AGU100 ADVANCING EARTH AND SPACE SCIENCE

Space Weather

RESEARCH ARTICLE
10.1002/2017SW001764

Key Points:
- Long short-term memory (LSTM) is introduced in *Kp* forecasting
- The samples are divided into storm and nonstorm, and then two models are trained separately
- Comparing six models and good performance is observed from the proposed model

Supporting Information:
- Supporting Information S1
- Data Set S1
- Data Set S2
- Data Set S3
- Data Set S4
- Data Set S5

Correspondence to:
Q. Hu,
huqinghua@tju.edu.cn

Citation:
Tan, Y., Hu, Q., Wang, Z., & Zhong, Q. (2018). Geomagnetic index *Kp* forecasting with LSTM. *Space Weather*, *16*, 406–416. https://doi.org/10.1002/2017SW001764

Received 13 NOV 2017
Accepted 26 MAR 2018
Accepted article online 6 APR 2018
Published online 21 APR 2018

Geomagnetic Index *Kp* Forecasting With LSTM

Yao Tan[1], Qinghua Hu[1], Zhen Wang[1], and Qiuzhen Zhong[2,3]

[1]School of Computer Science and Technology, Tianjin University, Tianjin, China, [2]National Space Science Center, Chinese Academy of Sciences, Beijing, China, [3]School of Astronomy and Space Sciences, University of Chinese Academy of Sciences, Beijing, China

Abstract Through making full use of the solar wind and interplanetary magnetic field data accumulated by ACE satellites we improve the prediction accuracy of the *Kp* geomagnetic index and accurately predict the occurrence of geomagnetic storms ($Kp \geq 5$). Specially, we use long short-term memory to train the *Kp* forecast model described in this study. Based on the large-scale data, we build the *Kp* forecasting model with solar wind, interplanetary magnetic field parameters, and the historical *Kp* value as input. In this study, we first analyze the distribution of *Kp* and the effect of the data imbalance on the prediction of geomagnetic storms. Second, we analyze the correlation between the different input parameters and *Kp*. Thus, the input parameters of the model are selected by the results of the correlation. We consider two types of forecasting: one is the overall *Kp* forecasting and the other is the geomagnetic storm ($Kp \geq 5$) forecasting. Hence, we design an integrated model which is then compared with other models. Some evaluation parameters are introduced: the root-mean-square error, the mean-absolute error, and the correlation coefficient, as well as the measurement of geomagnetic storms ($Kp \geq 5$) *F*1. The root-mean-square error and mean-absolute error of our model are 0.4765 and 0.6382, respectively. The experimental results show that the proposed model with long short-term memory improve the *Kp* forecasting.

Plain Language Summary The work is about *Kp* forecasting. *Kp* is one of the geomagnetic indexes that describes the intensity of the geomagnetic disturbance for a certain period of time. Our work uses the deep learning to train the forecasting model, whose forecasting error is smaller than other existing methods. The model is trained on many related parameters for more than 10 years.

1. Introduction

The magnetosphere is a complex and vast environment whose scale varies in orders of magnitude (Newell et al., 2007). Its shape is determined by the magnitude of the Earth's internal magnetic field and upstream solar wind conditions. The formation process of the geomagnetic activity is complex. Geomagnetic storms have a large impact on the Earth's space environment, which can affect power grids, satellite drag, and navigation systems (Ayala Solares et al., 2016). Hence, this can influence the electricity and GPS on Earth. Thus, it is necessary to accurately predict *Kp* and geomagnetic storms. A geomagnetic index describes the intensity of the geomagnetic disturbance for a certain period of time. There are sets of indices for describing the geomagnetic field. Some indices describe the overall level of geomagnetic activity, such as *Kp* or *Ap*, and some describe a particular type or specific area of geomagnetic activity, such as the *Dst* index. The *K* index is used to describe the geomagnetic disturbance at single-ground magnetic field observatories in 3 hr. The *Kp* index is calculated by the weighted average of the *K* indices of 13 geomagnetic stations around the world. There are eight values for any given day that are divided into 28 levels as follows: 0_0, 0_+, $1-$, 1_0, 1_+, …,$9-$, 9_0. The *Kp* index was first introduced by Bartels (1949). The Space Weather Prediction Center classifies geomagnetic activity into four levels: quiet to unsettled ($Kp < 4$), active ($Kp = 4$), minor storm ($Kp = 5$), and major to severe storm ($Kp > 5$). The *Kp* index is widely used as an input to other models related to the magnetosphere (Ji et al., 2013).

There are two kinds of models for *Kp* forecasting and nowcasting. We discuss forecast models in this research paper, and there are various studies related to *Kp* forecasting. The artificial neural network (NN) is the most popular algorithm for *Kp* forecasting. In 1998, Costello proposed a prediction model based on an NN (Costello, 1998), which is also used in practical operations, and its forecast results are on the National Oceanic and Atmospheric Administration website. Later, subsequent forecast models exploited different input parameters such as proton density and solar wind velocity (Boberg et al., 2000; Johnson & Wing, 2005) and *Kp* was as one of the input parameters in the Wing model, for the first time. The Bala model

discussed the forecasting of three indexes: *Kp*, Dst, and *AE* at 1 hr, 3 hr, and 6 hr ahead of time (Bala & Reiff, 2012). These models were all based on NN. In addition, the support vector machine (SVM) was also leveraged to build the model. Ji et al. compared the performance of SVM and NN in forecasting *Kp* (Ji et al., 2013). It was found that the root-mean-square error (RMSE) of NN was smaller than that of SVM. Additionally, they proposed using the contingency table to evaluate the forecasting accuracy of geomagnetic storms. The contingency table is similar to the confusion matrix whose rows and columns all represent predicted *Kp* and observed *Kp*, respectively. There was a *Kp* model using nonlinear autoregressive with exogenous inputs models, a powerful tool of nonlinear system identification (Ayala Solares et al., 2016). The best 3 hr in advance result for RMSE is 0.7582. Finally, Wang proposed a probabilistic model combining solar wind and interplanetary magnetic field (IMF) parameters (Wang et al., 2015). They analyzed the probability of the occurrence of several levels of geomagnetic storms. This was a new idea from the point of view of probability.

As we build the 3 hr advance model, we introduce two 3-h-related models, the Boberg and Bala models, which are compared with ours in section 4. The Boberg model used a multilayer feed forward network, which included one hidden layer whose input was sequences of Bz, *D*, and *V*. In addition, the activation function was a hyperbolic tangent function. They calculated the RMSE and correlation of training, validation, and testing. Then Bala compared the *Kp* forecasting models by changing different input parameters. There were significant differences in the RMSE and correlation among these models. Bala also discussed the impact of the prediction time on forecasting performance. They found that the longer the prediction time was, the larger the RMSE was. From their experimental results we just select the best *Kp* forecasting model whose input are a Newell function and a viscous term.

The above models consider the *Kp* forecasting error and they do not consider geomagnetic storm forecasting directly in their forecast models. We propose a *Kp* forecast model based on long short-term memory (LSTM). LSTM is developed from recurrent neural networks (RNNs) (Graves, 2012). LSTM can hold long-term memory and is helpful for learning sequential data. It can find complex nonlinear relationships in the data set. Hence, LSTM has been applied to many fields, that is, machine translation (Sutskever et al., 2014), image caption (Vinyals et al., 2015), and other interesting applications.

In this work, we design a model structure that first predicts the inputs to storm and nonstorm, and then use two separate submodels to forecasting the *Kp* value. One submodel predicts *Kp* when geomagnetic storm occurs and the other predicts *Kp* when the storm does not occur. Thus, combined framework is developed of a classification model and two regression submodels. Additionally, it not only leads to a smaller predicting error but also improves the geomagnetic storm forecasting performance.

This paper is organized as follows: Section 2 explains data analysis and processing that contains three parts: missing value processing, correlation analysis, and data imbalance. In section 3 we introduce the proposed forecasting model, the training algorithm LSTM, and the model structure in detail. Several comparison experiments are provided in section 4. Section 5 provides a conclusion to the research.

2. Data Analysis and Processing

Even though different input parameters are used in different models, these parameters are all correlated to the geomagnetic activity. We consider 11 parameters. IMF parameters include Bx, By, Bz, and *B*. Solar wind parameters include *V* and *D*, the related terms are BI = $10^{-4}V^2 + 11.7B\sin^3(\theta/2)$ (Boyle et al., 1997), $\frac{d\varphi_{MP}}{dt} = V^{\frac{4}{3}} B^{\frac{2}{3}} \sin^{\frac{8}{3}}(\theta/2)$ (Newell et al., 2008), vis $= D^{\frac{1}{2}}V^2$, *Dp*=1.6726E−6 ×DV^2, and the other parameter is *Kp*. The related terms can be calculated based on IMF and solar wind parameters.

Candidate input parameters of the forecast model are historical *Kp*, solar wind, IMF parameters, and related terms. The output of the model is *Kp* at the next timestamp. We collect the data set from 2000 to 2014, which contains the *Kp* value from GFZ German Research Centre for Geosciences and solar wind, as well as IMF parameters from ACE. Original sampling rates of solar wind and IMF parameters are 1 hr, and we compute the average in 3 hr. We build a forecast model for 3 hr in advance with solar wind and IMF parameters with sampling rate of 3 hr. To accomplish the training and test process, we divide the whole data set into a training data set (2000.8–2011.10), a validation data set (2012.11–2013.11), and a test data set (2013.12–2014.9). The validation data set is used to set the hyperparameters of the model. Each input parameter is essentially a time

Table 1
Eleven Candidate Input Parameters in This Research

Parameter	Symbol/Calculating method
The x component of IMF	Bx
The y component of IMF	By
The z component of IMF	Bz
Total magnitude of IMF	B
Flow speed	V
Proton density	D
Boyle index	$\mathrm{BI} = 10^{-4}V^2 + 11.7B\sin^3(\theta/2)$
Newell function	$\frac{d\varphi_{MP}}{dt} = V^{\frac{4}{3}}B^{\frac{2}{3}}\sin^{\frac{8}{3}}(\theta/2)$
Viscous function	$\mathrm{vis} = D^{\frac{1}{2}}V^2$
Dynamic pressure	Dp=1.6726e − 6 $\times DV^2$
Historical *Kp*	*Kp*

series. Thus, an algorithm suitable for time series prediction can be used to build the forecast model (Tables 1 and 2).

2.1. Data Processing

The original *Kp* value is indicated with the symbol "+", "−", "o" behind the number. We change the *Kp* value to real number where "*n*−" changes to $n - 0.33$, "no" changes to n and "*n*+" changes to $n + 0.33$. We identify some missing IMF and solar wind data values. Then, we use the interpolation method (Pang, 2012) to deal with the missing values and restore the original data trend.

Due to the strong correlation between adjacent moments, we design the interpolation method:

$$x_t = \begin{cases} x_{t-1}, & \text{if } x_{t-1} \text{ is not missing and } x_{t+1} \text{ is missing} \\ x_{t+1}, & \text{if } x_{t+1} \text{ is not missing and } x_{t-1} \text{ is missing} \\ \dfrac{(x_{t-1}+x_{t+1})}{2}, & \text{if } x_{t-1} \text{ is not missing and } x_{t+1} \text{ is not missing} \\ x', & x' \sim N(\overline{x}, \sigma^2), \text{if } x_{t-1} \text{ and } x_{t+1} \text{ is missing} \end{cases} \quad (1)$$

x_t represents the value of one observation at t moment. The parameter $\overline{x}$ is the mean.

2.2. Correlation Between the Parameters and *Kp*

To select the necessary and sufficient inputs of the model from the candidate parameters, we calculate the correlation between the candidate parameters and *Kp* at different lag moments (t' = 0,1,2,…10). The Pearson coefficient is introduced here:

$$R = \frac{\sum_{k=1}^{N}(X^k - \overline{X})(Y^k - Y)}{\sqrt{\sum_{k=1}^{N}(X^k - X)^2}\sqrt{\sum_{k=1}^{N}(Y^k - \overline{Y})^2}} \quad (2)$$

X and Y are two parameters, X^k is the kth component of the X vector and $\overline{X}$ is the average of the X vector. We consider 11 parameters mentioned above. They contain Bx, By, Bz, B, V, D, and BI, $\frac{d\varphi_{MP}}{dt}$, vis, Dp, and *Kp*.

From Figure 1, we can draw two conclusions. First, at the same lag moment, different parameters have different correlations with *Kp*. *Kp* itself, BI, V, and B are higher correlated among these parameters. Second, the correlation drops when the lag time increases. Considering the above conclusions and the usage of the input parameters in other *Kp* forecast models such as *Dp*, which corresponds with the red line, the input parameters in our paper are *Dp*, *Kp*, and BI. We use the *Kp* nowcast data provided by the official GFZ website.

2.3. Data Imbalance

We first analyze the distribution of *Kp* values. Figure 2 displays the distribution of *Kp*. We see that most samples are distributed in the interval of [0, 5). Fewer *Kp* values are larger than 5. This demonstrates that severe geomagnetic storms have a small probability. Then, in 32,886 training samples, only 1,063 samples output *Kp*

Table 2
The Meaning of the Output of Each Part of a Memory Cell

Mathematical symbol	The meaning
$h^{(t)}$	The output of a memory cell at moment t
$S^{(t)}$	The output of the internal state in a memory cell at moment t
$i^{(t)}$	The output of the input gate in a memory cell at moment t
$f^{(t)}$	The output of the forget gate in a memory cell at moment t
W_i, W_f, W_o, W, W_{hi}, W_{hf}, W_{ho}	The weight of the edges

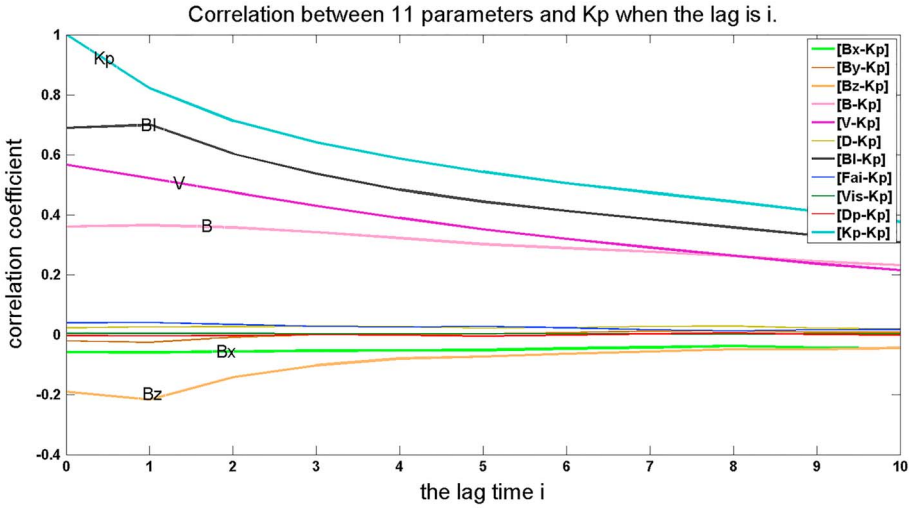

Figure 1. Correlation between 11 parameters and *Kp* at different lag moments.

larger than 5. Thus, the ratio of storm and nonstorm samples is 1:30. The ratio reflects the data imbalance. The general forecast models may achieve a small error in terms of the overall *Kp* forecasting, but the imbalance may lead to a low accuracy in geomagnetic storm forecasting.

There are some methods for dealing with the data imbalance, such as random sampling, cost-sensitive learning, and so on. Random undersampling (Estabrooks et al., 2004) is one of the methods used to mitigate the data imbalance. Though we have conducted some comparisonal experiments with other methods, we finally determine that random undersampling offers better improvements to the model.

If we just use the undersampling method to deal with the imbalance, the error of *Kp* forecasting will increase relative to the previous models. To reduce the impact of imbalance, we combine the classification model first and two regression submodels. We analyze the characteristics of the geomagnetic storm and nongeomagnetic storm data by clustering the two classes of data.

Considering the correlation result in Figure 1, when the lag time $i > 6$, the correlation coefficient of $Dp < 0.1$. In addition, the input length should be determined. After conducting several experiments by comparing the model performance of $T = 4, 5, 6, 7, 8$, and 9, finally, we choose the input length $T = 6$, which is slightly better than the other values. We have confirmed the input parameters of the model, such that the input of the LSTM model is made up of a time series of Bl, *Kp*, and Dp, with length = 6. It can be regarded as having 18 features. We divide the samples into three groups separately for storm and nonstorm samples using a Fuzzy C-means

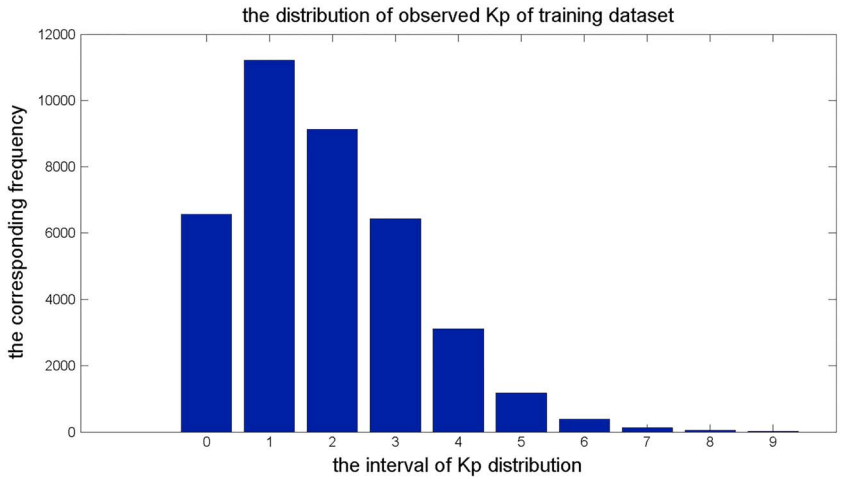

Figure 2. Distribution of *Kp* of training data set.

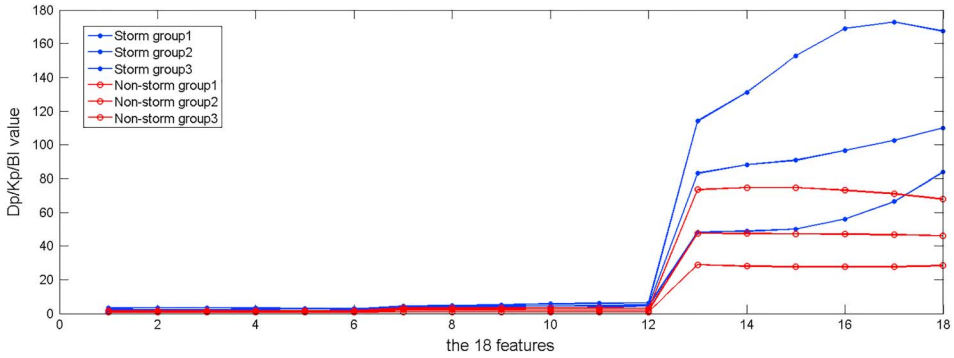

Figure 3. Feature curve of the center of three groups of storm and nonstorm data.

(FCM) algorithm. FCM is one of the cluster algorithms and when setting the number of clustering, we can get the cluster centers of each group. For storm and nonstorm data sets we use FCM respectively. The 18 features of each sample in a data set are the input, then we can get the three clusters for each data set. Then there are three centers for storm data set and three centers for nonstorm data set. Three centers of storm and nonstorm samples can reflect the overall characteristics. There are six curves representing six centers in Figure 3. The *x* label is the features and the *y* label is the *Dp*/*Kp*/BI value of centers. The blue and solid circle curves represent the storm class, and the red and hollow circle curves are the nonstorm class. Features 1 to 6 correspond to Dp_{t-1}, Dp_{t-2}, …, Dp_{t-6}. Features 7 to 12 correspond to Kp_{t-1}, Kp_{t-2}, …, Kp_{t-6},. Features 13 to 18 correspond to BI_{t-1}, BI_{t-2}, …, BI_{t-6}. After we normalize the parameters, we compare the average values of the three groups between storm and nonstorm samples. Finally, we find that the difference between the two types of samples is large whichever among features 1 to 6 or 7 to 12 or 13 to 18. The average values of the nonstorm centers are almost half of those of the storm centers, which means the big difference of *Dp*, BI, and *Kp* between three centers. In addition, the meaning of the three groups of storm and nonstorm samples is that they can reflect the general characteristics. Hence, the difference between the three centers can be considered as the difference between storm and nonstorm samples. We have found data imbalance above, and Figure 3 demonstrates that the storm and nonstorm data have different characteristics. Then we should use two submodels to train for them separately. The two submodels are the "storm" model and "nonstorm" model. The specific model structure is explained in section 3.

3. *Kp* Forecasting With LSTM

RNNs are a type of artificial NN architecture inspired by the cyclical connectivity of neurons in the brain. RNNs use iterative function loops to store information (Graves, 2012). RNNs add recurrent connections on the hidden layer of the original NN, which means that the output of the hidden layer at the last moment also acts on the hidden layer of the current moment. Therefore, RNNs emphasize the link between the samples and can remember previous information. Figure 4 displays the unfolding structure of RNNs through time, and their time series length is *T*. Since RNNs add the dimension of time, they are effective in solving sequential prediction tasks. Thus, they are suitable for *Kp* forecasting.

The standard RNN just adds the recurrent connections to the hidden layer. And the difference between an LSTM and a standard RNN is that the LSTM is able to store the information for long periods of time in each

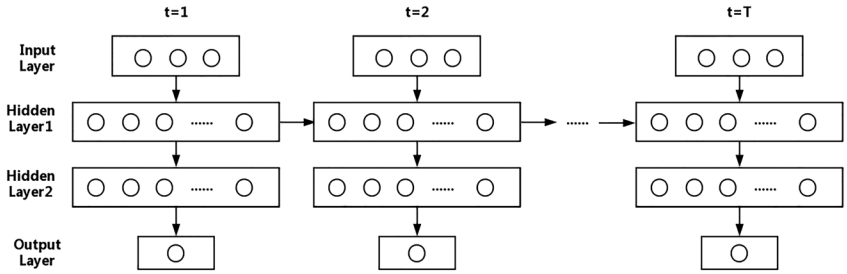

Figure 4. Unfolding network of Recurrent Neural Networks through time.

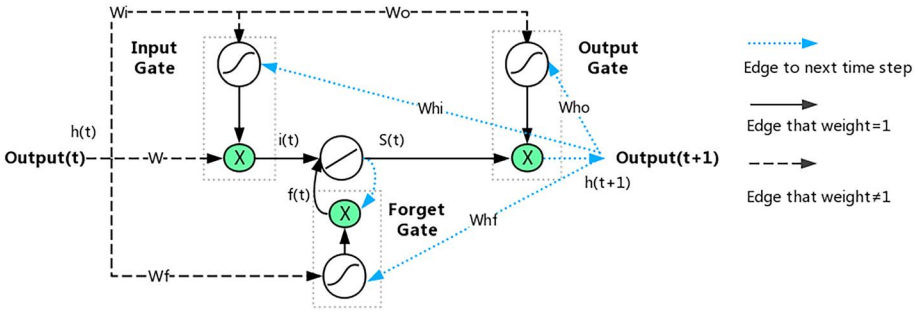

Figure 5. Working mechanism within a memory cell (some inputs are omitted in this figure).

memory cell. In fact, LSTM has been shown capable of storing and accessing information over very long time spans (Graves, 2012). The ordinary node of the hidden layer is replaced with a memory cell. All of the elements of the memory cell are described below.

1. Internal state: The core of a memory cell; a node with linear activation.
2. Input node: Receives input from other memory cells.
3. Constant error carousel: The self-connection with a weight = 1 in the internal state.
4. Input gate: A multiplied gate that controls how much the input content should provide.
5. Output gate: A multiplied gate that controls how much the output content should return as output.
6. Forget gate: A multiplied gate that controls how much content should be forgotten at the last moment.

$$i^{(t)} = \left(h^{(t)} \times W\right) \times \sigma\left(h^{(t)} \times W_i + h^{(t-1)} \times W_{hi}\right) \tag{3}$$

$$f^{(t)} = S^{(t-1)} \times \sigma\left(h^{(t)} \times W_f + h^{(t-1)} \times W_{hf}\right) \tag{4}$$

$$S^{(t)} = i^{(t)} + f^{(t)} \tag{5}$$

$$h^{(t+1)} = S^{(t)} \times \sigma\left(h^{(t)} \times W_o + h^{(t-1)} \times W_{ho}\right) \tag{6}$$

The training algorithm of LSTM is back propagation through time (Williams & Zipser, 1995). After adding the time dimension, the gradient of each neuron depends on the calculation not only of the current moment but also of the last moment. The whole training process is divided into a forward pass and a backward pass. After unfolding through time it adds a variable named "input length." Each time the input Xi of LSTM is a time series of length = T, Xi $\in R^{T \times m}$, where m is the number of attributes, which is also the node number of the input layer. Hence, the attribute element X of the data set D is a three-dimensional matrix, $X \in R^{n \times T \times m}$, with n as the size of the data set (Figure 5).

When building models, we have conducted many experiments to determine the suitable input length. Finally, we chose the input length $T = 6$ which provides a slight improvement when compared to other values. Now that we have confirmed the input parameters of the model, the input of the LSTM model is made up of the time series of BI, *Kp*, and *Dp* of length = 6 and $m = 3$. The output Y of the model is the *Kp* value at the moment $(T + 1)$.

Therefore, LSTM is the basic forecasting algorithm of the model described in this paper. Subsequently, we design a model structure that first classifies the samples into storm and nonstorm and then uses two separate forecasting submodels. LSTM ensures a smaller error of overall *Kp* forecasting because of its great learning capacity. Owing to the imbalance in the data set we discussed in section 2, we first combine the classification model and two regression submodels to improve the performance of geomagnetic storm forecasting. Hence, the system contains three submodels, denoted by O, A, and B and described in Table 3. The left part of Figure 6 displays the model structure described in this study.

As Table 3 shows, during training, the samples can be distinguished to be a storm or nonstorm sample according to whether $y_i < 5$ or $y_i \geq 5$. Therefore, we can use different training data sets to train submodels A and B. A is a "nonstorm" submodel and B is a "storm" submodel. We use the new training data set D1,

Table 3
Three Submodel Structures, Training Data Sets and Functions (D1 is Composed of Storm Samples and 1/10 Nonstorm Samples in the Original Training Data Set)

Submodel	The training algorithm	Training data set	Function
Model O	LSTM	D1	Classification of "storm" or "nonstorm"
Model A	LSTM	Original dataset	The model to predict *Kp* for "nonstorm" Samples
Model B	Linear regression	Storm samples	The model to predict *Kp* for "storm" samples

modified by random undersampling, to train the submodel O, which is the classification model. Hence, we reserve 1/10 nonstorm samples and storm samples of the original training data set to balance the ratio of the two classes of samples in D1. The new data set can accomplish a higher precision of the classification, which is actually geomagnetic storm forecasting. The input of model O is also the time series composed by *Kp*, BI, and *Dp* with length = 6. Model O is essentially a regression model, but we use the $y < 5$ and $y \geq 5$ to classify. Then, during the testing phase, we utilize the classification result to label all of the samples of the testing data set by "storm" or "nonstorm." Afterward, we send the "storm" samples of the testing data set into submodel A and the "nonstorm" samples of the testing data set into submodel B separately to obtain the predicted values, A_prediction and B_prediction.

It is worth mentioning that the model structures of submodels A and O are the same and their training algorithms are all LSTM. The two submodels just have different training data sets. The right part of Figure 6 represents the unfolding structure of submodels A and O in the paper. They have two hidden layers, the input layer has three nodes and the output layer has one node.

4. Experiments and Results

In order to test the effectiveness of deep learning in *Kp* forecasting, we compare another five *Kp* forecasting models trained by five types of algorithms dealing with the performance of *Kp* forecasting and geomagnetic storm forecasting. The five algorithms comprise the traditional autoregressive (AR) model (Kirchgässner & Wolters, 2007), linear regression, Lasso regression, support vector regression (SVR), and back propagation neural network (BPNN) (Zhou, 2016). They have their own characteristics, and all have been used in regression. The AR model emphasizes the link between the variables and itself, linear regression can find the linear relationship, SVR can solve different problems by leveraging different kernel functions, and we use the SVR with linear kernel. The Lasso regression is the linear regression with L1 regularization. Besides, BPNN can find the complex nonlinear relationships. The network setting of BPNN also has two hidden layers and the number of nodes about two hidden layers is the same as the LSTM's of our model. To ensure that the input variables are consistent with the model proposed in this paper, the inputs of the other four models, with the exclusion of the AR model are as follows: Kp_{t-1}, Kp_{t-2}, …, Kp_{t-6}, Dp_{t-1}, Dp_{t-2}, …, Dp_{t-6}, BI_{t-1}, BI_{t-2}, …, BI_{t-6}. Given that the AR model emphasizes the link between the variables and itself, the input of the AR model is Kp_{t-1}, Kp_{t-2}, …, Kp_{t-6}.

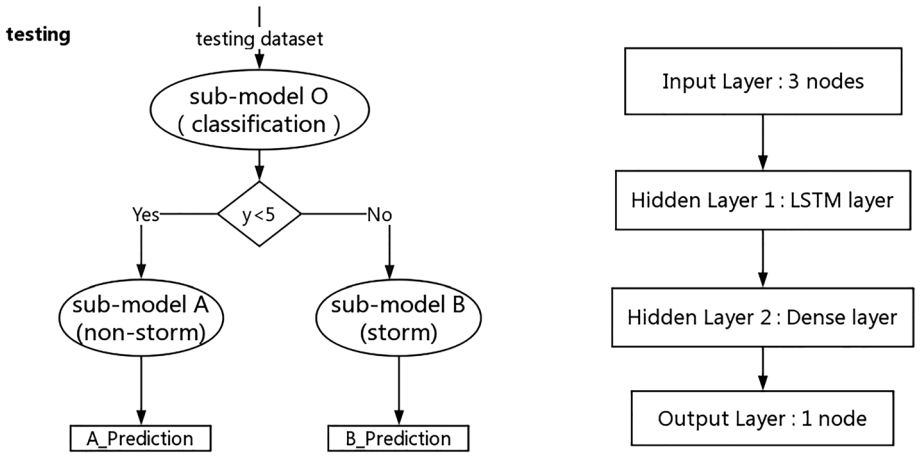

Figure 6. *Kp* forecasting model structure (left) and the structure of the submodels A and O (right).

Table 4
The Comparison on RMSE, MAE, and CC of Kp Forecasting Among Six Models

Model	MAE	RMSE	CC
AR model	0.5641	0.7593	0.7334
Linear regression	0.4969	0.6591	0.7997
Lasso regression	0.5091	0.6665	0.7968
SVR	0.4835	0.6522	0.8050
BP neural network	0.4946	0.6596	0.7990
Our model	0.4765	0.6382	0.8147

4.1. *Kp* Forecasting Evaluation

We choose four indicators to evaluate *Kp* forecasting performance. In the experiments, we compare the performance between the proposed forecasting model and the other five models. The four statistical parameters are: RMSE, mean-absolute error (MAE), correlation coefficient (CC), and the distribution of $\triangle Kp$. They are defined as follows:

$$\mathrm{CC} = \frac{\sum_{k=1}^{N}(P^k - \overline{P})(O^k - \overline{O})}{\sqrt{\sum_{k=1}^{N}(P^k - \overline{P})^2}\sqrt{\sum_{k=1}^{N}(O^k - \overline{O})^2}} \tag{7}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{k=1}^{N}(O^k - P^k)^2}{N}} \tag{8}$$

$$\mathrm{MAE} = \frac{\sum_{k=1}^{N}|O^k - P^k|}{N} \tag{9}$$

$$\Delta Kp = O^k - P^k \tag{10}$$

O^k is the kth observed *Kp*, P^k is the kth predicted *Kp*, $\overline{P}$, $\overline{O}$ are the average values, and N is the size of the data set.

Each of these indicators evaluates the model from a different perspective. RMSE and MAE measure the error, and CC measures the degree of fitting between the predicted and observed *Kp*. As such, the smaller RMSE and MAE are, the larger CC is, and the better the forecast performance will be. Then we compare the RMSE, MAE, and CC of *Kp* forecasting among the six models using the same data set. Table 4 shows the experimental results.

According to the results, we can find that the RMSE and MAE of the proposed model are the smallest among the six models which are 0.4765 and 0.6382, respectively. In addition, CC is the largest among the six models which is 0.8147. Additionally, Figure 7 shows the distributions of $\triangle Kp$, and they mostly obey a normal distribution. The mean μ of our proposed model's distribution is closer than that of the other four models, and the variance σ^2 is smaller than that of the other models. The distribution of our model is more concentrated, and the number of the peak value is the largest. Therefore, for each one of the four statistical parameters, our

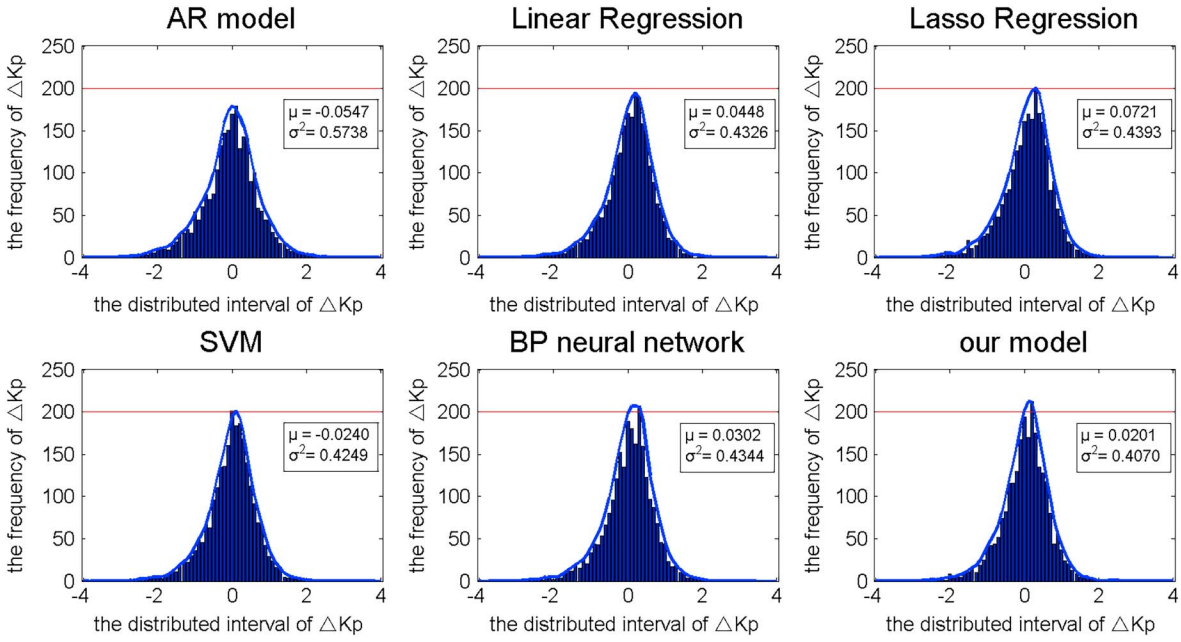

Figure 7. Comparison on the $\triangle Kp$ distribution of *Kp* forecasting among six models.

Table 5
The Confusion Matrix of Geomagnetic Storm

Observed *Kp*	Predicted *Kp* storm ($Kp \geq 5$)	Nonstorm ($Kp < 5$)
Storm	TP	FN
Nonstorm	FP	TN

model is better than the other five models in the *Kp* forecasting experiments. Specifically, the comparison of the BPNN and our model proves that LSTM is better at finding the regularity of *Kp* forecasting than the BPNN. It has also proved that it is helpful that LSTM emphasizes the link of the time series. Further, among the other five models, the RMSE, MAE, and CC of the model trained by SVR is better than the other four models. It is proven that SVR with a linear kernel can fit the model better, and it is probable that *Kp* has a linear relationship with the input vector. Although Lasso regression is a linear regression with L1 regularity, which can avoid overfitting, the RMSE and MAE of Lasso regression is worse than the one of linear regression without L1. Hence, it is unnecessary to limit the model complexity in the experiment.

4.2. The Geomagnetic Storm ($Kp \geq 5$) Forecasting

Geomagnetic storm forecasting is essentially a binary classification problem. In binary classification, samples can be divided into true positive (TP), false positive (FP), true negative (TN), and false negative (FN) by the true and predicted class. Additionally, the sum of TP, TN, FN, and FP is the size of the data set. Table 5 lists the confusion matrix (Zhou, 2016) of geomagnetic storms.

There are three parameters related to the confusion matrix, namely the precision ratio *P*, recall ratio *R*, and *F*1 measure. The definitions are listed below. *P* and *R* are a pair of contradictory measures that represents two different aspects. However, F1 is an integrated measure of *P* and *R* which makes *P* and *R* equally important. Hence, in order to evaluate geomagnetic storm forecasting, we choose the *F*1 measurement to reflect the forecasting performance.

$$P = \frac{TP}{TP + FP} \tag{11}$$

$$R = \frac{TP}{TP + FN} \tag{12}$$

$$F1 = \frac{2 \times P \times R}{P + R} \tag{13}$$

Similar to the above experiments, we compare the *F*1 of the six models. Table 6 is the experimental result of *F*1 with regard to the geomagnetic storm forecasting among the six models. Our model's *F*1 is larger than the other five models. Besides, a model just using LSTM algorithm whose network setting is the same as Figure 6 (right) for *Kp* forecasting without our model structure is denoted by LSTM (only). The comparison of *F*1 between our model and LSTM (only) can prove the advantage of our special model structure in geomagnetic forecasting. The last two rows are the experimental results. Our model is also larger than the LSTM model. Hence, the final model works for geomagnetic storm forecasting and our model makes storm forecasting more accurate.

In short, whether for *Kp* forecasting or geomagnetic storm forecasting experiments, our model has a clear advantage in two types of forecasting. It has a better performance than other models that are built on the AR model, linear regression, Lasso regression, SVR, or BPNN, as well as on the above enumerated parameters.

Given that some existing models predict 1 hr ahead (Bala & Reiff, 2012; Ji et al., 2013; Johnson & Wing, 2005) and the model in this paper predicts 3 hr ahead, we have compared the two 3 hr ahead models. One is the model proposed by Boberg in 2000 (Boberg et al., 2000) and the model inputs are *V*, *D*, and Bz. The other is the model proposed by Bala in 2012 (Bala & Reiff, 2012) and the input is the Newell function, viscous term. They were trained by a NN. We compare their models with the proposed model using the same data set.

Table 7 displays the experimental results regarding RMSE, MAE, and CC. Our model can get better experimental results compared to the existing models. Hence, our model not only achieves the improvement of *Kp* forecasting precision but essentially demonstrates that most geomagnetic storms can also be predicted accurately.

Table 6
The Comparison of F1 Among Six Models and the Model Trained Only by LSTM

Model	F1
AR model	0.1896
Linear regression	0.3448
Lasso regression	0.3448
SVR	0.3750
BP neural network	0.3571
Our model	0.5556
LSTM (only)	0.3871

Table 7
The Experiment Results Comparing Our Model and Other Existing Models

Model	Input parameter	Prediction time	MAE	RMSE	CC
Boberg (2000)	*V*, *D*, Bz	3 hr	0.6070	0.7986	0.6864
Bala (2012)	Newell function, viscous function	3 hr	0.5802	0.7508	0.7368
Our model	Bl, *Kp*,*Dp*	3 hr	0.4765	0.6382	0.8147

5. Discussion and Conclusion

In this paper, we compare the five other models with the proposed model based on LSTM. We regard LSTM as the basic algorithm of the proposed model in this work after the introduction of LSTM and evaluation of its effectiveness for the time series. We design the model structure using a classification technique. First, we use a classification model to classify the samples into "storm" and "nonstorm," then we train the two submodels to recognize the two types of samples.

In data analysis and processing, we first deal with the missing value. Then, we analyze the correlation between the 11 parameters and *Kp* at different lag moments; this is helpful for selecting the final inputs of the forecasting model. We also use a distribution of observed *Kp* to show the data imbalance and finally select the undersampling method to deal with the imbalance. For the imbalance, we consider that there is a significant difference between the storm and nonstorm samples. We found that the two classes of data need two separate submodels with the clustering result obtained by FCM. This is helpful for building the model structure.

In section 4, we compare the performances of *Kp* forecasting using RMSE, MAE, CC, and geomagnetic storm forecasting trained with *F*1. We compare five other models trained by other prediction algorithms with the proposed model. In the comparison experiments, for *Kp* forecasting the RMSE and MAE of our model are smaller than those of other models and CC is larger. Besides, *F*1 of our model is the largest among the six models when considering *Kp* forecasting or storm forecasting. This also shows that the *Kp* and other related parameters have a potential relationship between themselves at the moments before and after the prediction time. Therefore, the LSTM algorithm exploits these internal links to improve forecasting performance. Using LSTM for deep learning to build the *Kp* forecasting model is an innovative approach. In addition, geomagnetic storm forecasting has improved owing to classification first. Therefore, the proposed model is effective in raising the prediction accuracy and reducing the prediction error of *Kp*. In addition, the comparison of other existing models with our model with the same data set demonstrates that this research is helpful.

We discuss the forecasting models 3 hr ahead in the work. However, the sampling rates of SW and IMF parameters are 1 hr in the work. We will do some researches on forecasting models 1 hr ahead in the future.

Acknowledgments
We thank the GFZ German Research Centre for Geosciences for providing the Kp indices data (via ftp://ftp.gfz-potsdam.de/pub/home/obs/kp-ap/tab/). We thank the Advanced Composition Explorer (ACE) and National Aeronautics and Space Administration (NASA) for providing the ACE data (via ftp://sohoftp.nascom.nasa.gov/sdb/goes/ace/monthly/). This work was supported by the National Natural Science Foundation of China (grants 61432011 and 61732011).

References

Ayala Solares, J. R., Wei, H. L., Boynton, R. J., Walker, S. N., & Billings, S. A. (2016). Modeling and prediction of global magnetic disturbance in near-Earth space: A case study for *Kp* index using NARX models. *Space Weather*, *14*, 899–916. https://doi.org/10.1002/2016SW001463

Bala, R., & Reiff, P. (2012). Improvements in short-term forecasting of geomagnetic activity. *Space Weather*, *10*, S06001. https://doi.org/10.1029/2012SW000779

Bartels, J. (1949). The standardized index, *Ks*, and the planetary index, *Kp*. IATME Bull 12b,97.

Boberg, F., Wintoft, P., & Lundstedt, H. (2000). Real time *Kp* predictions from solar wind data using neural networks. *Physics and Chemistry of the Earth, Part C: Solar, Terrestrial & Planetary Science*, *25*(4), 275–280.

Boyle, C. B., Reiff, P. H., & Hairston, M. R. (1997). Empirical polar cap potentials. *Journal of Geophysical Research*, *102*(A1), 111–125. https://doi.org/10.1029/96JA01742

Costello, K. A. (1998). Moving the Rice MSFM into a real-time forecast mode using solar wind driven forecast modules, (Doctoral dissertation). Rice University.

Estabrooks, A., Jo, T., & Japkowicz, N. (2004). A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence*, *20*(1), 18–36.

Graves, A. (2012). *Supervised sequence labelling with recurrent neural networks* (Vol. 385). Heidelberg: Springer.

Ji, E. Y., Moon, Y. J., Park, J., Lee, J. Y., & Lee, D. H. (2013). Comparison of neural network and support vector machine methods for *Kp* forecasting. *Journal of Geophysical Research: Space Physics*, *118*, 5109–5117. https://doi.org/10.1002/jgra.50500

Johnson, J. R., & Wing, S. (2005). A solar cycle dependence of nonlinearity in magnetospheric activity. *Journal of Geophysical Research*, *110*, A04211. https://doi.org/10.1029/2004JA010638

Kirchgässner, G., & Wolters, J. (2007). *Introduction to modern time series analysis*. Berlin: Springer.

Newell, P. T., Sotirelis, T., Liou, K., Meng, C. I., & Rich, F. J. (2007). A nearly universal solar wind-magnetosphere coupling function inferred from 10 magnetospheric state variables. *Journal of Geophysical Research*, *112*, A01206. https://doi.org/10.1029/2006JA012015

Newell, P. T., Sotirelis, T., Liou, K., & Rich, F. J. (2008). Pairs of solar wind-magnetosphere coupling functions: Combining a merging term with a viscous term works best. *Journal of Geophysical Research*, *113*, A04218. https://doi.org/10.1029/2007JA012825

Pang, X. (2012). A comparative study of missing data interpolation processing method. *Statistics and Decisions*, *24*, 18–22.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, *4*, 3104–3112.

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3156–3164). Boston, MA: IEEE Computer Society.

Wang, J., Zhong, Q., Liu, S., Miao, J., Liu, F., Li, Z., & Tang, W. (2015). Statistical analysis and verification of 3-hourly geomagnetic activity probability predictions. *Space Weather*, *13*, 831–852. https://doi.org/10.1002/2015SW001251

Williams, R. J., & Zipser, D. (1995). Gradient-based learning algorithms for recurrent networks and their computational complexity. *Backpropagation: Theory, Architectures, and Applications*, *1*, 433–486.

Zhou, Z. H. (2016). *Machine learning* (in Chinese). Beijing: Tsinghua University Press.