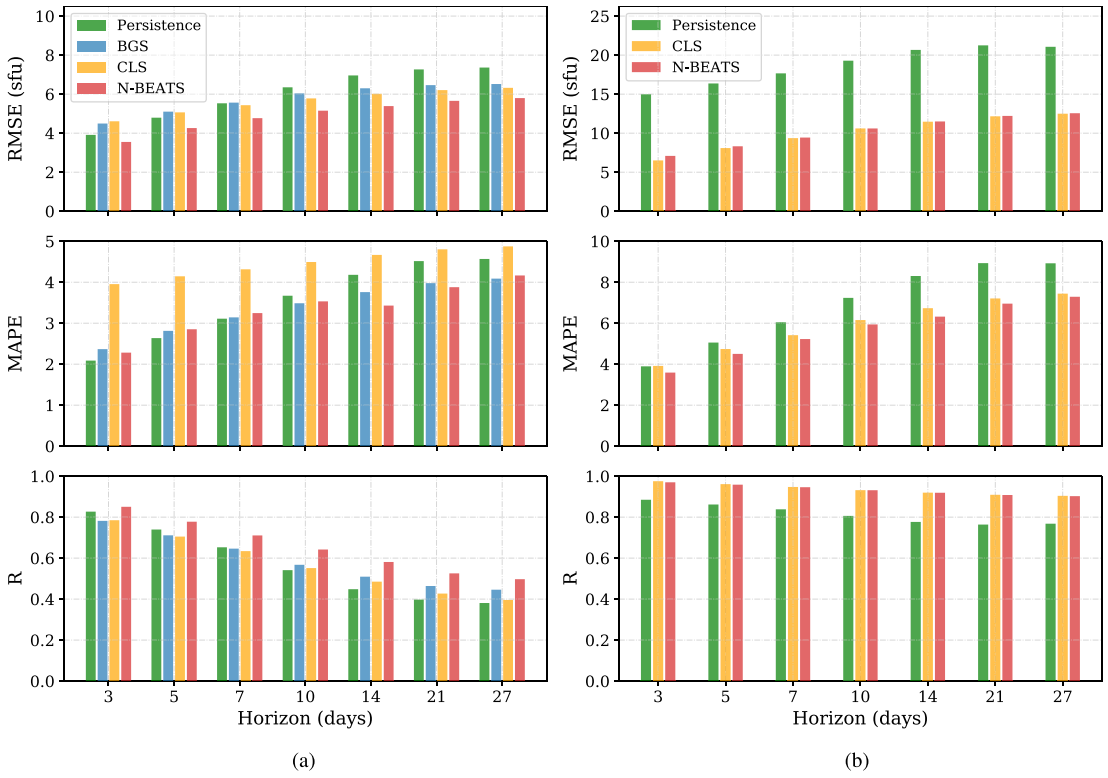

Fig. 5. Evolution of performance metrics with forecast horizon for validation periods covering (a) 2016–2020, (b) 2006–2020. Metrics are Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), Pearson Correlation Coefficient (R).

Table 5
Example periods of different levels of solar activity chosen for comparison of bias between models.

Solar activity level	Date range	
Increasing	01/05/2011	30/04/2012
High	30/09/2013	29/09/2014
Decreasing	31/08/2016	30/08/2017
Low	01/01/2019	31/12/2019

cycle 24. These are listed in Table 5, and were chosen, where possible, to overlap with those used for a similar analysis in [12].

It can be seen from Fig. 7, that CLS is more biased than N-BEATS for low solar activity, which explains its poor performance during these periods, as in Section 4.2.1. This bias is reduced during the periods of higher activity which can again be explained by its sole use of RMSE-like squared metrics, which are weighted more heavily towards higher errors and therefore higher activities.

N-BEATS, on the other hand, has a tendency to have better performance for lower solar activities, and be more biased for high activity. For both increasing and decreasing activity, N-BEATS has a near-zero bias up to a horizon of 6-days before it begins to underpredict, but this deviation is most pronounced during the high period of solar activity.

This systematic underprediction of high fluxes with increasing forecast horizon by N-BEATS can be better seen in Fig. 8. For low horizons, the R value between the observed and predicted fluxes is close to 1. However, as the forecast horizon increases, the model tends to underestimate high values of the F10.7, resulting in a negative bias.

This damping of high activity by the model then also leads us to the second point, the peak event in 2017. This corresponds to an intense storm period that occurred during September 2017 which produced the largest flares during Solar Cycle 24 [38]. This included 4 X-class flares, the highest flare class, which have the ability to disturb satellite trajectories, and are therefore important to capture for orbital prediction [39].

It can be seen from Fig. 6, that N-BEATS is unable to capture this event. This is a result of the overconfidence of deep neural networks with out-of-distribution data, by which rare events can be erroneously predicted as in-distribution values with high confidence [15]. The CLS neural network model appears to suffer similarly, although for CLS, the error evaluation here is performed against it's own provided archive of observations, which undergoes an anomaly screening [12]. The period of observations covering this event has been tagged as "flare corrected", and therefore may not provide a direct comparison to the N-BEATS forecast. The fact that, contrary to CLS, we are not training from or evaluating against systematically cleaned data, may also explain the apparent higher bias of N-BEATS. However, we chose to not to use cleaned data as we wanted to allow for flare capture by our model. This will be further investigated in future work.

4.3. Comparison of uncertainty estimation

In this section, we present an analysis of model performance in terms of uncertainty estimation. Of the available models for comparison with N-BEATS, only CLS provides a measure of prediction uncertainty, and we are therefore able to use the full validation set (20/10/2006 to 01/10/2020) for this analysis. Both models employ fundamentally different approaches to uncertainty quantification. For N-BEATS, we obtain the forecast uncertainty directly from the deep ensemble as the standard deviation over individual model runs, as described in Section 3.4. This differs to the approach used by CLS, for which the uncertainty estimation is not inherent to the model, but assigned after prediction using a linear fit between past model RMSE and solar flux [12].

Fig. 6 shows the 162-day-averaged level of forecast uncertainty for each model, 1σ for N-BEATS and RMSE for CLS, for a 10-day forecast horizon. From this, it can be seen that the N-BEATS ensemble approach is able to correctly characterise the shape of the forecast uncertainty, exhibiting the same high level of correlation with solar

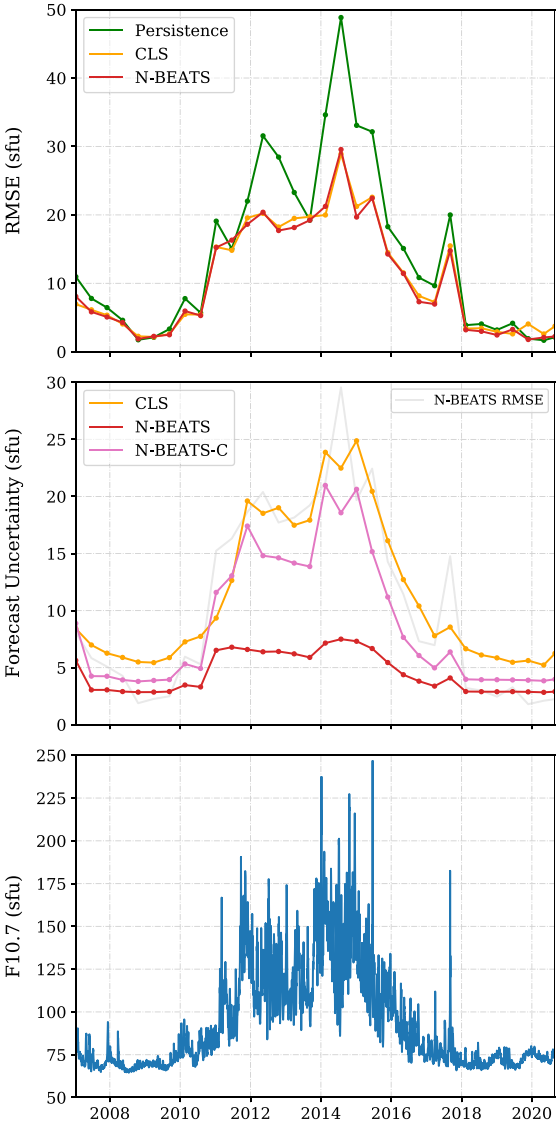

Fig. 6. Breakdown of model performance and forecast uncertainty over the validation period to illustrate correlation with solar activity. *Top*: 162-day average of model RMSE for a forecast horizon of 10 days. *Middle*: 162-day average of model forecast uncertainty for a forecast horizon of 10 days, with RMSE of N-BEATS, from the top plot, shown in grey for comparison. N-BEATS forecast uncertainty is the 1σ standard deviation over the ensemble; CLS uncertainty is an RMSE obtained by a linear fit of past model error with solar activity [12]. *Bottom*: Observed F10.7 over the validation period.

activity and single point forecast RMSE as the CLS model. The main distinction between the models is that the N-BEATS estimation is much narrower than that of CLS, and it should be noted that the same relative behaviour is true of all horizons. In and of itself, this observation cannot be used to form conclusions about the quality of uncertainty estimation in either approach: a narrower uncertainty window is preferable, provided it is not underestimating the uncertainty.

To quantify the relative performance, we consider the proportion of true observed values, y, that fall within the estimated uncertainty range, $\hat{y} \pm \sigma$, of each model. In order to perform a comparison, we then measure this performance against the expected fraction for a given confidence level, c. For this, we make two main assumptions. First, we assume that the ensemble average is normally distributed according to the central limit theorem [40]. Second, given the low bias of the CLS model, as demonstrated in Fig. 7, we assume that it is an unbiased predictor, and therefore the provided RMSE can be approximated as a standard deviation. Under these assumptions, we can

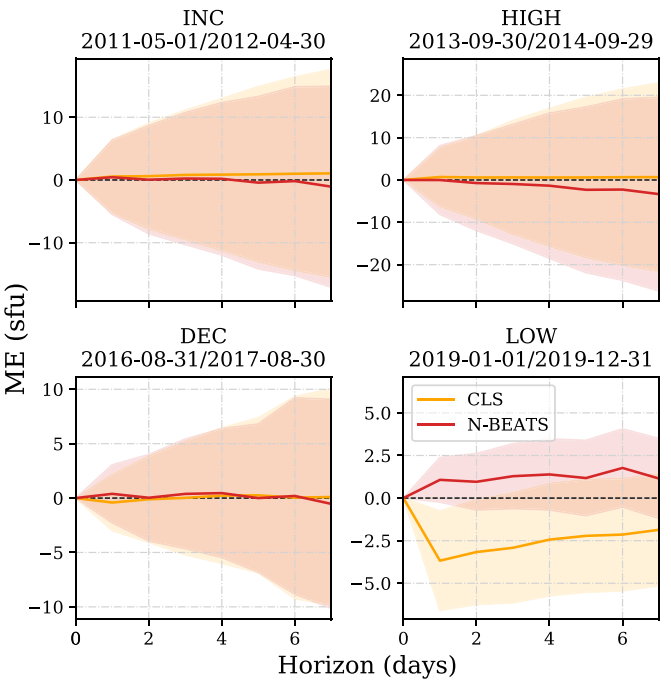

Fig. 7. Mean Error (or bias) (predicted - observed) and 1σ standard deviation of the bias for N-BEATS and CLS forecasts over a 10-day forecast horizon during four periods of different solar activity (increasing, high, decreasing, low). The black dashed line represents the ideal 0 bias, with positive and negative values of the bias representing a tendency of the model to over and under-predict respectively.

obtain the uncertainty interval for each model for a given confidence level using the probit function [40], which acts as a multiplier to the model standard deviation. The performance of each model for different confidence levels (a reliability diagram), is shown in Fig. 9.

From Fig. 9, it can be seen that the N-BEATS uncertainty estimation is overconfident, falling below the ideal black dashed line, and implying that the spread of ensemble predictions is too narrow. However, this result is not unexpected, as MSE-based deep ensembles have been found to routinely yield overoptimistic uncertainty estimates in the field of deep learning [15].

To account for this, we follow an approach developed for ensemble forecasting for climate predictions [40], which uses a correction, or calibration factor, to increase the width of the uncertainty. Following the principle that RMSE should be equal to the standard deviation for an unbiased predictor, this correction factor, γ, is derived as the ratio between the RMSE and ensemble standard deviation during the model training period:

$$\gamma = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_t - y_t)^2}{\sum_{t=1}^{n}\sigma_t^2}}, \tag{7}$$

where $t \in \{1, \ldots, n\}$ are the time points for which predictions are made over the training set. If this ratio does not vary significantly with time, γ may be taken as the temporal average over the entire training set. However, we found this not to be the case, with the ratio following the same seasonal trend as the solar flux. As such, we expanded this method in order to obtain a correction factor as a function of predicted solar flux, $\gamma(\hat{y})$. For this, γ factors were calculated over 365-day sliding windows over the training set, and a linear relation found with the mean window flux.

The resulting calibrated model, denoted as N-BEATS-C, comprises the same single point predictions (ensemble mean) as the original N-BEATS model, but with a corrected standard deviation, $\sigma_c = \gamma(\hat{y})\sigma$. This correction factor was found to work well, with N-BEATS-C providing significantly improved uncertainty estimation over the original

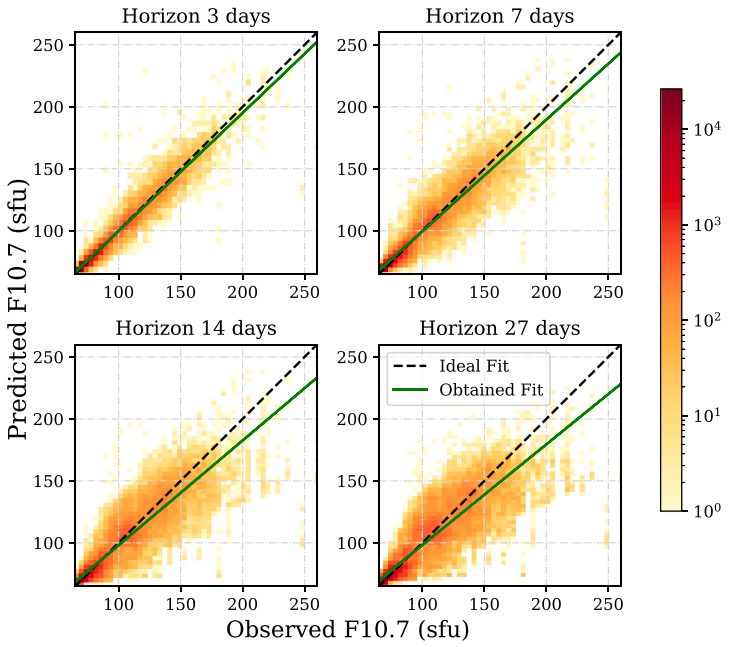

Fig. 8. Occurrence map of N-BEATS predicted values of the F10.7 with observed values over different forecast horizons for 2006–2020. The black dashed line represents the ideal $R = 1$ correlation, and the green line the fit obtained through linear regression.

N-BEATS (Fig. 9). From this figure, it can be seen that not only does N-BEATS-C no longer significantly underpredict the uncertainty, but also that it lies closer to the ideal than CLS, which overpredicts the uncertainty.

This can be explained from Fig. 6, in which the forecast uncertainties are compared to the single point model RMSE. For an uncertainty estimate to be realistic, it should contain the model RMSE (for legibility, here only that of N-BEATS is shown as it is sufficiently similar to CLS to enable this comparison). It can be seen that the forecast uncertainty of CLS is wider than that of N-BEATS-C, and that although it better encompasses the RMSE for some periods of high solar activity, it overestimates the uncertainty during all periods of low activity. On the other hand, N-BEATS-C has a better estimation of the uncertainty during low activity, and, as can be seen from Fig. 9, therefore better characterises the forecast uncertainty on average, with narrower windows that do not significantly underestimate it.

One caveat to this analysis concerns the validity of the central limit theorem. Each ensemble is formed of 90 members (Section 3.4), whose predictions are not necessarily independent (indeed time series in general often suffer from autocorrelation), nor identically distributed. However, although the reliability diagram shows some non-linearity, suggesting that the data is not normally distributed, there is no major deviation from the linear ideal, serving to provide some further justification for the approach. An alternative approach to correcting for overoptimistic estimates of deep ensembles, by updating the loss functions used during training to those that are able to also capture the quality of the predictive uncertainty of the model (the variance as well as the mean [15]), which do not require these assumptions, will be investigated in future work.

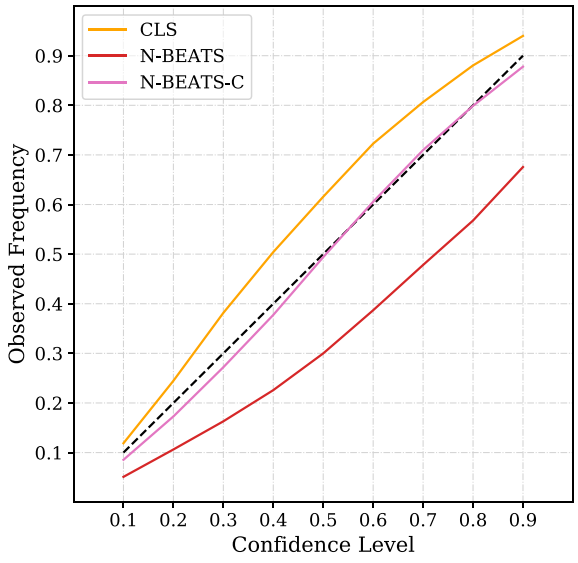

Fig. 9. Reliability diagram for CLS, N-BEATS and N-BEATS-C (N-BEATS with calibrated uncertainty) over a 10-day forecast horizon. The black dashed line represents the ideal, for which the observed frequency (proportion of future observations falling within the forecast uncertainty window) is equal to the given confidence level.

5. Conclusions and future work

This paper presents the use of the novel N-BEATS deep residual neural network for the daily prediction of the F10.7 solar proxy. This pure deep learning approach, which has provided a significant advancement in the field of time series forecasting within the last year, was found to be effective in this task up to a forecast horizon of 27-days, without the need for any specific expert knowledge of the data or feature engineering.

Forecasts generated using this deep, univariate approach were compared to a persistence baseline and two operationally available forecasts: BGS (a statistical approach) and CLS (a shallow neural network approach based on 4 flux wavelengths). For this comparison, several metrics were considered in order to obtain a comprehensive and unbiased evaluation of model performance. It was found that the N-BEATS model systematically outperformed the baseline and statistical approaches, and achieved an improved or similar performance to CLS in all metrics, despite only learning from a single flux wavelength. Therefore, not only was N-BEATS found to be a more powerful architecture for predicting the F10.7, in requiring less data to achieve the same level of performance, and consequently reducing the potential sources

of uncertainty, but also that the use of different loss functions in the N-BEATS approach lead to reduced training biases.

To capture the uncertainty in the forecasting, the N-BEATS model was additionally extended in this work to provide integrated uncertainty quantification using deep ensembles. Although initially overoptimistic, as is typical for deep ensembles of this type, it was found that applying a correction factor resulted in a superior characterisation of the forecast uncertainty when compared to that of CLS, especially during periods of low solar activity.

In future work, we will work to augment our model with improved uncertainty estimators to further mitigate the propensity of deep ensembles to produce overoptimistic uncertainty estimates, and expand the approach to produce probabilistic forecasts. We will also consider further improvements in the accuracy of the model by including auxiliary variables, such as additional flux wavelengths, which may also aid in correcting the tendency of the model to underpredict the flux at high solar activity. In a further step, this approach could then be used to extend the forecasting to these other variables, for example the F30, in a multivariate approach.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research is supported by the EU H2020 MSCA ITN Stardust-R, grant agreement 813644. The authors would to greatly acknowledge Collecte Localisation Satellites (CLS), the British Geological Survey (BGS) and the National Geophysical Data Center (NOAA), as well as the ESA Space Situational Awareness Programme and Dominion Radio Astrophysical Observatory (Penticton, Canada) for providing the data necessary to carry out this work. Finally, the authors would like to thank the two anonymous reviewers for their insightful comments and suggestions.

References

[1] C.D. Bussy-Virat, A.J. Ridley, J.W. Getchius, Effects of uncertainties in the atmospheric density on the probability of collision between space objects, Space Weather 16 (5) (2018) 519–537, http://dx.doi.org/10.1029/2017SW001705.

[2] B. Bastida Virgili, S. Lemmens, E. Stevenson, B. Reihs, Statistical comparison of ISO recommended thermosphere models and space weather proxy forecasting on re-entry predictions, in: Proceedings of the International Astronautical Congress, IAC, 2017, pp. 3554–3561.

[3] A. Vourlidas, S. Bruinsma, EUV irradiance inputs to thermospheric density models: Open issues and path forward, Space Weather 16 (1) (2018) 5–15, http://dx.doi.org/10.1002/2017SW001725.

[4] K.F. Tapping, The 10.7 cm solar radio flux ($F_{10.7}$), Space Weather 11 (7) (2013) 394–406, http://dx.doi.org/10.1002/swe.20064.

[5] D.A. Vallado, D. Finkleman, A critical assessment of satellite drag and atmospheric density modeling, Acta Astronaut. 95 (2014) 141–165, http://dx.doi.org/10.1016/j.actaastro.2013.10.005.

[6] R. Mugellesi-dow, D.J. Kerridge, T.D.G. Clark, A.W.P. Thompson, SOLMAG: an operational system for prediction of solar and geomagnetic activity indices, in: Proceedings of the First European Conference on Space Debris, 1993, pp. 373–376.

[7] W.K. Tobiska, S.D. Bouwer, B.R. Bowman, The development of new solar indices for use in thermospheric density modeling, J. Atmos. Sol.-Terr. Phys. 70 (5) (2008) 803–819, http://dx.doi.org/10.1016/j.jastp.2007.11.001.

[8] H.P. Warren, J.T. Emmert, N.A. Crump, Linear forecasting of the F10.7 proxy for solar activity, Space Weather 15 (8) (2017) 1039–1051, http://dx.doi.org/10.1002/2017SW001637.

[9] Z. Wang, Q. Hu, Q. Zhong, Y. Wang, Linear multistep $F10.7$ forecasting based on task correlation and heteroscedasticity, Adv. Earth Space Sci. 5 (12) (2018) 863–874, http://dx.doi.org/10.1029/2018ea000393.

[10] Z. Du, Forecasting the daily 10.7 cm solar radio flux using an autoregressive model, Sol. Phys. 295 (9) (2020) 1–23.

[11] C. Huang, D. Liu, J. Wang, Forecast daily indices of solar activity, $F10.7$, using support vector regression method, Res. Astron. Astrophys. 9 (6) (2009) 694–702, http://dx.doi.org/10.1088/1674-4527/9/6/008.

[12] P. Yaya, L. Hecker, T.D.d. Wit, C.L. Fèvre, S. Bruinsma, Solar radio proxies for improved satellite orbit prediction, J. Space Weather Space Clim. 7 (2017) A35, http://dx.doi.org/10.1051/swsc/2017032.

[13] E. Camporeale, The challenge of machine learning in space weather: Nowcasting and forecasting, Space Weather 17 (8) (2019) 1166–1207, http://dx.doi.org/10.1029/2018SW002061.

[14] B.N. Oreshkin, D. Carpov, N. Chapados, Y. Bengio, N-BEATS: Neural basis expansion analysis for interpretable time series forecasting, in: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, OpenReview.net, 2020.

[15] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30, Curran Associates, Inc., 2017, pp. 6402–6413.

[16] S. Fort, H. Hu, B. Lakshminarayanan, Deep ensembles: A loss landscape perspective, 2019, arXiv preprint arXiv:1912.02757.

[17] P. Pinson, Estimation of the Uncertainty in Wind Power Forecasting (Ph.D. thesis), École Nationale Supérieure des Mines de Paris, 2006.

[18] T. Januschowski, J. Gasthaus, Y. Wang, D. Salinas, V. Flunkert, M. Bohlke-Schneider, L. Callot, Criteria for classifying forecasting methods, M4 Competition, Int. J. Forecast. 36 (1) (2020) 167–177, http://dx.doi.org/10.1016/j.ijforecast.2019.05.008.

[19] G.E. Box, G.M. Jenkins, G. Reinsel, Time series analysis: forecasting and control Holden-day San Francisco, in: BoxTime Series Analysis: Forecasting and Control Holden Day1970, 1970.

[20] R.J. Hyndman, A.B. Koehler, R.D. Snyder, S. Grose, A state space framework for automatic forecasting using exponential smoothing methods, Int. J. Forecast. 18 (3) (2002) 439–454.

[21] X. Qiu, L. Zhang, Y. Ren, P.N. Suganthan, G. Amaratunga, Ensemble deep learning for regression and time series forecasting, in: 2014 IEEE Symposium on Computational Intelligence in Ensemble Learning (CIEL), IEEE, 2014, pp. 1–6.

[22] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control Signals Systems 2 (4) (1989) 303–314.

[23] S. Gugger, J. Howard, Deep Learning for Coders with Fastai and PyTorch: AI Applications Without a PhD, O'Reilly Media, Incorporated, 2020.

[24] B. Lim, S. Zohren, Time series forecasting with deep learning: A survey, 2020, arXiv preprint arXiv:2004.13408.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.

[26] B. Lim, S.O. Arik, N. Loeff, T. Pfister, Temporal fusion transformers for interpretable multi-horizon time series forecasting, 2019, arXiv:1912.09363 [cs, stat] arXiv:1912.09363.

[27] S. Makridakis, E. Spiliotis, V. Assimakopoulos, The M4 competition: Results, findings, conclusion and way forward, Int. J. Forecast. 34 (4) (2018) 802–808.

[28] R.J. Licata, W.K. Tobiska, P.M. Mehta, Benchmarking forecasting models for space weather drivers, Space Weather 18 (10) (2020) http://dx.doi.org/10.1029/2020SW002496.

[29] Y. Tan, Q. Hu, Z. Wang, Q. Zhong, Geomagnetic index Kp forecasting with LSTM, Space Weather 16 (4) (2018) 406–416, http://dx.doi.org/10.1002/2017SW001764.

[30] S. Bhattacharjee, R. Alshehhi, D.B. Dhuri, S.M. Hanasoge, Supervised convolutional neural networks for classification of flaring and nonflaring active regions using line-of-sight magnetograms, Astrophys. J. 898 (2) (2020) 98, http://dx.doi.org/10.3847/1538-4357/ab9c29.

[31] S. Dey, O. Fuentes, Predicting solar X-ray flux using deep learning techniques, in: 2020 International Joint Conference on Neural Networks (IJCNN), 2020, pp. 1–7.

[32] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[33] H. Rubin-Falcone, I. Fox, J. Wiens, Deep residual time-series forecasting: Application to blood glucose prediction, in: K. Bach, R.C. Bunescu, C. Marling, N. Wiratunga (Eds.), Proceedings of the 5th International Workshop on Knowledge Discovery in Healthcare Data Co-Located with 24th European Conference on Artificial Intelligence, KDH@ECAI 2020, Santiago de Compostela, Spain & Virtually, August 29-30, 2020, in: CEUR Workshop Proceedings, vol. 2675, CEUR-WS.org, 2020, pp. 105–109.

[34] L. Biewald, Experiment tracking with weights and biases, 2020, Software available from https://www.wandb.com/.

[35] M.W. Liemohn, J.P. McCollough, V.K. Jordanova, C.M. Ngwira, S.K. Morley, C. Cid, W.K. Tobiska, P. Wintoft, N.Y. Ganushkina, D.T. Welling, S. Bingham, M.A. Balikhin, H.J. Opgenoorth, M.A. Engel, R.S. Weigel, H.J. Singer, D. Buresova, S. Bruinsma, I.S. Zhelavskaya, Y.Y. Shprits, R. Vasile, Model evaluation guidelines for geomagnetic index predictions, Space Weather 16 (12) (2018) 2079–2102, http://dx.doi.org/10.1029/2018SW002067.

[36] S.A. Murray, The importance of ensemble techniques for operational space weather forecasting, Space Weather 16 (7) (2018) 777–783, http://dx.doi.org/10.1029/2018SW001861.

[37] J.A. Guerra, S.A. Murray, E. Doornbos, The use of ensembles in space weather forecasting, Space Weather 18 (2) (2020) http://dx.doi.org/10.1029/2020SW002443.

[38] P.C. Chamberlin, T.N. Woods, L. Didkovsky, F.G. Eparvier, A.R. Jones, J.L. Machol, J.P. Mason, M. Snow, E.M.B. Thiemann, R.A. Viereck, D.L. Woodraska, Solar ultraviolet irradiance observations of the solar flares during the intense september 2017 storm period, Space Weather 16 (10) (2018) 1470–1487, http://dx.doi.org/10.1029/2018SW001866.

[39] F. Deleflie, K. Doerksen, C. Briand, M.A. Sammuneh, L. Sagnières, Atmospheric density variations and orbit perturbations in relation to isolated solar X-flare events, in: EGU General Assembly Conference Abstracts, in: EGU General Assembly Conference Abstracts, 2019, p. 15338.

[40] E. Strobach, G. Bel, Quantifying the uncertainties in an ensemble of decadal climate predictions, J. Geophys. Res.: Atmos. 122 (24) (2017) 13,191–13,200, http://dx.doi.org/10.1002/2017JD027249.