Journal of Geodesy (2023) 97:52
https://doi.org/10.1007/s00190-023-01745-x

ORIGINAL ARTICLE

Ultra-short-term prediction of LOD using LSTM neural networks

Junyang Gou[1] · Mostafa Kiani Shahvandi[1] · Roland Hohensinn[1,2] · Benedikt Soja[1]

Received: 4 October 2021 / Accepted: 28 April 2023 / Published online: 29 May 2023

Abstract
Earth orientation parameters (EOPs) are essential in geodesy, linking the terrestrial and celestial reference frames. Due to the time needed for data processing and combining different space geodetic techniques, EOPs of the highest quality suffer latencies from several days to several weeks. However, real-time EOPs are needed for multiple geodetic and geophysical applications. Predictions of EOPs in the ultra-short term can overcome the latency of EOP products to a certain extent. Traditionally, predictions are performed using statistical methods. With the rapid expansion of computing capacity and data volume, the application of deep learning in geodesy has become increasingly promising in recent years. In particular, the Long Short-Term Memory (LSTM) neural networks, one of the most popular Recurrent Neural Network varieties, are promising for geodetic time series prediction. In this study, we investigate the potential of using LSTM to predict daily length of day (LOD) variations up to ten days in advance, accounting for the contribution of effective angular momentum (EAM). The data are first preprocessed to obtain residuals by combining physical and statistical models. Then, we employ LSTM networks to predict the LOD residuals using both LOD and EAM residuals as input features. Our methods outperform all other state-of-the-art methods in the first eight days with an improvement of up to 43% under the first EOP Prediction Comparison Campaign conditions. In addition, we assess the performance of LOD predictions using more extended time series to consider the improvements of EOP products over the last decade. The results show that extending data volume significantly increases the performance of the methods.

Keywords Earth orientation parameters (EOPs) · Length of day (LOD) · Long short-term memory (LSTM) · Deep learning · Hybrid modeling · Time series prediction

1 Introduction

Earth orientation parameters (EOPs) describe the motion of the Earth-centered-Earth-fixed reference frame (International Terrestrial Reference Frame, ITRF) w.r.t. the barycentric and barycentric-fixed celestial reference frames (International Celestial Reference Frame, ICRF) (Petit and Luzum 2010). This relation is a function of time and consists of several parameters, including Universal Time (UT1), coordinates of the pole (x, y) of Earth's rotation axis, and celestial pole offsets (dX, dY). Of particular interest is UT1, which is not a uniform time but includes variations due to irregularities in the Earth's rotation period. Two important parameters that can be derived from UT1 are the length of day (LOD, usually denoted by Δ) and the difference between UT1 and the Coordinated Universal Time (UTC), denoted by dUT1. LOD is the first negative derivative of dUT1 w.r.t. time (Gross 2007), i.e., $\mathrm{LOD} = -\mathrm{d}(\mathrm{UT1} - \mathrm{UTC})/\mathrm{d}t$ and usually provided with a diurnal sampling rate. LOD is not directly included in the transformation from ITRF to ICRF, but can be used to densify and predict dUT1, the determination of which relies on Very Long Baseline Interferometry (VLBI, Sovers et al. 1998; Senior et al. 2010; Mikschi et al. 2019). The variations in LOD are at the milliseconds level, corresponding to the sub-meter level on the Earth's surface at the equator. Therefore, accurate LOD plays a vital role in many space applications. Since high-precision EOPs (including LOD) are provided at latencies of up to around one month, their accurate prediction can overcome shortcomings for real-time demands in navigation or orbit determination, for example. As a result, the precise ultra-short-term prediction of LOD is an important task in geodesy.

✉ Junyang Gou
jungou@ethz.ch

[1] Institute of Geodesy and Photogrammetry, ETH Zurich, Robert-Gnehm-Weg 15, 8093 Zurich, Switzerland

[2] International Space Science Institute, Hallerstrasse 6, 3012 Bern, Switzerland

The fundamental requisite to understanding the changes in the EOPs is to apply the principle of conservation of angular momentum to the Earth system. Under this principle, two primary reasons for changes in Earth's rotation can be summarized: internal mass redistribution and the exchange of angular momentum between the solid Earth and fluid parts (Gross 2007). In addition, the applied external torques also have substantial impacts, known as tides (Dehant and Mathews 2015). Therefore, understanding the dynamics and redistribution of the fluid layers of the atmosphere, oceans, and the terrestrial hydrosphere is critical for understanding the non-tidal changes of EOPs, caused by the fluid layers of the Earth (Dobslaw and Dill 2019). The effective angular momentum functions (EAM), including the AAM (atmospheric angular momentum), OAM (oceanic angular momentum), HAM (hydrological angular momentum), and SLAM (sea-level angular momentum), describe the non-tidal geophysical excitations of EOPs (Dobslaw and Dill 2018, 2019). Among them, the zonal component (z-component) of the AAM dominates the non-tidal changes on a short timescale of LOD with some additional contributions from the ocean and hydrological system (Holme and De Viron 2013). In addition, the angular momentum is typically categorized into mass and motion terms, except SLAM, which only contains the mass term.

There have been many studies on the prediction of LOD time series, which can be divided into two categories, namely only based on LOD data or in a combination of LOD and its geophysical excitation. In the last few decades, most efforts focused on the first category. Many statistical forecasting methods have been applied, including the autoregressive moving average model (Hamdan and Sung 1996) and auto-covariance based procedure (Kosek et al. 1998). The combinations of individual methods were also realized, like the combination of fuzzy-inference system and wavelet filtering (Akyilmaz et al. 2011), which is superior to the purely fuzzy logic method for the ultra-short term. Xu et al. (2012) achieved accurate prediction with mean absolute error (MAE) up to 0.13 ms in the ultra-short term by combining least-squares (LS), autoregressive (AR) model and Kalman filter. Besides, many studies realized the potential application of machine learning algorithms for prediction of EOPs. Schuh et al. (2002) applied an artificial neural network (ANN) to predict polar motion and LOD, which showed comparable results to traditional statistical methods. Other studies using ANNs and their combination with stochastic methods can be found in (Kosek et al. 2005; Liao et al. 2012; Lei et al. 2017). In addition, Lei et al. (2015) applied another type of machine learning algorithm called extreme learning machine, which tried to reduce the computational complexity of the training phase. The first EOP Prediction Comparison Campaign (EOP PCC), organized by the Polish Academy of Sciences, took place from 2005 to 2009, including more than 20 methods (Kalarus et al. 2010). The primary conclusion of the first EOP PCC was that there was no best algorithm for all parameters in all prediction horizons. Besides, it also highlighted that more attention should be put on the contribution of EAM. Freedman et al. (1994) considered the contributions of meteorological AAM forecast data in their Kalman filter model. They concluded that the AAM information could be an important adjunct to geodetic measurements of Earth orientation. In recent years, more and more studies include forecasts of geophysical excitations (AAM or EAM) into EOP predictions (Niedzielski and Kosek 2008; Kosek 2012; Dill et al. 2019; Modiri et al. 2020; Kiani Shahvandi et al. 2022b). Among them, Modiri et al. (2020) combined the singular spectrum analysis (SSA) and Copula theory to predict LOD in the ultra-short term, which reached the state-of-the-art results of day-5 to 10 with MAE of 0.06 ms to 0.08 ms, whereas the Kalman filter algorithm (Freedman et al. 1994; Gross et al. 1998) dominated the prediction of the first four days with MAE of 0.04 ms to 0.06 ms. Only a few studies considered the EAM contributions in machine learning algorithms. Wang et al. (2008) incorporated the LOD and AAM prediction series into an ANN model to predict LOD for five days and proved a significant improvement up to 27.6%.

With the availability of more data and computational resources, deep learning approaches have gained colossal attention in recent years because of their high modeling and predicting power on different datasets. The recent achievements, shortcomings and outlooks of machine learning and deep learning approaches in geoscience are discussed by Reichstein et al. (2019) and Bergen et al. (2019). The authors concluded that the application of machine learning in geosciences is very promising and highlight the combination of physical modeling and machine learning approaches as a main future research direction. Motivated by the success of time series forecasting with deep learning, we present the application of this approach in the context of LOD prediction. In particular, we focus on the so-called sequential deep learning models because of the time series nature of LOD data. More specifically, among the sequential models, we use the Long Short-Term Memory (LSTM, Hochreiter and Schmidhuber 1997; Gers et al. 2000) neural network architecture, which is one of the most popular varieties of a Recurrent Neural Network (RNN, Rumelhart et al. 1985), as it can solve the vanishing gradient problem and thus, capture long-term dependencies. LSTM has proven to be highly accurate in the prediction of time series data in geosciences, including land deformation analysis (Pu et al. 2018), fog forecasting (Miao et al. 2020), and ocean temperature prediction (Zhang et al. 2020). The geodetic community also realizes the potential application of LSTM for predicting geodetic time series (Kitpracha et al. 2019; Kiani Shahvandi and Soja 2021; Kiani Shahvandi and Soja 2022).

Our study focuses on ultra-short-term prediction of LOD using a hybrid modeling approach, which combines physical, statistical and deep learning methods while considering geophysical excitations. We test four varieties of LSTM neural networks, considering both LOD and EAM (AAM+OAM+HAM) data as inputs for LOD prediction. In the preprocessing phase, we remove the secular trend, tidal and seasonal variations of LOD by combining the Savitzky–Golay (SG) filtering (Savitzky and Golay 1964), tidal corrections (Petit and Luzum 2010), and least-squares sinusoidal regression (Brockwell and Davis 2002) to generate the LOD residuals. We also use the sinusoidal regression to remove seasonal variations in the EAM time series (including 6-day and 10-day forecasts) to create the EAM residuals. Then, our networks consider the LOD and EAM residuals as input features to learn from them and predict the LOD residuals in the next ten days. Additionally, signal portions removed in the preprocessing phase (we call them *deterministic* in the following) are predicted to recover the full LOD signal. We first compare the performance of four LSTM variants with other studies of EOP PCC in a hindcast experiment (1996–2008) under the same conditions. Then, extended data from 1985 to the end of 2020 are considered to study the impact of data volume. In addition, all the experiments are repeated once with AAM and corresponding forecasts instead of EAM to analyze the contributions of oceanic and hydrological systems.

The rest of this paper is organized as follows: Sect. 2 introduces the principles of applied methods. The data, the preprocessing, as well as the prediction strategy, are discussed in Sect. 3. The results and discussions are reported in Sect. 4. Finally, conclusions are stated in Sect. 5.

2 Methods

2.1 Savitzky–Golay filter

For the extraction and the modeling of interannual and decadal trend signals, we use a SG filter (Savitzky and Golay 1964), which belongs to the class of finite impulse response filters (Oppenheim and Schafer 2013; Schafer 2011). The SG filter is based on the concept of least-squares curve fitting in successive windows. It resembles the so-called weighted moving average method in which the weights of smoothing differ according to the location in the window. However, to derive the weights, the coefficients of the mentioned curve are determined using the least-squares approach. In mathematical representations, with m being an integer, let the $(2m+1)$ values in the window around the j^{th} point of the time series y be $y_{j-m}, \ldots, y_{j+m}$. The SG filter tries to minimize the following quadratic function

$$\sum_{i=-m}^{+m} \sum_{k=0}^{q} (a_k i^k - y_{j-i})^2 \longrightarrow min \tag{1}$$

The coefficients $a_k,\ k = 0, \ldots, q$ are determined based on $y_{j-m}, \ldots, y_{j+m}$ and multiplied with the powers of $i = -m, \ldots, m$ to give a weighted moving average.

Compared to other filtering methods, the SG filter obtains a good trade-off between waveform smoothing and preservation of important signals and thus, outperforms other filtering methods (Liu et al. 2016). The kernel of a SG filter is only needed to be computed once and the smoothing process is based on discrete convolution (Savitzky and Golay 1964; Schafer 2011), which results in a remarkable computational efficiency. Therefore, SG filters are broadly used by the geoscience community to separate additive noise from signals (Liu et al. 2016; Roy 2020).

2.2 LSTM

In this part, we give an overview of the LSTM network architecture (Hochreiter and Schmidhuber 1997). Motivated by resolving the issue in modeling the long-term dependencies in RNNs (Rumelhart et al. 1985), LSTM uses the forget mechanism to prevent overflow of information (Gers et al. 2000). As a result, the so-called problem of vanishing gradients, which is a drawback of RNNs, is avoided. This makes LSTM a powerful method for sequential data prediction, where the order of and dependencies between values are essential. In the following, we review the LSTM architecture.

Definition 1 Let the inputs and targets to the LSTM network be denoted by $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ and $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]$, respectively, where $\mathbf{x}_i \in \mathbb{R}^{s \times f}$ and $\mathbf{y}_i \in \mathbb{R}^{o \times 1}$. The number of samples, sequence length, number of features and output length are denoted by the integers n, s, f and o, respectively. The basic LSTM network uses 8 groups of weights and 4 groups of biases to map the input $\mathbf{x}_i$ to the target $\mathbf{y}_i$. For this purpose, the temporal components of each input sample $\mathbf{x}_i$, which are denoted by $\mathbf{x}_{i,t}$ with $t = 1, \ldots, s$, are added recurrently. The mathematical representation of a single LSTM layer is as follows:

$$\begin{aligned}
\mathbf{I}_t &= \sigma\left(\mathbf{U}^I \cdot \mathbf{x}_t + \mathbf{W}^I \cdot \mathbf{h}_{t-1} + \mathbf{b}^I\right)\\
\mathbf{F}_t &= \sigma\left(\mathbf{U}^F \cdot \mathbf{x}_t + \mathbf{W}^F \cdot \mathbf{h}_{t-1} + \mathbf{b}^F\right)\\
\mathbf{O}_t &= \sigma\left(\mathbf{U}^O \cdot \mathbf{x}_t + \mathbf{W}^O \cdot \mathbf{h}_{t-1} + \mathbf{b}^O\right)\\
\mathbf{C}_t &= \mathbf{F}_t * \mathbf{C}_{t-1} + \mathbf{I}_t \odot \sigma\left(\mathbf{U}_i^C \cdot \mathbf{x}_t + \mathbf{W}_i^C \cdot \mathbf{h}_{t-1} + \mathbf{b}^C\right)\\
\mathbf{h}_t &= \tanh\left(\mathbf{C}_t\right) \odot \mathbf{O}_t
\end{aligned} \tag{2}$$

in which **I**, **F** and **O** are the input, forget, and output gates, respectively, whereas **C** refers to cell state. $\mathbf{h}_t$ denotes the so-called hidden states. In addition, the weights are denoted by $\mathbf{U}$ and $\mathbf{W}$, biases by $\mathbf{b}$, sigmoid function with σ, hyperbolic tangent function with tanh, and the Hadamard (element-wise) product with $\odot$.

Using the initial value of $\mathbf{h}_0 = \mathbf{0}$, the input, forget, output gates, and cell input are computed. Then, based on the initial value $\mathbf{C}_0 = \mathbf{0}$, the cell state is updated for the current step. The cell state and output gates are then used for the update of the hidden state. The computed values for hidden and cell states are used for the next steps. The essential steps would have larger hidden state values so that the input, forget and output states can be mapped onto a larger value (closer to 1). The reason is that a value closer to 1 in the sigmoid function can keep the hidden states in the next cell since smaller values (closer to 0) would have value 0 in the hyperbolic tangent function. Therefore, LSTM can effectively capture time series dynamics.

Definition 2 Let the outputs of the neural networks be denoted by $\hat{\mathbf{Y}}$. The optimization of the network by the loss function $\mathscr{L}(\hat{\mathbf{Y}}, \mathbf{Y})$ refers to the following minimization problem

$$\arg\min_{\theta} \sum_{i=1}^{n} \mathscr{L}(\hat{\mathbf{Y}}_i, \mathbf{Y}_i) \tag{3}$$

where θ denoting all the trainable parameters including the weights $\mathbf{U}$, $\mathbf{W}$ and biases $\mathbf{b}$.

Remark 1 We use the L_1 norm for the optimization, thus Eq. (3) reads

$$\arg\min_{\theta} \sum_{i=1}^{n} |\hat{\mathbf{Y}}_i - \mathbf{Y}_i| \tag{4}$$

Other choices for the loss function would be the $L_k,\ k = 2, 3, \ldots$ norms. However, for our problem of LOD prediction the L_1 norm has been proven to perform the best compared to our experiments using other loss functions. Furthermore, evaluation metrics will also be based on the L_1 norm, so it is logical to train the networks trying to minimize this norm. For the optimization process, the Adam optimizer is used to derive the optimal set of weights and biases (Kingma and Ba 2014).

Definition 3 The Encoder–Decoder LSTM (EDLSTM) architecture (Nayak and Ng 2020) is a variant of LSTM for sequence-to-sequence prediction problems where the number of input items differs from the number of output items. The encoder of the network is usually based on LSTM neurons, and the last hidden state of the encoder is considered as the context $\mathbf{C}$. Then, the context $\mathbf{C}$ will be repeated o time to fit the dimensions of the outputs, allowing application of sequential modeling in the decoder. As a result, the sequential relationship in the outputs can also be considered.

Definition 4 The CNN-LSTM (Donahue et al. 2015), which was originally designed for visual recognition and description, combines the concepts of Convolutional Neural Network (CNN) and EDLSTM. It introduces a CNN layer before the LSTM layers to further extract patterns from features. In our case, we introduce one-dimensional convolutional layers before the LSTM layers to extract patterns from the input features. The outputs of the CNN layers will be fed into the LSTM layers, which have different features as the original inputs $\mathbf{X}$.

Definition 5 Convolutional LSTM (ConvLSTM), proposed by Shi et al. (2015), is an architecture combining the convolution operator with the LSTM network. The difference between the LSTM and ConvLSTM lies in how the weights are multiplied with the inputs and hidden states. In ConvLSTM weights are convolved with the inputs, while in LSTM weights are multiplied with the inputs by usual matrix operations. In contrast to CNN-LSTM, which first produces high-level feature maps and then, feeds them into the sequential modeling procedure, the ConvLSTM considers the spatial and temporal correlations jointly.

2.3 Evaluation metrics

In this study, we evaluate the performance of our predictions using two metrics, namely the mean absolute error (MAE) and absolute error (AE). The difference between the j-days ahead prediction of the i^{th} sample of LOD is defined as:

$$\delta_{i,j} = \Delta^{\text{obs}}_{i,j} - \Delta^{\text{pred}}_{i,j} \tag{5}$$

where Δ^{obs} denotes the observed LOD, Δ^{pred} denotes the predicted LOD, i is the index of date, and j is from 1 to 10 for the ultra-short term. Then, AE is defined as:

$$\text{AE}_{i,j} = |\delta_{i,j}| \tag{6}$$

which shows the absolute difference between the predictions and the observations. To assess the averaged performance, we calculate the MAE:

$$\text{MAE}_j = \frac{1}{n_p} \sum_{i}^{n_p} \text{AE}_{i,j} \tag{7}$$

where n_p denotes the number of samples in the test set.

3 Data and processing strategy

3.1 Data description

3.1.1 LOD time series

In this study, we use the LOD time series provided by the International Earth Rotation and Reference Systems Service (IERS). The EOPs are derived from the combination of observations from different space geodetic techniques with diurnal sampling rate. In the first hindcast experiment, we use the IERS EOP 05C04 time series (Bizouard and Gambis 2009) to compare the performance of our methods with the results of the first EOP PCC under the same conditions. The IERS EOP C04 time series was updated during the first EOP PCC from 97C04 to 05C04, which caused difficulties in the evaluation process. However, the impact on LOD quality was minor (Kalarus et al. 2010), so we do not consider it here. For the subsequent experiments with longer time series, we use IERS EOP 14C04 (Bizouard et al. 2019), which results from the combination of operational EOP series that are consistent with ITRF2014 (Altamimi et al. 2016).

3.1.2 EAM time series

One of the pivotal EAM analysis data sets are provided by the GFZ German Research Center for Geosciences (GFZ). GFZ provides effective angular momentum functions, including AAM, OAM, HAM, and SLAM from 1976 onward (Dobslaw and Dill 2018). In addition, they provide the corresponding 6-day forecast data of the individual components and 90-day prediction data of EAM (AAM+OAM+HAM). As we described in Sect. 1, the z-components have a major impact on LOD. Therefore, we consider the z-components (mass+motion) of EAM, denoted as EAMz in the following, and the 6-day forecast data of EAMz (denoted as EAMz forecasts) as additional features. Additionally, we implemented the same experiments with AAMz-only (including the forecasts) instead of EAMz to assess the contributions of oceanic and hydrological systems. In the following text, we will only describe the approaches focusing on EAMz, since all the processing steps are the same for EAMz and AAMz. A difficulty is that the 6-day forecast data are only archived back to 2016. Therefore, we have to simulate the forecast data for the time interval before 2016. To achieve this goal, we downgrade the observed EAMz data by adding white noise considering the root-mean-square deviation (RMSD) of the 6-day forecast data reported in (Dobslaw and Dill 2019, Figure 8.4) to generate pseudo-forecast data. To verify the quality of the synthetic data, we generated the pseudo-forecast data also for 2016 onward and compared them with the real forecast data. The noise levels agree with each other. Nevertheless, the actual errors are not purely white noise, which may harm the performance of our networks (Dill et al. 2021).

Since 2021, ETH Zurich has also investigated EAM forecasts based on GFZ EAM products to improve the accuracy and prediction horizon (Kiani Shahvandi et al. 2022a). The operational 14-day forecasts of the complete set of EAM components, based on first-order neural ordinary differential equations, are available from the Geodetic Prediction Center[1] at ETH Zurich (GPC, Soja et al. 2022). The details about these EAM forecasting algorithms can be found in Appendix A. Therefore, we also implemented the experiments with the ETH EAM forecasts as features instead of GFZ EAM forecasts, which also helps to analyze the impact of longer EAM forecasts. The 10-day forecasts of ETH EAMz and AAMz are used since the errors of the forecasts on the latter days increase steadily to a point where the longer forecasts have negative impacts on the LOD predictions. Here, we have a similar problem as for the GFZ forecasts: the real EAM forecasts are only archived back to 2021. Therefore, we generate the pseudo-forecasts following the same approach. All the used RMSD for generating pseudo-forecasts are reported in the supplementary information.

3.1.3 Splitting of data sets

For training the networks, we divide the whole dataset into training, validation, and test sets. Figure 1 shows the LOD datasets used in this study. For the first experiment, the training and validation data sets start on 1996–09–01 and ends on 2005–09–30, whereas the test set starts on 2005–10–01 and ends on 2008–10–31, as depicted in Fig. 1a. However, not all the daily data in this time range will be used for the test set because the submission of predictions for the first EOP PCC was only on Thursdays. Therefore, there are 161 test samples in total. The splitting of the training and validation sets is not based on the date but on the number of samples. Here, we have 3000 training samples and 248 validation samples. For the other experiments that rely on the extended data, the training and validation data sets start on the beginning of 1985 and the test set starts on 2016–01–01 and ends on 2020–12–31, as shown in Fig. 1b. In this experiment, the models predict the LOD every day. Therefore, we have 1827 test samples in total. The training and validation sets are also split based on number of samples. We have 9025 (80%) training and 2026 (20%) validation samples, respectively.

3.2 Data processing

Figure 2 shows the data processing workflow used in this study. Since we can treat some parts of the LOD-contributing

[1] https://gpc.ethz.ch/.

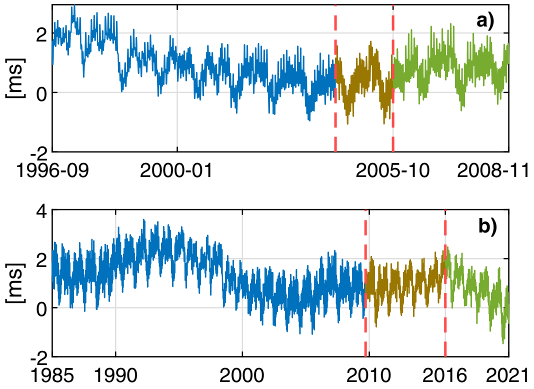

Fig. 1 LOD datasets used in this study. The training (blue), validation (dark yellow) and the test (green) data of **a** experiment 1 under the first EOP PCC conditions, and **b** experiments 2 and 3 with extended time series from 1985 to the end of 2020

signals as deterministic, we subtracted them prior to the application of LSTM and restored them afterward. Since the inputs to LSTM neural networks should be detrended, standardized, and normalized (Goodfellow et al. 2016), the preprocessing approach is necessary and beneficial. We first remove the secular trend and known signals from the LOD time series by combining the SG filter, tidal corrections and least-squares adjustment to generate the LOD residuals (Δ_R). Then, we use LSTM networks to predict Δ_R.

As described by Gross (2007), the LOD time series contains decadal, interannual, seasonal, and interseasonal signals. Besides, a secular trend of around 1.8 ms/century is observed. Since the tidal deformations (zonal tides) can explain a large part of the LOD signals, we first removed the tidal variations (Δ_{T}) following the IERS conventions 2010 (Petit and Luzum 2010) and obtained LOD without tidal variations Δ_1. For computing Δ_2, we removed the decadal trend of LOD (Δ_{S}) from Δ_1 by applying the SG filter with the polynomial order of 2 and the window length of 3653. The motivation for this choice is to optimally capture the long-term trend using a low-order nonlinear polynomial and obtain a smooth Δ_{S}. However, in the first experiment using data from 1996 to 2005, the length of the LOD time series is shorter and the secular trend within around ten years is easier to model. Thus, we use a quadratic function to model the secular trend for EOP PCC instead of the SG filter. We further applied the least-squares sinusoidal regression to estimate the seasonal signals with annual, semiannual and terannual periods (Δ_{P}). By further removing Δ_{P} from Δ_2, we received the LOD residuals (Δ_{R}).

Before applying a similar strategy to the EAM observations and forecasts to obtain residuals, we should project them

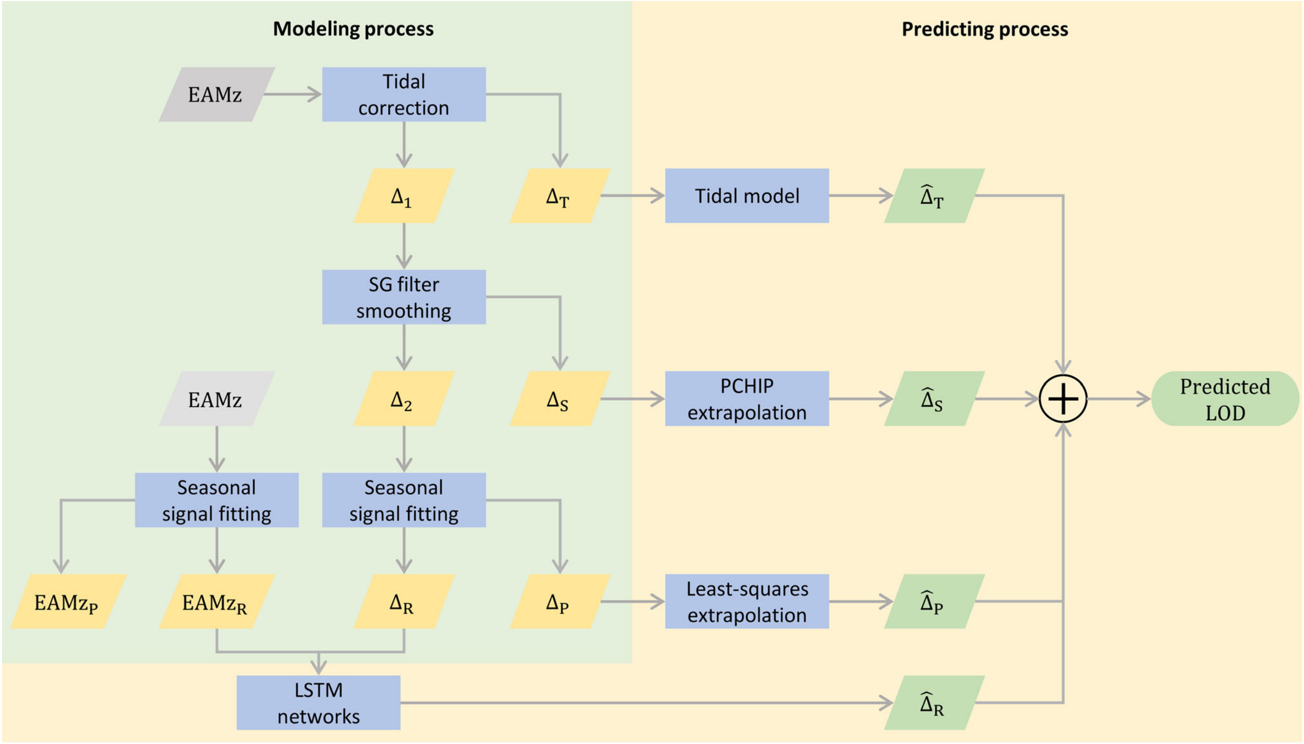

Fig. 2 The data processing workflow of this study. The gray parallelograms indicate the input data. The yellow parallelograms indicate the intermediate values after certain processing steps. The green parallelograms show the predicted and modeled values in the future. The blue rectangles show the processing methods. The plus symbol means sum of the four terms

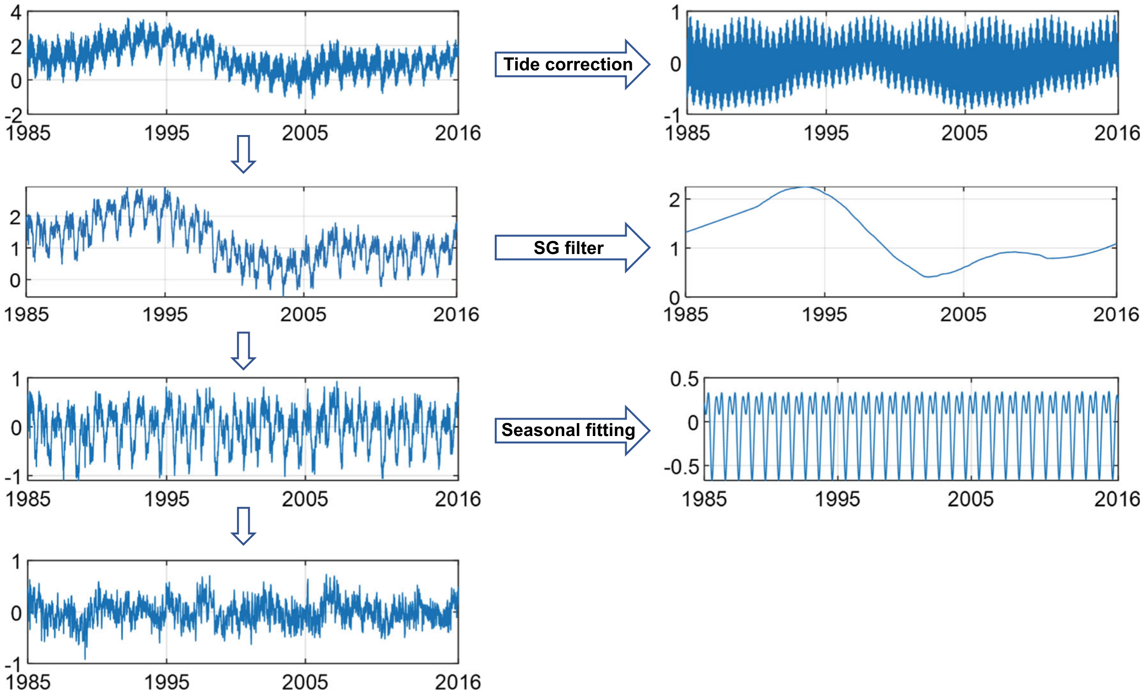

Fig. 3 An example of preprocessing the LOD [ms] data based on the IERS EOP 14C04 time series between 1985 and 2016. The left-bottom plot shows the LOD residuals that enter the LSTM neural networks beside the other features

to similar value ranges as the LOD data to reduce training difficulties. Since the used EAM functions from both sources are in the form of non-tidal normalized EAM (Dobslaw and Dill 2018, 2019), they are unitless and highly correlated with the LOD after removing tidal effects and secular trend. Therefore, we cannot apply geophysical equations to convert EAMz into LOD but should use a simple empirical relationship to convert the unit and obtain a similar value range:

$$\mathrm{EAMz}\ [\mathrm{ms}] = \mathrm{EAMz} \cdot 86400 \cdot 10^{3} \tag{8}$$

which agrees with the equation given by Gross (2007). Then, the least-square sinusoidal regression was applied to the EAMz time series to remove the seasonal signals with the same frequencies as for LOD. In the end, we obtained both LOD and EAMz residuals, which are in the interval of ± 1 ms and highly correlated with each other. Both Δ_{R} and $\mathrm{EAMz}_{\mathrm{R}}$ will later enter the LSTM neural networks as input features. To preprocess the test set, we always concatenated the ’new’ data with the training and validation set and reran the preprocessing process, which caused edge effects as a consequence of the filtering step. To overcome this issue, we applied an additional linear offset adjustment as described in Appendix B.

In summary, Fig. 3 shows the whole preprocessing progress for LOD and corresponding modelled signals using the training and validation set of the experiments with extended data. The right column of Fig. 3 shows the modelled signals, and the left column shows the corresponding remaining signals before and after removing the modelled parts. The outputs of the preprocessing approach (left-bottom plot) will be used as the inputs of the LSTM neural networks. After the LSTM-prediction step, the predictions of the deterministic signals (right-hand-side of Fig. 3) are added to $\hat{\Delta}_R$ (Fig. 2) in order to produce the predictions of LOD.

3.3 Prediction of LOD residuals using LSTM neural networks

After preprocessing, we reshaped the whole time series into samples and divided them into training, validation, and test sets. In order to tune the hyperparameters and choose the best structure of networks, we applied the grid search strategy and tenfold cross-validation (Goodfellow et al. 2016). In this step, we tested different input sequence lengths for the prediction of ten days in future and concluded that the best results were obtained by considering 30 previous days as inputs. Another motivation for using 30 previous days as input is to model the monthly and biweekly LOD signals, which are not explicitly removed beforehand, using the networks. We reshaped the whole time series into three-dimensional tensors with the dimensions of $n \times s \times f$ to fulfill the requirement of LSTM networks (Sect. 2.2), where n is the number of samples as reported in Sect. 3.1, s is the length of input sequence, and f is the number of features. The features of each input epoch are the LOD residuals, the observed EAMz residuals, and the

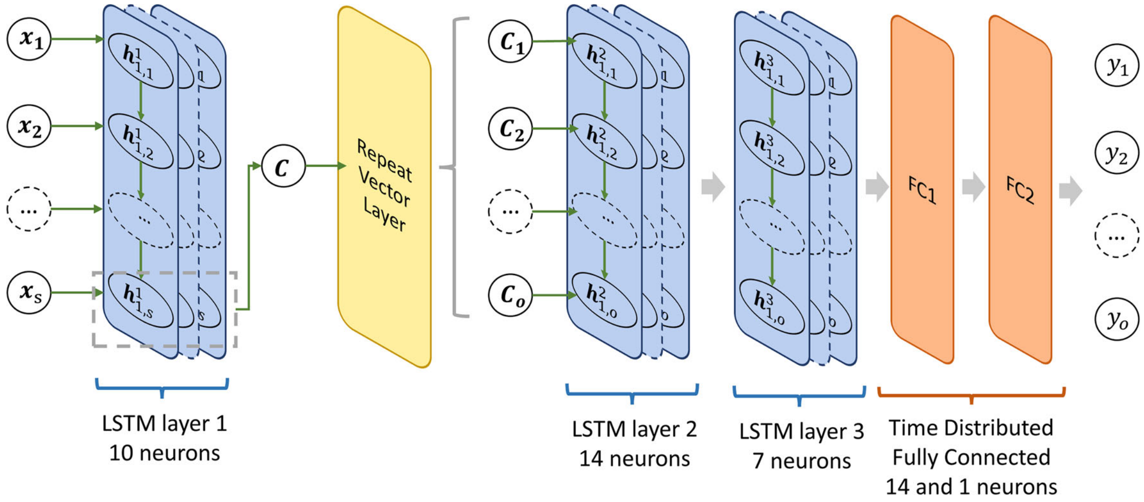

Fig. 4 Schematic diagram of the EDLSTM network in the experiments with longer time series. The inputs, hidden states and outputs are denoted using **x**, **h**, y, respectively. **C** refers contexts. The data flow through the layers from left to right, but also from the begin to the end of a sequence in the LSTM layers

forecasted EAMz residuals, respectively. The LSTM, EDLSTM, and CNN-LSTM networks take this three-dimensional matrix as input. However, this matrix has to be reshaped again to fulfill the requirement of the ConvLSTM network. In order to prevent overfitting, we applied an early-stopping strategy: we monitored the loss of the validation set and stopped the training phase when the loss stops decreasing or starts increasing. We noticed that the limited data volume under the conditions of the first EOP PCC is not enough to train our networks so that they can distinguish different prediction days. Therefore, we implemented ten separate networks to predict the ten different days in future to reduce this issue to a certain degree. In the second and third experiments, one network was sufficient to predict all ten days of LOD because a significantly larger amount of data was available.

We used *TensorFlow V2.4.0* (Abadi et al. 2015) to implement all the networks as well as the training, validation, and test processes. Figure 4 shows the structure of the EDLSTM network for the experiments with longer time series. The first *LSTM* layer is the encoder section, which is responsible for interpreting the input sequence. This layer takes the input tensor $\in \mathbb{R}^{30\times f}$ and outputs a non-sequence context $\mathbf{C} \in \mathbb{R}^{o}$ with o denoting the output dimension. Then, the *RepeatVector* layer repeats the context so that the required length of the output is fulfilled, followed by the decoder section that consists of two *LSTM* layers. In the end, two *Dense* layers map the outputs of the decoder section with the outputs. Since the output of the decoder section is a sequence, the *Dense* layers have to be wrapped using the *TimeDistributed* wrapper. The structures of all the other networks are reported in the supplementary information.

3.4 Prediction of deterministic components

In addition to predicting the LOD residuals with the LSTM networks, we have to predict the deterministic components of LOD. During the preprocessing steps described in Sect. 3.2, we obtained functional forms to describe the tidal variations (IERS convention 2010, Petit and Luzum 2010) and the parameters of the seasonal signals (least-squares sinusoidal regression). Therefore, we could extrapolate these two components into the future to obtain $\hat{\Delta}_{\mathrm{T}}$ and $\hat{\Delta}_{\mathrm{P}}$. Since only 30 previous days were considered as inputs in our networks, the trend function in this short time interval was usually linear, sometimes with a slight curvature. Therefore, we used the PCHIP (Piecewise Cubic Hermite Interpolating Polynomial) extrapolation (Fritsch and Carlson 1980) to precisely predict the secular trend.

3.5 Setups of experiments

In this study, we performed three experiments to evaluate the performance of our proposed methods comprehensively. The basic setups of the experiments are summarized in Table 1. First, we implemented a hindcast experiment to compare the performance of our methods to the other studies of the first EOP PCC. In an operational setting, we cannot use the IERS final products as our inputs due to the latency of one month and would have to resort to the rapid products which have a latency of two days for LOD. However, the historic rapid products are not fully archived once the final products are available. Therefore, we can only use the final products to train our networks and compute the predictions. The latency

Table 1 The setups of the three experiments

ID	Training and validation time span	Test time span	Output shape	Remark
Experiment 1	Sept. 1996–Sept. 2005	Oct. 2005–Nov. 2008	(n, 1)	12 networks for 12 days
Experiment 2	Jan. 1985–Dec. 2016	Jan. 2017–Dec. 2020	(n, 12)	One network for 12 days
Experiment 3	Jan. 1985–Dec. 2016	Jan. 2017–Dec. 2020	(n, 10)	One network for 10 days

of two days was still considered to provide the most realistic results. For the following experiments, we expanded the data volume to investigate the benefits of longer time series on the training process and the prediction accuracy that would be achievable nowadays. In the second experiment, we considered both the longer time series as well as the latency of input data. Therefore, the reported quality should be operationally achievable. In the final experiment, we studied the most optimistic accuracy by considering the optimal data availability without any latency to assess the theoretical limits of the proposed methods.

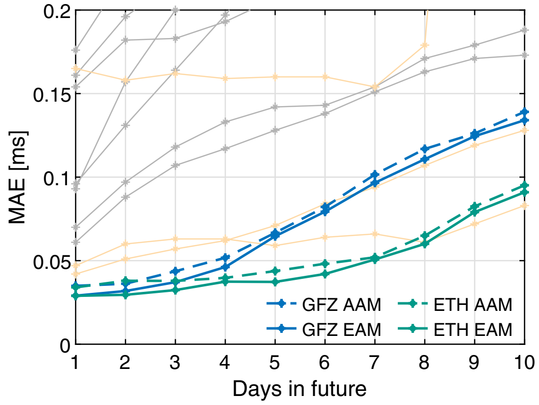

Fig. 5 The MAEs of the predicted LOD using EDLSTM networks between Oct. 2005 and Oct. 2008 under the first EOP PCC conditions (first experiment). The results of other studies under the EOP PCC conditions are denoted as light gray lines (without AAM/EAM) and light yellow lines (with AAM/EAM) for comparison

4 Results and discussion

We have generated predictions using the four LSTM variants in all three experiments. The results for the individual methods are reported in the supplementary information. We observed that the differences between the four variants are minor while the EDLSTM networks slightly outperform the others in most of the cases, with the lowest number of trainable parameters. Therefore, we conclude that the key to LOD prediction problem is the LSTM neuron rather than the different structures in encoders. We will thus only report the results using EDLSTM networks in this section and focus on the impacts of AAM and EAM and their forecasting horizons.

4.1 First experiment

In the first hindcast experiment, our study was carried out under the same conditions as for the first EOP PCC (Kalarus et al. 2010), resulting in a relatively fair comparison with others. The splitting of training, validation and test sets is reported in Sect. 3.1.3. We consider the latency of the IERS LOD rapid products, typically two days. Therefore, we generate LOD predictions for the next twelve days and take the latter ten predictions as our 10-day predictions. In this context, the 1-day delay of EAM forecasts has no impact. Figure 5 shows the MAE of our results and compares them to other studies of EOP PCC and the study by Modiri et al. (2020), which is the state-of-the-art study for days 5–10 (Modiri et al. 2020, Table 1). The MAE values are reported in the supplementary information. Concerning the comparisons with Modiri et al. (2020), we note that the EAM forecasts are generated differently and based on different data sources. Moreover, the ways to handle the latencies of the underlying products are also different (Modiri, personal communication).

This comparison proves the promising short-term performance of our method. Using the GFZ 6-day forecasts, we achieve the best results over the first four days. Afterward, the accuracy of our methods decreases more rapidly due to a lack of EAM forecasts, indicating the importance of the EAM forecasts for our purpose. Therefore, the ETH 10-day EAM forecasts significantly increase the quality of LOD predictions with an average improvement of 29% compared to using only 6-day EAM forecasts. As a result, our method outperforms other studies in the first eight days with a maximum improvement of 43% on day 3, whereas the method of Modiri et al. (2020) is still slightly better for days 9 and 10. Moreover, we observe steadily better performance if the networks using EAM instead of AAM with an average improvement of 10% for ETH 10-day forecasts, which proves the important contributions of the oceanic and hydrological systems to the LOD variations.

Figure 6 shows the AEs of the individual predicted days of all the test samples between 2005 and 2008. Due to the small amount of data, the networks are not extensively trained, which leads to more noisy patterns starting from day 5. When

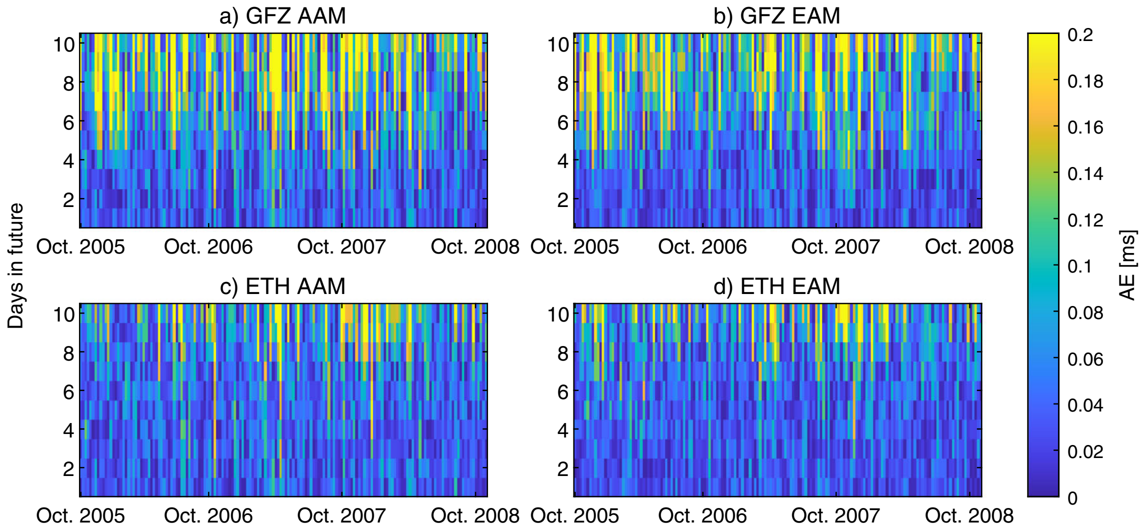

Fig. 6 Absolute errors of the predicted LOD using EDLSTM networks with different input features (first experiment). The test time interval follows the first EOP PCC conditions

comparing the results of using GFZ 6-day and ETH 10-day predictions, we observe significant reductions of the outliers on latter days. This phenomenon indicates the considerable help of the future EAM information to the LOD predictions. The contributions of OAM and HAM are clearly visible, given the fact that the errors in extreme cases, such as the one around October 2006, can be reduced.

4.2 Second experiment

For this experiment, we use data from 1985 to 2016 to train our networks. In order to obtain results under operational conditions, we also consider a LOD latency of two days, which means we again generate 12-day-ahead predictions and consider the last 10 days as the predictions in the future. The networks benefit from the more extensive data volume and therefore achieve a better generalization. As a consequence, we are able to determine all twelve outputs from one network rather than twelve separate networks as in the first experiment. Figure 7 shows the MAEs of our results with the numerical values shown in the supplementary information. We note that the comparison with the results from the first EOP PCC is not entirely fair since the expanded data volume and improvements in the EOP time series (from EOP 05C04 to EOP 14C04) positively impact the results. The goal of plotting the results of other studies is to give an intuition about the performance of our methods.

The more extended amount of data enables us to train one network with twelve outputs simultaneously, which results in more stable predictions. The contributions of more extended EAM forecasts are significant since the average MAE of the results from the network using 10-day EAM forecasts is about 32% lower than from the network using 6-day forecasts. As a result, the network with 10-day EAM forecasts can provide high-quality 10-day LOD predictions with MAE of about 0.08 ms. However, the benefits of using EAM are not ensured for the latter days since we can also observe that the network using 10-day EAM forecasts can only give comparable or even worse results compared to the network using 10-day AAM predictions. A possible reason for this degradation is the higher forecasting errors of AAM, OAM and HAM. Since the EAM time series is the sum of the three terms, the forecasting errors of all three sources also propagate into the EAM forecasts. Therefore, the signal-to-noise ratios for the latter days are worse than the earlier days, and the network cannot effectively benefit from the meaningful signals.

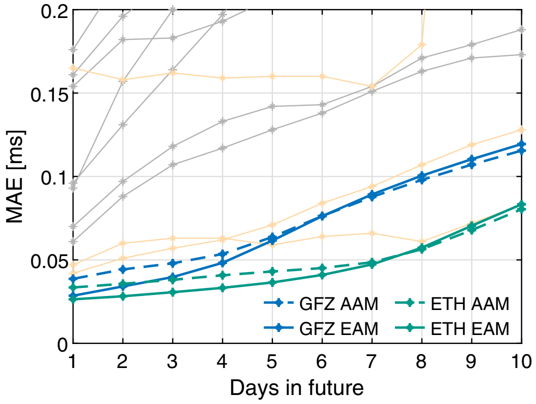

Fig. 7 The MAEs of the predicted LOD using EDLSTM networks between 2016 and 2020 (second experiment). The results of other studies under the EOP PCC conditions are denoted as light gray lines (without AAM/EAM) and light yellow lines (with AAM/EAM)

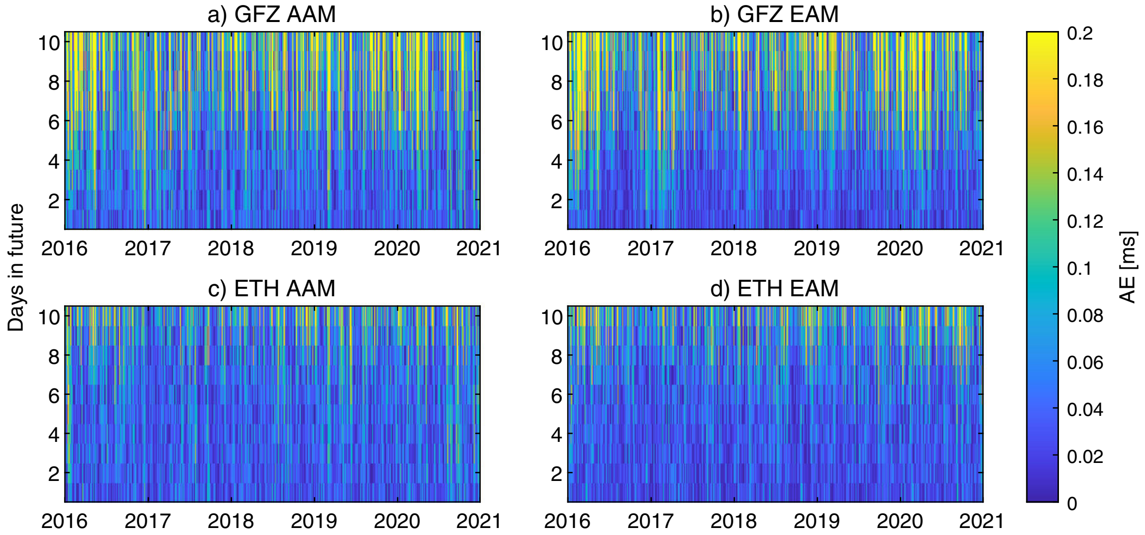

Fig. 8 Absolute errors of the predicted LOD using EDLSTM networks with different input features (second experiment). The test time interval is from 2016 to the end of 2020

Figure 8 shows the AEs of individual prediction days of all the samples in the test set of the second experiment. We observe outliers at the beginning of 2016 and 2020, which are likely caused by the biases in the GFZ 6-day forecasts (Dill et al. 2021). Due to the existence of these biases, the distribution of the generated pseudo-forecasts of EAM before 2016 differs from the distribution of the actual forecasts, which can cause a systematic error in our LOD predictions. In contrast, no noticeable irregularity is detected from the results using ETH 10-day predictions. However, this report could be slightly over-optimistic since no real EAM forecasts are available in this time interval and, therefore, the pseudo-predictions are used, which optimistically assumes the identical distributions of the training and test sets. In future work, we will continuously monitor the performance of our EAM forecasts to analyze the distributions of errors.

4.3 Third experiment

In the third experiment, we want to explore the achievable accuracy when using our methods under optimal conditions without the latency of the LOD and EAM observations. There are three motivations behind this experiment: First, it is crucial to fill in the missing values of LOD caused by the delay to receive the continuous time series, even though these missing values are not in the future. Secondly, we want to analyze the potential benefits if the processing time of LOD observations can be reduced. Furthermore, it is also more trivial to understand the correlation between EAM forecasts and LOD predictions since the time shift of two days does not exist anymore. Figure 9 shows the MAEs of the results with the numbers shown in the supplementary information.

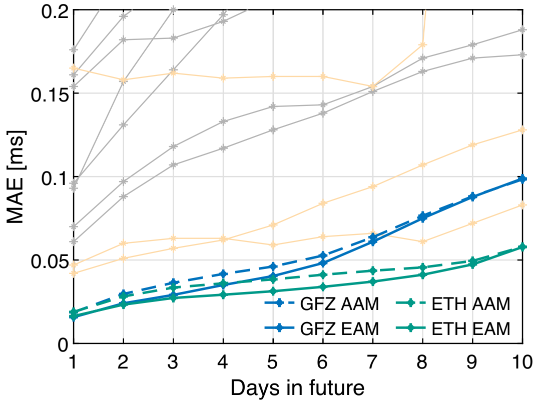

Fig. 9 The MAEs of the predicted LOD using EDLSTM networks between 2016 and 2020 without considering the latency of LOD observations (third experiment). The results of other studies under the EOP PCC conditions are denoted as light gray lines (without AAM/EAM) and light yellow lines (with AAM/EAM)

As the evidence in Fig. 9, the networks provide accurate LOD predictions with MAEs under 0.1 ms using GFZ 6-day EAM forecasts and under 0.06 ms using ETH 10-day EAM forecasts. We notice that the results shown in Fig. 9 are not fully equivalent to the ones shown in Fig. 7 shifted by two days. Because the networks only have ten outputs instead of twelve, the losses changed as well, see Eq. (4). The higher degradation of predictions based on 6-day EAM forecasts demonstrates the importance of longer EAM forecasts for our task. In contrast, MAEs of the predictions based on 10-day EAM forecasts increase more slowly due to the availability of EAM forecasts for the whole prediction horizon. The slightly higher gradient on latter days can be explained by the increasing uncertainties of the EAM forecasts on latter days.

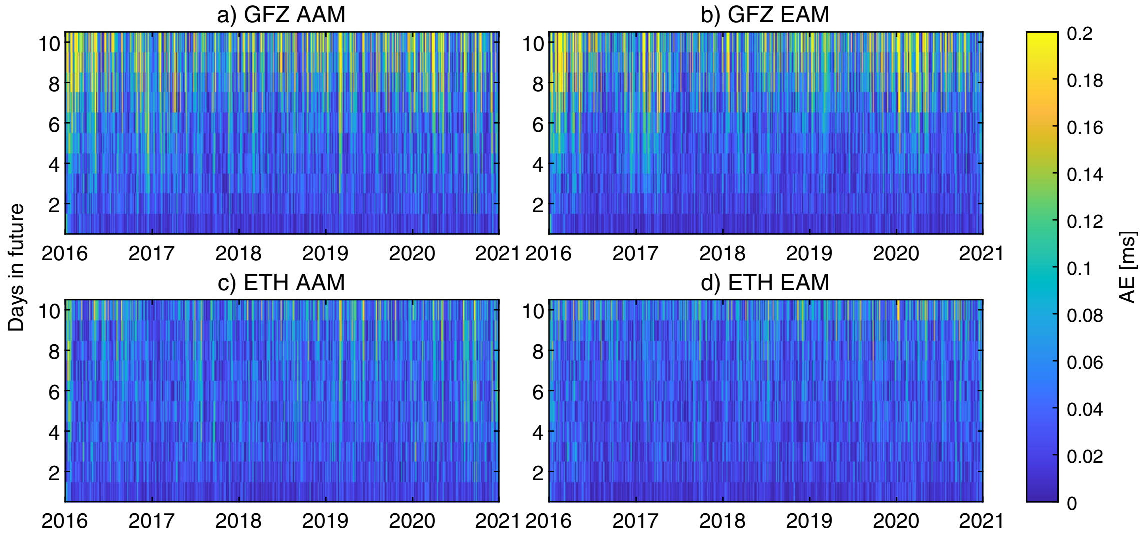

Fig. 10 Absolute errors of the predicted LOD using EDLSTM networks with different input features under the most optimistic conditions (third experiment). The test time interval is from 2016 to the end of 2020

The impact of the uncertainties in EAM forecasts can also be observed by the convergence of MAEs using EAM and AAM. As we discussed in the second experiment, the superiority of EAM forecasts decreases with predicting horizon as the forecasting errors of EAM forecasts are higher than that of AAM forecasts.

Figure 10 shows the optimistically achievable AE of individual prediction days of all the test samples between 2016 and the end of 2020 for the third experiment. We observe that most of the significant errors during the latter days are reduced in all four cases. However, the problems caused by the biases in the GFZ forecast are still observable at the beginning of 2016 and 2020. The results for ETH 10-day forecasts of AAM and EAM are similarly homogeneous.

4.4 Feature analysis

In this section, we discuss the impact of various features and different parts of the input sequences. The deep learning models have a black-box nature to a certain extent because they can, in principle, approximate any function, but it is difficult to understand the internal structure of the trained networks. Unlike other machine learning methods (e.g., random forests), there is no simple method to describe the importance of features in LSTM neural networks. To address this issue, we implemented empirical experiments, namely modifying the inputs by randomly adding or removing values equal to 30% of their magnitudes, followed by an investigation of the impact on the resulting MAE changes. The motivation for changing the magnitude of input values by 30% is that the changes are significant enough for us to learn the consequences. At the same time, the changes are also not too large to significantly impact the distribution of the features, which would hurt the performance of neural networks. We repeated this process 1000 times with different random seeds and computed the averaged MAE of these 1000 MAEs to obtain a general conclusion. Then, the relative changes of MAE are computed using the equation:

$$\mathrm{MAE}_{\mathrm{changed}}[\%] = \frac{\mathrm{MAE}_{\mathrm{new}} - \mathrm{MAE}_{\mathrm{original}}}{\mathrm{MAE}_{\mathrm{original}}} \times 100\% \qquad (9)$$

We performed this analysis based on the networks in the third experiment. In this case, there is no temporal gap between the EAM forecasts and LOD observations, which simplifies the interpretation of the temporal relationship between the features and outputs.

First, we changed 30% of individual features on all the 30 input days to study the impact of different features. Figure 11 shows the resulting changes of MAE in percent on a logarithmic scale. As expected, the LOD observations are the most prominent features because they provide direct information related to our targets. Beside LOD, the analysis data of EAM are also important, especially for the first days, although they do not offer any horizon in the future. The critical contribution of the analysis data is to provide information about misalignments between the geophysical excitations and the LODs. Therefore, the EAM forecast data can contribute more to the LOD predictions by including the analysis data. Note the diagonal structure in Fig. 11, which outlines the impact of the EAM forecasts on the prediction quality, meaning the EAM forecasts of the day i have the most significant impact on the LOD predictions around day i, as expected. Nevertheless, the EAM forecast data on the following days also impact the LOD prediction of previous days. This effect can also be observed by comparing the results using 10-day and 6-

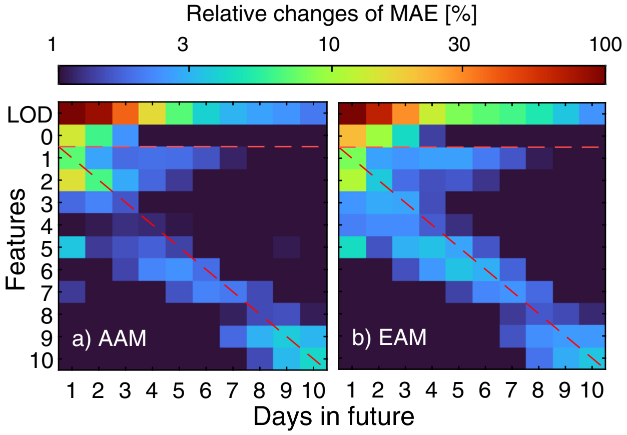

Fig. 11 Relative change of MAE w.r.t. changing 30% of the different features on a logarithmic scale. The numbers of the y-axes show the AAM (**a**), and EAM (**b**) forecasts of the corresponding days, whereas 0 indicates analysis data

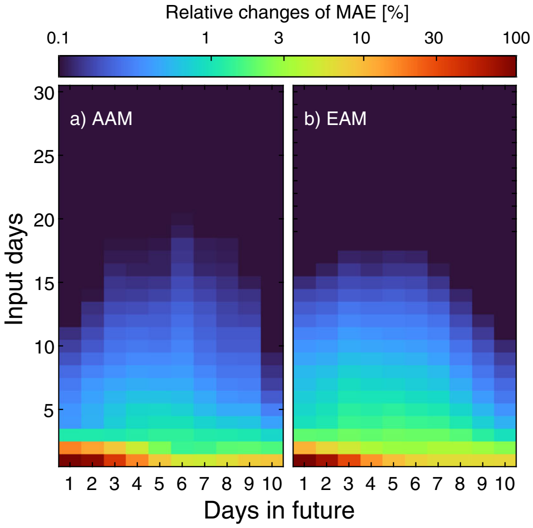

Fig. 12 Relative change of MAE w.r.t. changing 30% of all features on different input time steps on a logarithmic scale

day EAM forecasts in aforementioned results such as Fig. 9: Including EAM forecasts of latter days do not only improve the accuracy of the predictions after day six but also affect the predictions in the first six days positively. This finding is counterintuitive since there cannot exist a causal relationship between the EAM forecasts on latter days and LOD on earlier days. The reason is that the EAM forecasts are fed into the networks as separate features, while the LSTM networks do not consider the sequence between different features. Therefore, the networks benefit from the high correlation between input features. In view of causality, the observations of earlier days do not depend on the EAM of the next days; however, they are highly correlated, and the network can take advantage of this effect. Therefore, information from the future can also be helpful for the networks to predict the previous days.

For a second run, we changed 30% of all features on individual input days and computed the relative change of MAE to study the impact of different input days, shown in Fig. 12 on a logarithmic scale. The features on the last input day dominate the prediction process since this input day contains the most recent information. The context is the last output of the encoder (Fig. 4), where the inputs on the last day only go through the nonlinear activation functions once. Therefore, the outputs are most sensitive to the last input epoch. Due to the existence of the forget gate in LSTM neurons, the high-frequency variations of inputs on previous epochs are more likely discarded, and only the major signals are restored. Therefore, we observe that the predictions are not sensitive to the 30% change in inputs from more than one week ago. However, there is still a visible impact of inputs from a week ago on the predictions of days 3–6, which may roughly indicate the known 14-day periods of the LOD time series.

5 Conclusions

In order to accommodate the need for real-time EOPs of the highest quality in geodetic applications, precise prediction of EOPs, especially in the ultra-short term, is necessary. In this study, we employed one of the most popular variants of RNN called LSTM to predict LOD for the next ten days, in accordance with the definition of ultra-short-term in the first EOP PCC. To obtain highly accurate predictions, we implemented a hybrid modeling approach, which combines physical, statistical, and deep learning modeling. First, we corrected the tidal effects and estimated the deterministic components (long-term trend, seasonal variations), and removed them from the LOD time series to generate LOD residuals. We also removed the seasonal signals from the geophysical excitations (including forecasts) to generate EAM and AAM residuals. Both LOD and EAM/AAM residual time series were considered as the inputs of the LSTM neural networks. Compared with the state-of-the-art studies, we achieved significant improvements up to 43% under the comparable conditions of the first EOP PCC and reduced the MAE further by including more data. Overall, accurate ultra-short-term LOD predictions with MAE below 0.08 ms are achieved in the realistic scenario. We investigated the predicting accuracy without considering the data latency. In this scenario, our method achieves MAE under 0.06 ms by day ten. Moreover, we discussed the contributions of atmospheric, oceanic and hydrological systems to LOD as well as the impact of their forecasting horizon. We conclude that the atmospheric contributions are the most dominant, while including the oceanic and hydrological systems also has clear benefits, with an additional MAE reduction of around 10%. The drawback of the EAM time series is its higher forecasting error level inherited from the three components (AAM, OAM, HAM), which impedes its contribution in the latter days. The feature analysis shows that the most crucial feature is LOD itself with clear contributions of EAM forecasts. An interesting fact is that the forecasts on latter days also impact predictions of previous days. With this finding, we can explain the reason for the better predictions on the first days while including longer EAM forecast data. Besides, this result also indicates the potential improvement by including more extended EAM forecast data, even after the 10th day. However, the rapidly decreasing quality of EAM forecast data hinders this approach.

In this study, we proved the advantages of LSTM neural networks for LOD prediction and the benefits of large data volumes for this approach. With the help of LSTM neural networks, we can provide reliable LOD predictions in the ultra-short term operationally, from which many real-time geodetic and geophysical applications can benefit. We conclude that more attention should be paid to including the geophysical excitations in the machine-learning-based LOD prediction framework. The higher errors caused by the summation of the atmospheric, oceanic and hydrological systems should be investigated further. This study also confirms the benefits of accurate EAM forecasts to LOD predictions, which should also apply to other EOPs, such as polar motion. We identified two needs in further improving EOP predictions: First, more attention should be paid to employing and testing other machine learning approaches for predicting EOPs. In this regard, suitable techniques may vary among different EOPs due to the different characteristics of the data. Second, improving the quality of rapid EOP products is also crucial, since the operational prediction must rely on them. In the end, more accurate EAM forecasts with a more extended forecasting horizon also have the potential to improve the quality of EOP predictions significantly.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00190-023-01745-x.

Author Contributions Junyang Gou implemented the experiments, did the analysis and prepared the original draft. Mostafa Kiani Shahvandi provided the 10-day EAM forecasts, helped with the methodology and prepared the original draft. Roland Hohensinn and Benedikt Soja proposed the topic, supervised the study and revised the draft. All authors worked on the theoretical considerations, discussed the results, and contributed to the final manuscript.

Funding Open access funding provided by Swiss Federal Institute of Technology Zurich.

Data Availability The used EOPs data are available at the IERS FTP server (ftp://ftp.iers.org/products/eop/). The used EAM data are available at the GFZ FTP server (ftp://esmdata.gfz-potsdam.de/EAM/). The 14-day EAM forecasts as well as the LOD predictions using the EDLSTM network provided by ETH Zurich can be found at the Geodetic Prediction Center (https://gpc.ethz.ch/). All the other datasets and models generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare that they do not have any conflict of interest.

Appendix A: ETH 14-day EAM forecasts

Both the accuracy and prediction horizon of EAM forecasts are of great importance to improve the quality of EOP predictions (Kiani Shahvandi et al. 2022b; Kur et al. 2022). We developed a method based on the concept of Neural Ordinary Differential Equations (Neural ODE, Chen et al. 2018) to forecast EAM up to 14 days ahead. A recurrent neural network, whose hidden state follows a first-order ordinary differential equation, is fitted to the EAM values. The inputs to the network are the historical observations and forecasts of EAM provided by GFZ. We trained 18 individual models for the different components (x, y, z), and mass and motion terms of AAM, OAM, and HAM plus three models for SLAM, since it only contains mass terms. The input sequence contains two epochs: the current time epoch t and the previous epoch $t-1$. At each time epoch t, we have seven features, namely the GFZ analysis products (o_t) and 6-day forecasts $f_{t+1:t+6}$. The output sequence length is 14, corresponding to the forecasts at epochs $t+1 : t+14$. Therefore, the following model can be represented as the hidden states of the algorithm:

$$\mathbf{h}_t = \mathrm{M}\left(o_t, o_{t-1}, \mathbf{f}_t, \mathbf{f}_{t-1}, \mathbf{W}_1\right) \tag{10}$$

$$\frac{\mathrm{d}\mathbf{h}_{t+1}}{\mathrm{d}t} = \mathrm{RNN}\left(\mathbf{h}_t, \mathbf{W}_2\right) \tag{11}$$

$$\mathbf{r} = \mathbf{W}_3 \cdot \mathbf{h}_{t+1} + \mathbf{b}_3 \tag{12}$$

where $\mathbf{h}_t$ and $\mathbf{h}_{t+1}$ denote the hidden states at epochs t and $t+1$, respectively. The hidden state $\mathbf{h}_t$ can be obtained from the two previous epochs with analysis data (o_t, o_{t-1}) and 6-day forecast data $(\mathbf{f}_t, \mathbf{f}_{t-1})$ by considering a transformation M with weights $\mathbf{W}_1$. We note that each forecast vector

$\mathbf{f}_t$ contains six components $f_{t,1:6}$ corresponding to the 6-day forecasts. The RNN cell with parameters $\mathbf{W}_2$ describes the differential equation. In the end, the resulting forecasting vector $\mathbf{r}$, including the 14-day forecasts at time epochs $t+1$ to $t+14$, can be obtained by a linear transformation from the hidden states $\mathbf{h}_{t+1}$ with parameters $\mathbf{b}_3$, $\mathbf{W}_3$.

The operational quality control at GPC shows that ETH EAM forecasts are slightly more accurate than GFZ EAM forecasts in the first six days. On days 7 to 14, the predictions show good agreement with the corresponding observations, but the forecasting errors accumulate more rapidly. In our experiments, we observed that the EAM forecasts after day 10 negatively impacted the LOD predictions due to their higher noise level. As a result, we only use the first ten days of EAM forecasts in this study. The RMSDs used to generate the pseudo-forecasts are shown in Table 1 of the supplementary information.

Appendix B: Preprocessing using the Savitzky–Golay filter

In this study, we choose the SG filter to model and remove the secular trend in the LOD time series. For the time series from 1985 to 2015, we use the frame length of 3653 with order 2. However, in the first experiment using data from 1996 to 2005, the length of the LOD time series is shorter and the secular trend within around ten years is easier to model. Thus, we use a quadratic function to model the secular trend for EOP PCC instead of the SG filter.

To preprocess the test set, we always concatenate the ’new’ data with the training and validation set, and rerun the preprocessing process. However, there is a boundary issue of the SG filter in the last windows since the number of data is smaller than the length of window, see Fig. 13. Therefore, the resulting LOD residuals sometimes differ from the EAMz residuals. To solve this issue, we estimate the difference of Δ_2 and EAMz within the input interval (30 days) linearly by applying one more least-squares adjustment. Then, we remove this difference from Δ_2 and add this in Δ_S to generate Δ_2' and Δ_S' as the following:

$$\Delta_2' = \Delta_2 - b \tag{13}$$

$$\Delta_S' = \Delta_S + b \tag{14}$$

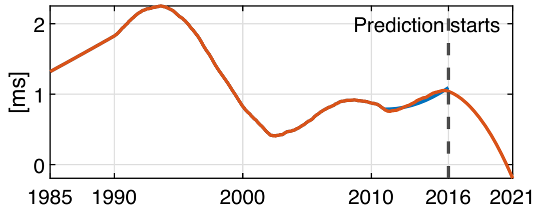

Fig. 13 The boundary issue of the SG filter. The blue line shows the real estimated secular trend based on data until the beginning of 2015, and the red line shows the theoretical secular trend based on data until the end of 2020

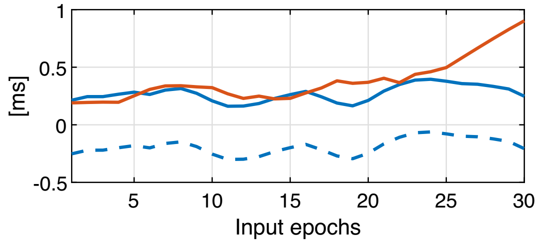

Fig. 14 One example from the test set. The blue dashed line shows the LOD residuals without linear offset correction (Δ_R). The blue line shows the LOD residuals after removing the linear offset (Δ_R'). The red line shows the EAMz residuals

where b is the estimated linear offset between Δ_2 and EAMz. Then, we can estimate the seasonal signals based on Δ_2' for the test dataset and generate the corresponding Δ_R'. In the end, the LOD residuals and EAMz residuals are highly correlated. Figure 14 shows an example of the test set in the second experiment. The dashed blue line shows Δ_R, which is the original LOD residuals without any linear offset correction. Since we can detect a significant linear offset between Δ_R and EAMz residuals, the LSTM neural networks cannot provide correct predictions relying on these two features. However, after removing the linear offset, we get Δ_R' (blue line), which is highly correlated with the EAMz residuals. The remaining differences are relatively small and mainly caused by data noise and the remaining low-frequency Earth’s interior signals, which are potentially insufficiently modeled by the SG filter. Therefore, the LSTM neural networks can provide accurate predictions based on Δ_R' and EAMz residuals.

References

Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mané D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viégas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X (2015) TensorFlow: large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/ software available from tensorflow.org

Akyilmaz O, Kutterer H, Shum C, Ayan T (2011) Fuzzy-wavelet based prediction of Earth rotation parameters. Appl Soft Comput 11:837–841. https://doi.org/10.1016/j.asoc.2010.01.003

Altamimi Z, Rebischung P, Métivier L, Collilieux X (2016) ITRF2014: a new release of the International Terrestrial Reference Frame modeling nonlinear station motions. J Geophys Res: Solid Earth 121:6109–6131. https://doi.org/10.1002/2016JB013098

Bergen KJ, Johnson PA, de Hoop MV, Beroza GC (2019) Machine learning for data-driven discovery in solid Earth geoscience. Science. https://doi.org/10.1126/science.aau0323

Bizouard C, Gambis D (2009) The combined solution C04 for Earth orientation parameters consistent with international terrestrial reference frame 2005. In: Geodetic reference frames. Springer, pp 265–270. https://doi.org/10.1007/978-3-642-00860-3_41

Bizouard C, Lambert S, Gattano C, Becker O, Richard J-Y (2019) The IERS EOP 14C04 solution for Earth orientation parameters consistent with ITRF 2014. J Geodesy 93:621–633. https://doi.org/10.1007/s00190-018-1186-3

Brockwell PJ, Davis RA (2002) Introduction to time series and forecasting. Springer, Berlin. https://doi.org/10.1007/978-3-319-29854-2

Chen RTQ, Rubanova Y, Bettencourt J, Duvenaud DK (2018) Neural ordinary differential equations. In: Advances in neural information processing systems, vol 31. Curran Associates, Inc. https://proceedings.neurips.cc/paper_filespaper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf

Dehant V, Mathews P (2015) 3.10—earth rotation variations. In: Schubert G (ed) Treatise on geophysics, 2nd edn. Elsevier, Oxford, pp 263–305. https://doi.org/10.1016/B978-0-444-53802-4.00066-X

Dill R, Dobslaw H, Thomas M (2019) Improved 90-day Earth orientation predictions from angular momentum forecasts of atmosphere, ocean, and terrestrial hydrosphere. J Geodesy 93:287–295. https://doi.org/10.1007/s00190-018-1158-7

Dill R, Saynisch-Wagner J, Irrgang C, Thomas M (2021) Improving atmospheric angular momentum forecasts by machine learning. Earth Sp Sci 8:e2021EA002070. https://doi.org/10.1029/2021EA002070

Dobslaw H, Dill R (2018) Predicting Earth orientation changes from global forecasts of atmosphere-hydrosphere dynamics. Adv Sp Res 61:1047–1054. https://doi.org/10.1016/j.asr.2017.11.044

Dobslaw H, Dill R (2019) Effective angular momentum functions from Earth system modelling at GeoForschungsZentrum in Potsdam. Technical Report Technical Report, Revision 1.1 (March 18, 2019), GFZ Potsdam, Germany

Donahue J, Anne Hendricks L, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K, Darrell T (2015) Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2625–2634

Freedman A, Steppe J, Dickey J, Eubanks T, Sung L-Y (1994) The short-term prediction of universal time and length of day using atmospheric angular momentum. J Geophys Res: Solid Earth 99:6981–6996. https://doi.org/10.1029/93JB02976

Fritsch FN, Carlson RE (1980) Monotone piecewise cubic interpolation. SIAM J Numer Anal 17:238–246

Gers FA, Schmidhuber J, Cummins F (2000) Learning to forget: continual prediction with LSTM. Neural Comput 12:2451–2471. https://doi.org/10.1162/089976600300015015

Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT Press

Gross R (2007) 3.09—Earth rotation variations—long period. Treat Geophys. https://doi.org/10.1016/B978-044452748-6.00057-2

Gross RS, Eubanks T, Steppe J, Freedman A, Dickey J, Runge T (1998) A Kalman-filter-based approach to combining independent Earth-orientation series. J Geodesy 72:215–235. https://doi.org/10.1007/s001900050162

Hamdan K, Sung L-Y (1996) Stochastic modeling of length of day and universal time. J Geodesy 70:307–320. https://doi.org/10.1007/BF00868182

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9:1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Holme R, De Viron O (2013) Characterization and implications of intradecadal variations in length of day. Nature 499:202–204. https://doi.org/10.1038/nature12282

Kalarus M, Schuh H, Kosek W, Akyilmaz O, Bizouard C, Gambis D, Gross R, Jovanović B, Kumakshev S, Kutterer H et al (2010) Achievements of the Earth orientation parameters prediction comparison campaign. J Geodesy 84:587–596. https://doi.org/10.1007/s00190-010-0387-1

Kiani Shahvandi M, Gou J, Schartner M, Soja B (2022a) Data driven approaches for the prediction of Earth's effective angular momentum functions. In: 2022 IEEE international geoscience and remote sensing symposium IGARSS. IEEE, pp 6550–6553. https://doi.org/10.1109/IGARSS46834.2022.9883545

Kiani Shahvandi M, Schartner M, Soja B (2022) Neural ODE differential learning and its application in polar motion prediction. J Geophys Res: Solid Earth. https://doi.org/10.1029/2022JB024775

Kiani Shahvandi M, Soja B (2021) Small geodetic datasets and deep networks: attention-based residual LSTM autoencoder stacking for geodetic time series. In: International conference on machine learning, optimization, and data science. Springer, pp 296–307. https://doi.org/10.1007/978-3-030-95467-3_22

Kiani Shahvandi M, Soja B (2022) Inclusion of data uncertainty in machine learning and its application in geodetic data science, with case studies for the prediction of Earth orientation parameters and GNSS station coordinate time series. Adv Sp Res 70:563–575. https://doi.org/10.1016/j.asr.2022.05.042

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980

Kitpracha C, Modiri S, Asgarimchr M, Heinkelmann R, Schuh H (2019) Machine Learning based prediction of atmospheric zenith wet delay: a study using GNSS measurements in Wettzell and co-located VLBI observations. In: EGU general assembly conference abstracts, p 4127

Kosek W (2012) Future improvements in EOP prediction. In: Geodesy for planet Earth. Springer, pp 513–520. https://doi.org/10.1007/978-3-642-20338-1_62

Kosek W, Kalarus M, Johnson T, Wooden W, McCarthy D, Popinski W (2005) A comparison of LOD and UT1-UTC forecasts by different combined prediction techniques. Artif Satell 40:119–125

Kosek W, McCarthy D, Luzum B (1998) Possible improvement of Earth orientation forecast using autocovariance prediction procedures. J Geodesy 72:189–199. https://doi.org/10.1007/s001900050160

Kur T, Dobslaw H, Śliwińska J, Nastula J, Wińska M, Partyka A (2022) Evaluation of selected short-term predictions of UT1-UTC and LOD collected in the second earth orientation parameters prediction comparison campaign. Earth, Planets Sp 74:1–9. https://doi.org/10.1186/s40623-022-01753-9

Lei Y, Guo M, Hu D-D, Cai H-B, Zhao D-N, Hu Z-P, Gao Y-P (2017) Short-term prediction of UT1-UTC by combination of the grey model and neural networks. Adv Sp Res 59:524–531. https://doi.org/10.1016/j.asr.2016.10.030

Lei Y, Zhao D, Cai H (2015) Prediction of length-of-day using extreme learning machine. Geodesy Geodyn 6:151–159. https://doi.org/10.1016/j.geog.2014.12.007

Liao D, Wang Q, Zhou Y, Liao X, Huang C (2012) Long-term prediction of the earth orientation parameters by the artificial neural network technique. J Geodyn 62:87–92. https://doi.org/10.1016/j.jog.2011.12.004

Liu Y, Dang B, Li Y, Lin H, Ma H (2016) Applications of Savitzky–Golay filter for seismic random noise reduction. Acta Geophys 64:101–124. https://doi.org/10.1515/acgeo-2015-0062

Miao K-C, Han T-T, Yao Y-Q, Lu H, Chen P, Wang B, Zhang J (2020) Application of LSTM for short term fog forecasting based on meteorological elements. Neurocomputing 408:285–291. https://doi.org/10.1016/j.neucom.2019.12.129

Mikschi M, Böhm J, Böhm S, Horozovic D (2019) Comparison of Integrated GNSS LOD to dUT1. In: Proceedings of the 24th European VLBI group for geodesy and astrometry working meeting, vol. 4. Chalmers University of Technology, pp 247–251

Modiri S, Belda S, Hoseini M, Heinkelmann R, Ferrándiz JM, Schuh H (2020) A new hybrid method to improve the ultra-short-term prediction of LOD. J Geodesy 94:1–14. https://doi.org/10.1007/s00190-020-01354-y

Nayak T, Ng HT (2020) Effective modeling of encoder-decoder architecture for joint entity and relation extraction. In: Proceedings of the AAAI conference on artificial intelligence, vol 34, pp 8528–8535

Niedzielski T, Kosek W (2008) Prediction of UT1-UTC, LOD and AAM χ 3 by combination of least-squares and multivariate stochastic methods. J Geodesy 82:83–92. https://doi.org/10.1007/s00190-007-0158-9

Oppenheim AV, Schafer RW (2013) Discrete-time signal processing. Pearson Education India

Petit G, Luzum B (2010) IERS Conventions (2010). Frankfurt am Main: Verlag des Bundesamts für Kartographie und Geodäsie

Pu F, Xu Z, Chen H, Xu X, Chen N (2018) A DLM-LSTM framework for north-south land deformation trend analysis from low-cost GPS sensor time series. J Sens. https://doi.org/10.1155/2018/3054295

Reichstein M, Camps-Valls G, Stevens B, Jung M, Denzler J, Carvalhais N et al (2019) Deep learning and process understanding for data-driven Earth system science. Nature 566:195–204. https://doi.org/10.1038/s41586-019-0912-1

Roy IG (2020) An optimal Savitzky–Golay derivative filter with geophysical applications: an example of self-potential data. Geophys Prospect 68:1041–1056. https://doi.org/10.1111/1365-2478.12892

Rumelhart DE, Hinton GE, Williams RJ (1985) Learning internal representations by error propagation. Technical Report California University San Diego La Jolla Inst for Cognitive Science

Savitzky A, Golay MJ (1964) Smoothing and differentiation of data by simplified least squares procedures. Anal Chem 36:1627–1639. https://doi.org/10.1021/ac60214a047

Schafer RW (2011) What is a Savitzky–Golay filter? [Lecture notes]. IEEE Signal Process Mag 28:111–117. https://doi.org/10.1109/MSP.2011.941097

Schuh H, Ulrich M, Egger D, Müller J, Schwegmann W (2002) Prediction of Earth orientation parameters by artificial neural networks. J Geodesy 76:247–258. https://doi.org/10.1007/s00190-001-0242-5

Senior K, Kouba J, Ray J (2010) Status and prospects for combined GPS LOD and VLBI UT1 measurements. Artif Satell 45:57–73. https://doi.org/10.2478/v10018-010-0006-7

Shi X, Chen Z, Wang H, Yeung D-Y, kin Wong W, Chun Woo W (2015) Convolutional LSTM network: a machine learning approach for precipitation nowcasting. arXiv:1506.04214

Soja B, Kiani Shahvandi M, Schartner M, Gou J, Kłopotek G, Crocetti L, Awadaljeed M (2022) The new geodetic prediction center at ETH Zurich. In: EGU general assembly conference abstracts, pp EGU22–9285. https://doi.org/10.5194/egusphere-egu22-9285

Sovers OJ, Fanselow JL, Jacobs CS (1998) Astrometry and geodesy with radio interferometry: experiments, models, results. Rev Mod Phys 70:1393–1454. https://doi.org/10.1103/RevModPhys.70.1393

Wang Q, Liao D, Zhou Y (2008) Real-time rapid prediction of variations of Earth's rotational rate. Chin Sci Bull 53:969–973. https://doi.org/10.1007/s11434-008-0047-5

Xu X, Zhou Y, Liao X (2012) Short-term earth orientation parameters predictions by combination of the least-squares, AR model and Kalman filter. J Geodyn 62:83–86. https://doi.org/10.1016/j.jog.2011.12.001

Zhang K, Geng X, Yan X-H (2020) Prediction of 3-D ocean temperature by multilayer convolutional LSTM. IEEE Geosci Remote Sens Lett 17:1303–1307. https://doi.org/10.1109/LGRS.2019.2947170