The Astrophysical Journal, 977:19 (12pp), 2024 December 10 https://doi.org/10.3847/1538-4357/ad8b20

21CMLSTM: A Fast Memory-based Emulator of the Global 21 cm Signal with Unprecedented Accuracy

J. Dorigo Jones[1], S. M. Bahauddin[2], D. Rapetti[1,3,4], J. Mirocha[5,6], and J. O. Burns[1]
[1] Center for Astrophysics and Space Astronomy, Department of Astrophysical and Planetary Sciences, University of Colorado Boulder, CO 80309, USA; johnny.dorigojones@colorado.edu
[2] Laboratory for Atmospheric and Space Physics, University of Colorado, Boulder, CO 80303, USA
[3] NASA Ames Research Center, Moffett Field, CA 94035, USA
[4] Research Institute for Advanced Computer Science, Universities Space Research Association, Washington, DC 20024, USA
[5] Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA
[6] California Institute of Technology, 1200 E. California Boulevard, Pasadena, CA 91125, USA
Received 2024 August 14; revised 2024 September 20; accepted 2024 October 8; published 2024 November 28

Abstract

Neural network (NN) emulators of the global 21 cm signal need an emulation error much less than the observational noise in order to be used to perform unbiased Bayesian parameter inference. To this end, we introduce `21cmLSTM`—a long short-term memory (LSTM) NN emulator of the global 21 cm signal that leverages the intrinsic correlation between frequency channels to achieve exceptional accuracy compared to previous emulators, which are all feedforward, fully connected NNs. LSTM NNs are a type of recurrent NN designed to capture long-term dependencies in sequential data. When trained and tested on the same simulated set of global 21 cm signals as the best previous emulators, `21cmLSTM` has an average relative rms error of 0.22%—equivalently 0.39 mK—and comparably fast evaluation time. We perform seven-dimensional Bayesian parameter estimation analyses using `21cmLSTM` to fit global 21 cm signal mock data with different adopted observational noise levels, σ_{21}. The posterior 1σ rms error is ≈three times less than σ_{21} for each fit and consistently decreases for tighter noise levels, showing that `21cmLSTM` can sufficiently exploit even very optimistic measurements of the global 21 cm signal. We have made the emulator, code, and data sets publicly available so that `21cmLSTM` can be independently tested and used to retrain and constrain other 21 cm models.

1. Introduction

Neutral hydrogen (H I) emits radiation at 1420.4 MHz ($\lambda \approx 21$ cm) via the spin-flip transition that coupled with the gas kinetic temperature during the Epoch of Reionization (EoR; ending by $z \approx 6$), Cosmic Dawn ($10 \lesssim z \lesssim 30$), and Dark Ages ($z > 30-40$; for reviews, see S. R. Furlanetto et al. 2006; A. Bera et al. 2023). As a result, the differential brightness temperature of the 21 cm line with respect to the cosmic microwave background, δT_b, is expected to be a powerful probe of the astrophysics and cosmology of each of these cosmic epochs. Numerous low-frequency radio experiments ($\nu \lesssim 225$ MHz, corresponding to $z \gtrsim 5.3$) have pursued measurements of the sky-averaged (i.e., global; P. A. Shaver et al. 1999) 21 cm signal (J. D. Bowman et al. 2018; S. Singh et al. 2018, 2022; E. de Lera Acedo et al. 2022), as well as its power spectrum (G. Paciga et al. 2011; F. G. Mertens et al. 2020; C. M. Trott et al. 2020; H. Garsden et al. 2021; HERA Collaboration et al. 2023), although systematic effects, mainly from the galactic foreground in combination with beam chromaticity and radio frequency interference (RFI), have so far prevented a clear detection of the global signal (e.g., R. Hills et al. 2018; R. F. Bradley et al. 2019; P. H. Sims & J. C. Pober 2020; K. Tauscher et al. 2020). Efforts are also underway to measure the Dark Ages 21 cm signal from the Moon (e.g., LuSEE-Night; S. D. Bale et al. 2023).

To constrain the physical parameters able to describe the global 21 cm signal, Bayesian inference is a powerful tool (e.g., G. Bernardi et al. 2016; A. Liu & J. R. Shaw 2020; D. Rapetti et al. 2020; E. Shen et al. 2022). Likelihood-based inference techniques such as Markov Chain Monte Carlo (MCMC) and nested sampling (J. Skilling 2004) are used to numerically estimate the parameters of models from data and constrain the full joint posterior distribution (e.g., C. J. Schmit & J. R. Pritchard 2018; J. Mirocha & S. R. Furlanetto 2019; R. A. Monsalve et al. 2019; H. T. J. Bevins et al. 2022a, 2024). Bayesian inference can require 10^6 or more model evaluations to fully search the prior volume and calculate the posterior, which can become exceedingly computationally expensive, especially when constraining many parameters and jointly fitting for different systematics.

Machine learning, in the form of artificial neural networks (NNs), can be employed to mimic the physical models of interest being sampled in a Bayesian fitting analysis and efficiently obtain converged posteriors. Through supervised learning of labeled data generated by the physical model, networks can be taught the relationship between the input parameters and the output (in this case, δT_b) to quickly and accurately emulate the model. Emulation error on the order of 1 mK can result in significantly biased posteriors even when fitting global 21 cm signal mock data with statistical noise of 25 mK (J. Dorigo Jones et al. 2023), and so emulation error <1 mK is needed to sufficiently exploit optimistic or standard measurements of the 21 cm signal and obtain unbiased posteriors.

Long short-term memory (LSTM; S. Hochreiter & J. Schmidhuber 1997; F. A. Gers et al. 2000) networks are a type of recurrent NN (RNN), which differ from feedforward networks such as fully connected NNs (FCNNs; e.g., D. E. Rumelhart et al. 1986), also called multilayer perceptrons, and convolutional NNs. In short, FCNNs have a one-directional flow of information that is not specifically designed for temporal

awareness, while RNNs have feedback connections that allow them to learn trends or sequences of features in data (e.g., Y. LeCun et al. 2015). LSTM networks have been successful in numerous temporal prediction and classification problems in astrophysics (H. Liu et al. 2019; L. Hu et al. 2022; Z. Sun et al. 2022; A. Iess et al. 2023; S. S. Tabasi et al. 2023; Y. Zheng et al. 2023; S. Huber & S. H. Suyu 2024; J. I.-H. Li et al. 2024), although their ability to emulate a physical, numerical model is relatively unexplored (R. Zhang et al. 2020).

So far, LSTM networks or RNNs have not been utilized to emulate the global 21 cm signal or any summary statistic in 21 cm cosmology. At the time of writing this paper, there exist four publicly available NN-based emulators of the global 21 cm signal—21CMGEM (A. Cohen et al. 2020), `globalemu` (H. T. J. Bevins et al. 2021), `21cmVAE` (C. H. Bye et al. 2022), and `21cmEMU` (D. Breitman et al. 2024)—all of which use FCNNs to predict δT_b as a function of the independent variable, being redshift or frequency, given (seven to nine) input astrophysical parameters. In this paper, we present a novel LSTM-based emulator of the global 21 cm signal, called `21cmLSTM`, which exploits the intrinsic correlation of information between adjacent frequency channels (i.e., auto-correlation) in 21 cm data to achieve unprecedented emulation accuracy. D. Prelogović et al. (2022) found a similar benefit of LSTM RNNs, but as a regressor for 21 cm 3D lightcones when used with convolutional layers (see X. Shi et al. 2015; D. Kodi Ramanah et al. 2022).

For detailed descriptions of RNNs and LSTM cells, see, e.g., R. C. Staudemeyer & E. R. Morris (2019) and A. Sherstinsky (2020); here, we provide a conceptual overview. The "hidden state" is the key element of RNNs, which reuses the same weights and biases on each step and updates them via back propagation through time (BPTT; R. J. Williams & D. Zipser 1995). Information is fed through the RNN sequentially, and the hidden state output from each step is used to inform the output of all future steps. Basic RNNs are limited to predicting ∼10 time steps, though, because of the "vanishing gradient" problem, whereby the back-propagated error either vanishes or explodes as more weights are multiplied together (e.g., R. Pascanu et al. 2013). LSTM cells were invented to avoid this problem by incorporating a "memory cell internal state," or "information highway," which enforces constant error flow. LSTM cells contain forget, input, and output gates that determine the relative importance of each time step and ensure the gradient can bridge 1000 or more steps without vanishing, thereby helping to identify both short-term and long-term correlations in data. For a single-layer (i.e., nonstacked) LSTM network, the number of activation operations is the data resolution (i.e., the number of channels or bins), and so hyperparameter optimization relies purely on determining the best number of layers of nonlinear activation and the number of training epochs, whereas FCNNs contain an additional dimension to optimize, being the number of nodes per hidden layer.

The paper is organized as follows: in Section 2, we describe the architecture and training of 21CMLSTM; in Section 3, we present the emulation accuracy and speed of 21CMLSTM; in Section 4, we present the posterior constraints when using 21CMLSTM in a Bayesian nested sampling analysis fitting mock data; and in Section 5, we summarize the conclusions.

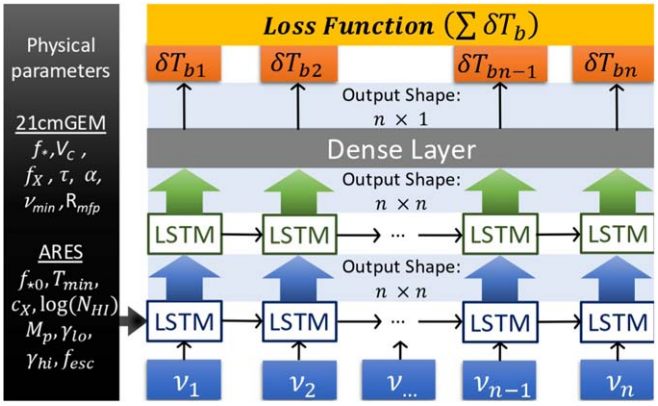

Figure 1. Schematic diagram of 21CMLSTM network architecture. The user inputs the physical parameter values for the desired model, and the emulator predicts δT_b for all frequencies. The arrows indicate inputs to or outputs of layers and LSTM cells. The input array is (N, n, p), where N is the number of signals, n is the number of frequency channels per signal, and p is the number of physical parameters plus one for the frequency channel. For emulating the 21CMGEM and `ARES` training sets, (N, n, p) is (24,562, 451, 8) and (23,896, 449, 9), respectively. See Section 2.1 for further details.

2. Methods

In this section, we describe the components of 21CMLSTM, including the network architecture, the data sets used for training, validation, and testing, the data preprocessing steps, the training settings, and the optimization performed to ensure robust and accurate emulation results. The emulator is written in PYTHON, using the KERAS (F. Chollet et al. 2015) machine learning libraries with a TENSORFLOW (M. Abadi et al. 2015) backend. The code[7] and data[8] are both publicly available, making 21CMLSTM simple to use and retrain.

2.1. Architecture

The emulator model is composed of two LSTM layers (i.e., two layers of nonlinear activation, equivalent to two hidden layers in an FCNN), followed by a dense layer with output dimensionality of one. Figure 1 shows a schematic diagram of the 21CMLSTM network architecture, with arrows indicating connections between layers or between LSTM cells. The emulator takes as input the physical parameters, which are user-defined, along with the list of frequencies, which is initialized within the emulator, and outputs the brightness temperature, δT_b, for all frequencies. The emulator creates a 3D input array, (N, n, p), for the first LSTM layer, where N is the number of signals, n is the number of frequency channels in each signal, and p is the number of physical parameters plus one for the frequency channel. The LSTM cells are "many-to-many," meaning each cell predicts the entire signal sequence, and so each LSTM layer has output dimensionality equal to the number of frequency channels. As mentioned, the second LSTM layer is connected to a fully connected output layer that predicts δT_b for each frequency channel, which is used to calculate the loss during BPTT.

The LSTM layers use hyperbolic tangent (tanh) activation function, and the output layer uses linear activation. The model uses the Adam stochastic gradient descent optimization method

[7] doi:10.5281/zenodo.13916935 (J. Dorigo Jones & S. Bahauddin 2024) and https://github.com/jdorigojones/21cmLSTM.

[8] doi:10.5281/zenodo.5084114 (A. Cohen et al. 2021); doi:10.5281/zenodo.13840725.

Table 1
Astrophysical Parameters Varied in 21CMGEM and ARES Data Sets and Fit in Nested Sampling Analyses

Model	Parameter	Description	Range (with Units)
21CMGEM	f_*	SFE	Log unif. $[10^{-4}, 5 \times 10^{-1}]$
	V_c	Minimum circular velocity of star-forming halos	Log unif. [4.2, 100] km s^{-1}
	f_X	X-ray efficiency of sources	Log unif. $[10^{-6}, 10^{3}]$
	τ	Cosmic microwave background optical depth	Uniform [0.04, 0.2]
	α	Slope of X-ray spectral energy distribution (SED)	Uniform [1, 1.5]
	ν_{min}	Low-energy cutoff of X-ray SED	Uniform [0.1, 3] keV
	R_{mfp}	Mean free path of ionizing radiation	Uniform [10, 50] Mpc
ARES	$f_{*,0}$	Peak SFE	Log unif. $[10^{-5}, 10^{0}]$
	T_{min}	Minimum temperature of star-forming halos	Log unif. $[3 \times 10^{2}, 5 \times 10^{5}]$ K
	c_X	Normalization of L_X–SFR relation	Log unif. $[10^{36}, 10^{44}]$ erg s^{-1}($M_\odot$ yr^{-1})$^{-1}$
	log$N_{H\,I}$	Neutral hydrogen column density in galaxies	Uniform [18, 23]
	M_p	Dark matter halo mass at $f_{*,0}$	Log unif. $[10^{8}, 10^{15}]$ $M_\odot$
	γ_{lo}	Low-mass slope of $f_*(M_h)$	Uniform [0, 2]
	γ_{hi}	High-mass slope of $f_*(M_h)$	Uniform [−4, 0]
	f_{esc}	Escape fraction of ionizing radiation	Uniform [0, 1]

(D. Kingma & J. Ba 2015) and mean squared error (MSE) loss function:

$$\mathrm{MSE} = \langle (\delta T_b(\nu) - \delta \hat{T}_b(\nu))^2 \rangle, \quad (1)$$

where $\delta \hat{T}_b(\nu)$ is the emulated signal produced by 21CMLSTM, and $\delta T_b(\nu)$ is the simulated, "true" signal produced by the model on which the emulator is trained. We performed "hyperparameter tuning" to minimize the prediction error, by testing one LSTM layer, three LSTM layers, different activation functions, and the mean absolute error loss function, and found the choices stated above result in the most accurate network on average. We also trained a one-layer Bidirectional LSTM (Bi-LSTM) model, which is an LSTM network trained in both directions, and found that it performs only slightly worse than the two-layer LSTM.

2.2. Data Sets

We train and test 21CMLSTM on the exact same publicly available set of global 21 cm signals used to originally train and test the previous emulators 21CMGEM, globalemu, and 21cmVAE. The data set was created (see A. Cohen et al. 2020 for a description) by a seminumerical model (E. Visbal et al. 2012; A. Fialkov et al. 2013, 2014) that is similar to 21CMFAST (A. Mesinger et al. 2011). We refer to this data set as the 21CMGEM set, in which seven astrophysical parameters are varied (Table 1; see A. Cohen et al. 2020) and each signal spans the redshift range $z = 5-50$ with resolution $\delta z = 0.1$. We apply the same parameter range restrictions and observational constraints as stated in A. Cohen et al. (2020), H. T. J. Bevins et al. (2021), and C. H. Bye et al. (2022) to obtain 24,562 training signals, 2730 validation signals, and 1704 test signals. A representative subset of the 21CMGEM combined training +validation set is shown in the top left panel of Figure 2.

We also train and test 21CMLSTM on a different data set generated by another popular model for the global 21 cm signal, Accelerated Reionization Era Simulations (ARES;[9] J. Mirocha 2014; J. Mirocha et al. 2017), which is a physically motivated semianalytical code that is the union of a 1D radiative transfer code (J. Mirocha et al. 2012) and a uniform radiation background code (J. Mirocha 2014). We created the ARES set to be nearly equivalent to the 21CMGEM set, in order to directly compare the accuracy of 21CMLSTM between the two models, with: (i) the same size of the test set (1704) and a similar (to within 3%) size of the combined training+validation set (26,552, also split 90% for training and 10% for validation); (ii) eight (instead of seven) astrophysical parameters varied over wide ranges (see Table 1), which also control the star formation efficiency (SFE) and ionizing photon production in galaxies, although via a different parameterization (see Section 3.3 for a comparison between the two sets); (iii) the same redshift resolution and nearly identical range ($z = 5.1-49.9$); and (iv) a similar physical EoR constraint on the neutral hydrogen fraction ($x_{H\,I}$) at $z < 6$; we require $x_{H\,I} < 5\%$ at $z = 5.3$, while the 21CMGEM set requires $x_{H\,I} < 16\%$ at $z = 5.9$, based on less recent constraints (see, e.g., X. Fan et al. 2006; I. D. McGreer et al. 2015; C. A. Mason et al. 2019; S. E. I. Bosman et al. 2022; Y. Zhu et al. 2022; X. Jin et al. 2023). A representative subset of the ARES combined training +validation set is shown in the bottom left panel of Figure 2.

Before the emulator is trained, the training and validation data are preprocessed to be normalized between zero and one, which is usual to facilitate network performance. Some of the physical parameters are uniform only in log 10-space, and so the log 10 is taken of these parameters: f_X, V_c, and $f_\star$ for 21CMGEM, and c_X, T_{min}, $f_{*,0}$, and M_p for ARES. We note that $f_X = c_X/2.6 \times 10^{39}$ erg s^{-1}($M_\odot$ yr^{-1})$^{-1}$. The signals (i.e., δT_b labels and frequency list) are flipped so that the network is trained from high-z to low-z. Finally, we performed a min–max normalization (Equation (2)) on each feature, x, in the data (i.e., physical parameter values and the list of frequencies) and labels (i.e., δT_b):

$$\tilde{x} = \frac{x - x_{min}}{x_{max} - x_{min}}. \quad (2)$$

We found that normalizing the labels bin-by-bin per signal caused the preprocessed signals to blow up at frequencies with little variation (i.e., a small denominator in Equation (2)). Therefore, the min–max normalization is performed globally for the signal labels in order to preserve their original smooth shape, and for consistency the same is done when normalizing the data.

[9] https://github.com/mirochaj/ares

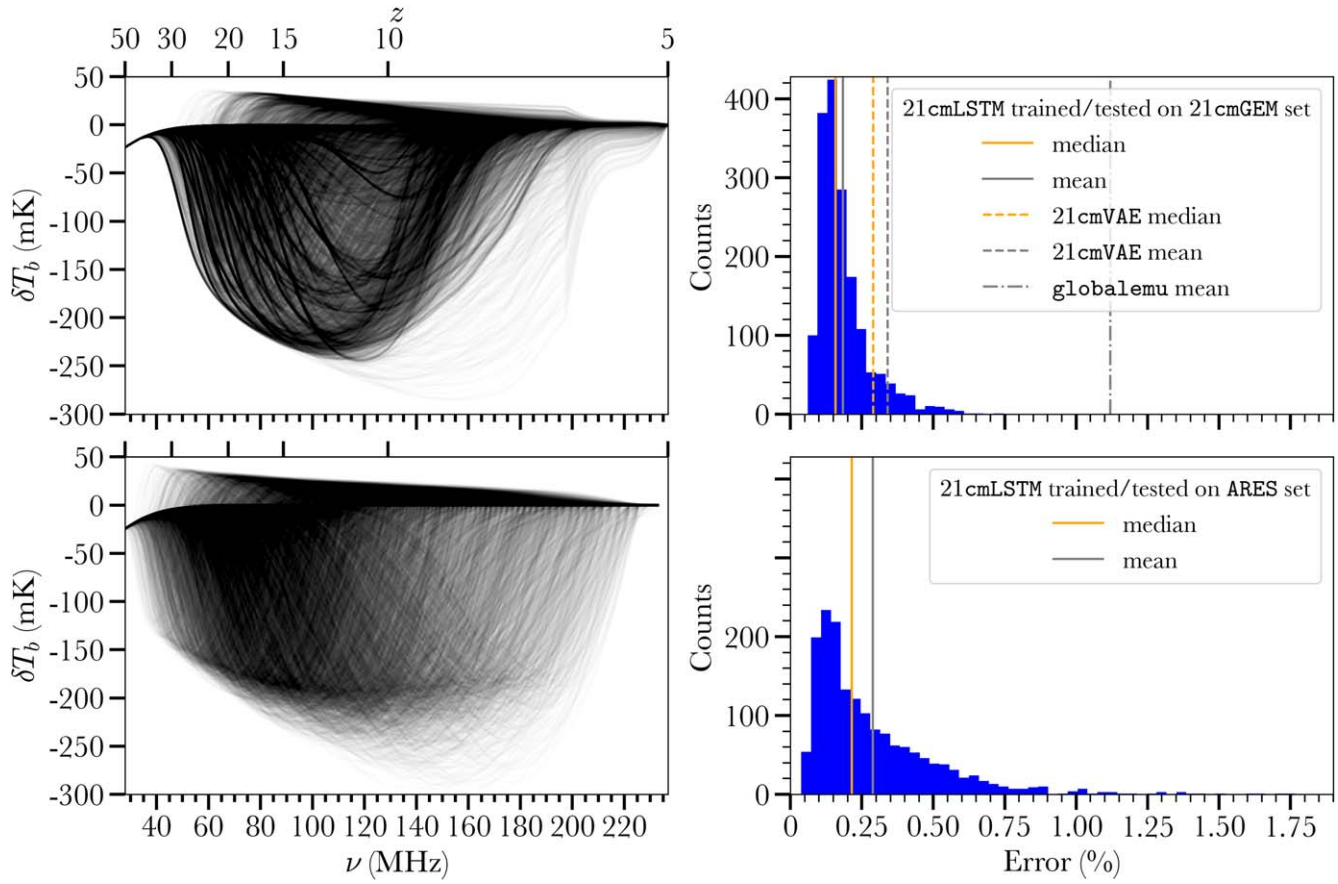

Figure 2. Left: model realizations of 10,000 global 21 cm signals randomly drawn from the 21CMGEM (top) and ARES (bottom) combined training+validation sets (Section 2.2). The ARES set has 1.3 times more statistical outliers and a higher PCA error (see Section 3.3), which is consistent with more signal variation and consequently larger emulation error (Section 3.1). Right: histograms of the relative rms error (Equation (3)) for the best trial of 21CMLSTM, trained on the 21CMGEM (top) and ARES (bottom) training sets and evaluated on the 1704 signals in each test set. The vertical gray (orange) lines depict the mean (median) error for 21CMLSTM (solid), 21CMVAE (dashed), and globalemu (dashed–dotted).

The emulator is trained on the preprocessed training set signals and saves at each epoch the MSE loss of the network evaluated on the training and validation sets. Only the training set errors are used during BPTT to update the network weights, while the validation set is used to gauge the emulator's ability to generalize to unseen signals and to check for overfitting. The test set, which is created separately from the training and validation sets, determines the ultimate accuracy of the trained instance of 21CMLSTM.

2.3. Training

For the results presented in this work, we trained and tested 21CMLSTM using a single NVIDIA A100 GPU with 32 CPU cores on the Blanca shared "condo" compute cluster operated by University of Colorado Research Computing. The emulator is trained first for 75 epochs with a batch size of 10 (i.e., training on batches of 10 signals at a time), then for 25 epochs with a batch size of one, then finally for another 75 epochs with a batch size of 10. The final saved network loads the model weights and biases from the final epoch of training. Training with a large batch size before and after a smaller batch size (i.e., batch size scheduling; see S. L. Smith et al. 2017) is an increasingly common alternative to decaying the learning rate, which facilitates robust gradient descent and speeds up the overall training time, which is an average of 12.4 hr $\pm$ 0.1 hr (utilizing $\approx$6 GB of memory) when training 21CMLSTM on the 21CMGEM set. We tested different batch sizes between 1 and 32 and found the ones stated above ensured the model learns efficiently, generalizes well to unseen data, and makes effective use of computational resources.

Instead of incorporating an early stopping condition for the training, we determined the approximate number of training epochs that produces the most accurate and robust resulting network on average. We trained and tested 21CMLSTM on the 21CMGEM set for 20, 25, 30, and 40 epochs of batch size one, running six trials for each, and find that they have average relative rms errors (see Equation (3) below) of 0.34%, 0.24%, 0.44%, and 0.29%, respectively. We performed this testing with both 75 epochs and 100 epochs of batch size 10 before and after the epochs of batch size one, and we find marginal difference between the two. Therefore, since 25 epochs of training with batch size one, with 75 epochs of batch size 10 before and after, produced the most accurately trained 21CMLSTM, and with no spurious outlier trials, we employ this training epoch configuration. We find that the validation loss curves reach a stable solution near the end of training (see Figure A1), rather than increasing, which indicates that there is no overfitting.

3. Emulation Results

3.1. Accuracy

We report the emulation accuracy of 21CMLSTM when trained and tested on the same sets of global 21 cm signals that

Table 2
Accuracy and Speed Metrics of Global 21 cm Signal Emulators

Emulator	Mean Error (%)	Maximum Error (%)	Speed (ms)
21CMLSTM	0.22	0.82	46
21CMVAE	0.35	1.84	74
globalemu	1.12	6.32	3
21CMGEM	1.59	10.55	160

Note. The information provided for each emulator is the average mean and maximum rms errors across the full frequency range of ≈1700 test signals (see Section 3.1) and the average evaluation speed when predicting one signal at a time (see Section 3.2). The errors quoted for other emulators are from their respective original papers (A. Cohen et al. 2020; H. T. J. Bevins et al. 2021; C. H. Bye et al. 2022). For direct comparison purposes, the speeds quoted for the first three emulators were measured using the same computational resources stated in Section 2.3, while we note that the speeds measured in the original papers for 21CMVAE (C. H. Bye et al. 2022) and globalemu (H. T. J. Bevins et al. 2021) are 41.4 ms and 1.3 ms, respectively. The speed quoted for 21CMGEM is from its original paper (A. Cohen et al. 2020).

were used in the original papers for 21CMVAE (C. H. Bye et al. 2022), globalemu (H. T. J. Bevins et al. 2021), and 21CMGEM (A. Cohen et al. 2020), allowing a direct comparison to these existing emulators (see Table 2). We also train and test 21CMLSTM on an equivalent data set created by ARES (see Section 2.2).

Using the optimized network architecture and training settings described in Section 2, we trained 20 identical trials of 21CMLSTM on the 21CMGEM set in order to characterize the stochasticity of the training algorithm. We evaluated each trained network at the parameter values of the 1704 signals in the 21CMGEM test set and compared the resulting emulated signals to their corresponding "true" signals, computing for each signal the rms error across the full frequency range in both absolute units (millikelvins), and relative units (percent) as:

$$\text{Error} = \frac{\sqrt{\text{MSE}}}{\max(|\delta T_b(\nu)|)}, \tag{3}$$

where MSE is defined by Equation (1) and $\max(|\delta T_b(\nu)|)$ is the signal amplitude.

The distribution of the mean relative rms error for all 20 trials is shown in Figure 3. Across the 20 trials, 21CMLSTM has an average relative mean error of $0.223\% \pm 0.031\%$ (corresponding to an average absolute error of $0.389\,\text{mK} \pm 0.047\,\text{mK}$), an average median error of $0.197\% \pm 0.025\%$, and an average maximum error of $0.824\% \pm 0.183\%$. The best trial (top right panel of Figure 2) has a mean relative error of 0.18% (corresponding to an absolute error of 0.30 mK), a median error of 0.16% (corresponding to 0.26 mK), and a maximum error of 0.75% (corresponding to 1.34 mK). Therefore, when trained and tested on the same data for the same number of trials, 21CMLSTM has a 1.6 times lower average error and ≈two times lower maximum error than those reported for 21CMVAE (see Table 2).

When trained and tested on the equivalent ARES sets, the best trial (bottom right panel of Figure 2) has a mean relative error of 0.29% (corresponding to 0.42 mK), a median error of 0.21%, and a max error of 1.77% (see Section 3.3).

3.2. Speed

We report the emulation speed of 21CMLSTM as the average time to predict a single global 21 cm signal in the 21CMGEM test set (i.e., predict δT_b for all $n = 451$ frequencies) from the seven input physical parameters or the emulator evaluation time including steps for data preprocessing (see Section 2.2) and signal denormalization (see Equation (2)). We employed the same computational resources stated in Section 2.3 and used the time module to measure the total processing time, which we note naturally depends on the computing power (e.g., number and type of CPU cores and GPUs). We report the speed for a single evaluation[10] for proper benchmarking with other emulators, as some architectures are inherently more conducive to parallel processing than RNNs, which are serial in nature.

Across 20 trials, the average emulation speed of 21CMLSTM is 46 ms, which is similar to those reported for other emulators of the global 21 cm signal, except for globalemu (H. T. J. Bevins et al. 2021), which was designed to be faster. We performed the same timing test using the latest versions of 21cmVAE (C. H. Bye et al. 2022) and globalemu (H. T. J. Bevins et al. 2021) and measured their average speeds to be 74 ms and 3 ms, respectively (see Table 2). The speed and unprecedented accuracy of 21CMLSTM make it capable of efficient Bayesian multiparameter estimation, as we carry out in Section 4, which reflects the success of a two-layer LSTM RNN in emulating 21 cm models.

3.3. Model Comparison

In Section 3.1, we found that 21CMLSTM performs somewhat better when trained and tested on the 21CMGEM data sets than on the ARES sets, with a best trial mean relative error of 0.183% compared to 0.288% (right panel of Figure 2). We remind the reader that the 21CMGEM data set was created by a large-volume seminumerical model (E. Visbal et al. 2012; A. Fialkov et al. 2013, 2014), similar to 21CMFAST (A. Mesinger et al. 2011; see A. Cohen et al. 2020), while ARES is a semianalytical model that does not calculate 3D volumes. The difference in performance is likely caused by differences in the parameterizations and parameter ranges between the models, which can be qualitatively compared in Table 1. In particular, a single parameter for the SFE ($f_\star$) is varied in the 21CMGEM sets, while four SFE parameters ($f_{\star,0}$, M_p, γ_{lo}, and γ_{hi}, which describe a double power law) are varied in the ARES sets, which may result in a smaller range of cosmic star formation histories and thus less variation among the signals in the 21CMGEM sets compared to ARES (see the left panel of Figure 2).

We briefly investigated the differences between the two combined training+validation sets by performing a principal component analysis (PCA) decomposition of each set. By default, we set the number of components extracted equal to the number of features or the number of physical parameters varied in each data set (i.e., seven for 21CMGEM and eight for ARES). We calculate the Mahalanobis distance (P. C. Mahalanobis 2018) for each signal, which is a common metric used for multidimensional outlier detection and defined between two points u and v as $d = \sqrt{(u - v)(1/V)(u - v)^T}$, where $(1/V)$ is the inverse covariance. We define outliers as those signals with $d > 3$, meaning they are $>3\sigma$ from the sample mean vector. We find that 32% (8499) of the ARES set are outliers, while 25% (6800) of the 21CMGEM set are outliers (shown in red in Figure 4). This statistical analysis is consistent with the larger variation in the ARES set, causing 21CMLSTM to have a higher

[10] This uses the eval_21cmGEM.py script on the GitHub (see the first footnote link).

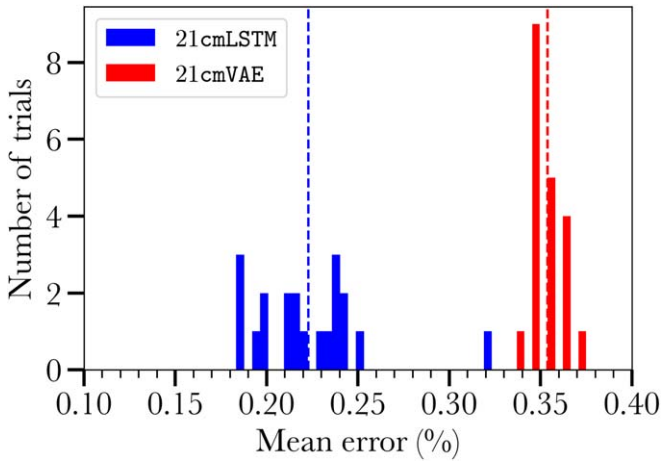

Figure 3. Histogram of mean relative error for 20 trials of 21CMLSTM (in blue) trained and tested on the 21CMGEM data set. The red histogram is the approximate error for 20 trials of 21CMVAE trained and tested on the same data, adapted from Figure 6 of C. H. Bye et al. (2022). The dashed blue (red) line depicts the average error for 21CMLSTM (21CMVAE).

emulation error when trained and tested on ARES compared to the 21CMGEM set. Beyond the visual comparison of model features offered by Figure 4, we leave for future work a detailed study of the similarities and differences between these two popular models of the global 21 cm signal.

4. Posterior Emulation

In this section, we use 21CMLSTM as the model in the likelihood of Bayesian nested sampling analyses to fit mock global 21 cm signals with added statistical noise and numerically estimate seven astrophysical parameters. We describe the steps to obtain converged posterior distributions, and we present the signal posterior constraints obtained by the emulator compared to the fiducial signal for three 21 cm noise levels. Note that this analysis ignores systematic uncertainties from the beam-weighted foreground (see, e.g., G. Bernardi et al. 2016; J. J. Hibbard et al. 2020, 2023; D. Anstey et al. 2023; P. H. Sims et al. 2023; M. Pagano et al. 2024; A. Saxena et al. 2024), RFI (Y. Shi et al. 2022; S. A. K. Leeney et al. 2023), and environmental effects (see, e.g., S. Singh et al. 2018; N. S. Kern et al. 2020; N. Bassett et al. 2021; S. G. Murray et al. 2022; E. Shen et al. 2022; J. H. N. Pattison et al. 2024).

4.1. Bayesian Inference Analysis

We perform Bayesian parameter inference analyses to numerically estimate the posterior distribution $P(\theta|\boldsymbol{D}, m)$ of a set of parameters θ in a physical model m, given observed (mock) data $\boldsymbol{D}$ with priors π on the parameters. Bayes' theorem states this as:

$$P(\theta|\boldsymbol{D}, m) = \frac{\mathcal{L}(\theta)\pi(\theta)}{Z}, \tag{4}$$

where $\mathcal{L}$ is the likelihood function and Z is the Bayesian evidence, which can be used for model comparison. We sample from a multivariate log-likelihood function assuming Gaussian-distributed noise:

$$\log \mathcal{L}(\theta) \propto [\boldsymbol{D} - m(\theta)]^T \boldsymbol{C}^{-1} [\boldsymbol{D} - m(\theta)], \tag{5}$$

where the noise covariance, $\boldsymbol{C}$, is a diagonal array of constant values corresponding to the square of the estimated noise σ_{21}.

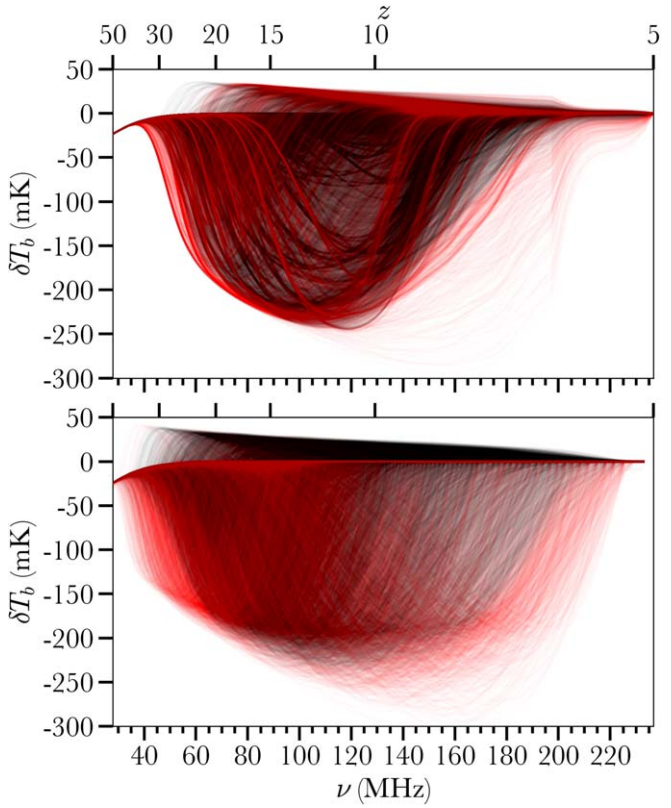

Figure 4. Top: 15,000 signals randomly selected from the 21CMGEM combined training+validation set with PCA outliers shown in red (see Section 3.3). Bottom: the same as the top panel, but for ARES.

We employ the Bayesian inference method of nested sampling (J. Skilling 2004; for reviews, see G. Ashton et al. 2022; J. Buchner 2023), which converges on the best parameter estimates by iteratively removing regions of the prior volume with lower likelihood and computes the evidence and posterior samples simultaneously. As mentioned in the introduction, Monte Carlo methods like nested sampling and MCMC are computationally expensive because they require many likelihood evaluations to sample the multidimensional posterior, and so model emulators are desired to speed up or make feasible such analyses. We choose nested sampling rather than MCMC because the former is designed to constrain parameter spaces with complex degeneracies or multimodal distributions (J. Buchner 2023), which are expected when fitting 21 cm mock or real data (e.g., H. T. J. Bevins et al. 2022b; J. Dorigo Jones et al. 2023; D. Breitman et al. 2024; also see A. Saxena et al. 2024). For all analyses, we employ MultiNest (F. Feroz & M. P. Hobson 2008; F. Feroz et al. 2009, 2019), with default evidence tolerance and sampling efficiency and three times the default initial "live" point number (1200), which we find results in consistent, converged posteriors. For an in-depth description of MultiNest and other algorithms, see, e.g., P. Lemos et al. (2023).

We use 21CMLSTM trained on the 21CMGEM set as the model for the global 21 cm signal in the likelihood. The trained instance of 21CMLSTM used for all analyses has a mean rms error of 0.20% (corresponding to 0.33 mK) and a maximum error of 0.63% when evaluated on the 21CMGEM test set, which is consistent with the average accuracy found in Section 3.1. We evaluate this emulator's ability to constrain a synthetic 21 cm signal with added statistical noise.

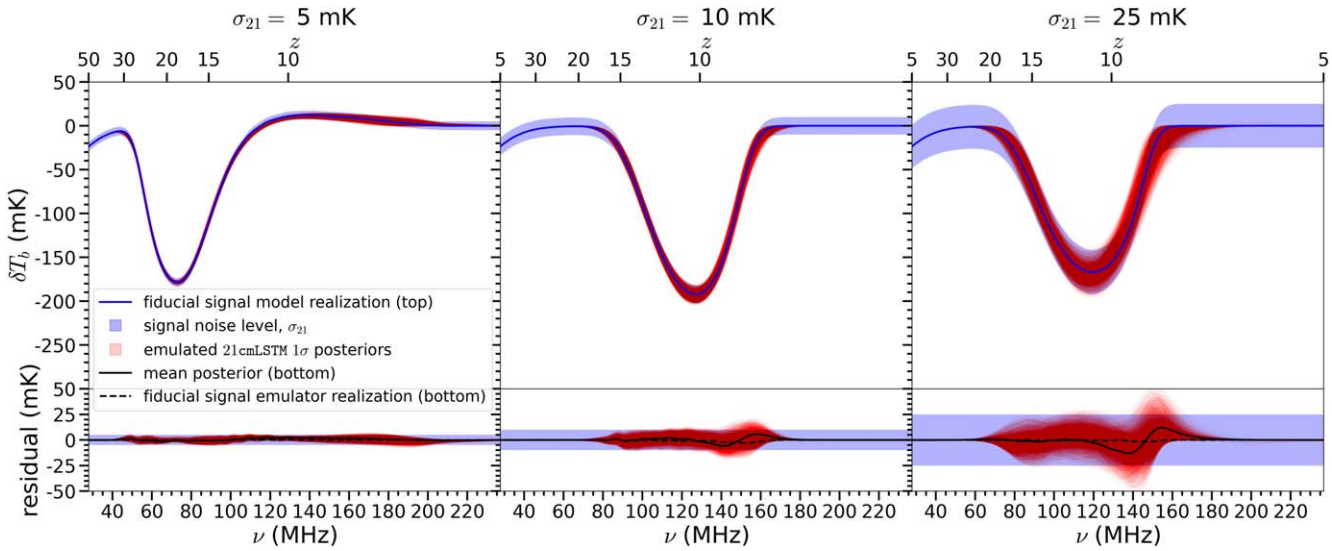

Figure 5. Top: signal realizations of the 1σ posterior samples (red; see Section 4.2) obtained from Bayesian nested sampling analyses using 21CMLSTM to fit three fiducial global 21 cm signals (dark blue) randomly selected from the 21CMGEM test set with added Gaussian-distributed noise (light blue bands) of 5 mK (left), 10 mK (middle), and 25 mK (right). Bottom: residuals between the corresponding true fiducial signal and each 21CMLSTM 1σ posterior (red; see Table 3), the posterior mean (solid black), and the emulation of the fiducial signal (dashed black).

We fit three different fiducial mock global 21 cm signals that are randomly selected from the 21CMGEM test set and have different levels of added statistical white noise that is Gaussian-distributed. The three different 21 cm noise levels tested include the optimistic and fiducial scenarios for the REACH radiometer (E. de Lera Acedo et al. 2022): $\sigma_{21} = 5$ mK or 10 mK (referred to as "optimistic") and $\sigma_{21} = 25$ mK (referred to as "standard"), where σ_{21} is the standard deviation noise estimate. We note that in J. Dorigo Jones et al. (2023), we tested these same noise levels, as well as 50 and 250 mK, and compared emulated posteriors obtained using `globalemu` (H. T. J. Bevins et al. 2021) to the corresponding "true" posteriors obtained using `ARES`. From the ideal radiometer sensitivity equation (e.g., J. Kraus 1966) for a non-systematics-limited 21 cm experiment, assuming $\nu = 30$ MHz (i.e., Dark Ages) and $\Delta\nu = 0.5$ MHz, the noise levels 5 mK, 10 mK, and 25 mK correspond to integration times of ≈7100 hr, ≈1800 hr, and ≈300 hr, respectively.

4.2. Posterior Results

In the top panels of Figure 5, we present sets of posterior signal realizations (shown in red) when using 21CMLSTM to fit three different mock 21 cm signals (shown in dark blue), with added noise levels (shown in light blue) of 5, 10, and 25 mK, from left to right. We show the 1σ posteriors in red, defined as the 68% of samples with the lowest relative rms error with respect to the fiducial signal (Equation (3)). In the bottom panels of Figure 5, we present the residuals between the fiducial signal and each 1σ posterior sample (i.e., $\delta T_b(\nu) - \delta\hat{T}_b(\nu)$; red), the mean of all the posterior samples (solid black), and the emulator realization of the fiducial signal (i.e., $m_{\texttt{21cmLSTM}}(\theta_0)$; dashed black). Table 3 summarizes each fit.

We find that for each 21 cm noise level tested, the posterior mean residual is significantly less than the signal noise estimate, σ_{21}, across the full redshift range and approaches the emulator error (0.33 mK; see Section 4.1) as the noise level decreases. This is seen visually in the bottom panels of Figure 5, as well as quantitatively in the mean and 1σ (i.e., 68th percentile) rms errors of the posteriors for each fit (see $\sigma_{\text{posterior}}$ in Table 3). The mean relative rms error (Equation (3)) between all the emulated posteriors and the true fiducial signal is 0.87% (corresponding to 1.56 mK absolute error) for $\sigma_{21} = 5$ mK, 1.42% (corresponding to 2.73 mK) for $\sigma_{21} = 10$ mK, and 4.11% (corresponding to 6.87 mK) for $\sigma_{21} = 25$ mK. The posterior mean and 1σ errors are thus each ≈three times less than σ_{21} for each fit. The fit obtained using 21CMLSTM consistently improves for decreasing 21 cm noise levels, corresponding to longer integration times, as expected due to the increase in constraining power. We note that this general trend of a more accurate fit to the mock signal for decreasing noise levels is robust, as it does not depend on the random signals being fit.

Table 3
Summary of Nested Sampling Analyses

σ_{21} (mK)	n_{live}	$n_{\text{evaluations}}$	f_{accept}	$\log Z$	$\sigma_{\text{posterior}}$ (mK)
5	1200	125,232	0.202	-255.9 ± 0.1	1.6
10	1200	64,106	0.340	-254.6 ± 0.1	2.9
25	1200	46,784	0.369	-251.1 ± 0.1	7.9

Note. The information provided for each fit is the noise level of the mock 21 cm signal (σ_{21}), the number of initial live points (n_{live}), the total number of likelihood evaluations ($n_{\text{evaluations}}$), the final acceptance rate (f_{accept}), the final evidence ($\log Z$), and the posterior 1σ rms error ($\sigma_{\text{posterior}}$). The analysis methods and results are described in Sections 4.1 and 4.2, respectively. The 1σ posterior signal realizations and residuals with respect to the true fiducial signal are shown in Figure 5, and the full posterior distributions for $\sigma_{21} = 25$ mK and $\sigma_{21} = 5$ mK are shown in Figures B1 and B2, respectively.

In addition to the posterior signal realizations discussed, we can examine the marginalized 1D and 2D posterior distributions. We present the full posterior parameter distribution for the $\sigma_{21} = 25$ mK fit in Figure B1 and for the $\sigma_{21} = 5$ mK fit in Figure B2. For the standard noise level tested (i.e., $\sigma_{21} = 25$ mK), we find that the 1D posteriors for three

astrophysical parameters (f_*, V_c, and τ) are well constrained and unbiased with respect to (i.e., within 2σ of) their fiducial values, while the other four parameters (f_X, α, ν_{min}, and R_{mfp}) are relatively unconstrained. For the optimistic noise levels tested (i.e., $\sigma_{21} = 5$ and 10 mK), our findings are similar, although the constraints improve somewhat, in particular for f_X and ν_{min}, reflecting the improved signal posterior realizations seen in Figure 5.

These results demonstrate that, as a result of its low emulation error, 21CMLSTM can sufficiently exploit even outstandingly optimistic measurements of the global 21 cm signal and obtain unbiased posterior constraints. We find that, from our non-systematics-limited global 21 cm mock data analysis, we obtain unbiased posterior constraints when the emulator error is $\approx 1\%-5\%$ of the signal observational noise level, σ_{21}. These ratio values are based on the emulation rms errors for the fiducial mock signals (black dashed lines in the bottom panels of Figure 5) fit with $\sigma_{21} = 25$ mK and $\sigma_{21} = 5$ mK, which are 0.31 mK and 0.24 mK, respectively. Furthermore, these results are consistent with other comparable Bayesian analyses of mock 21 cm data that have found that jointly fitting complementary summary statistics or data sets is needed to break the degeneracies between certain astrophysical parameters (e.g., Y. Qin et al. 2020; A. Chatterjee et al. 2021; H. T. J. Bevins et al. 2023; J. Dorigo Jones et al. 2023; D. Breitman et al. 2024).

5. Conclusions

Achieving unbiased Bayesian parameter inference of the global 21 cm signal using an NN emulator requires the emulation error to be much lower than the observational noise on the signal (e.g., J. Dorigo Jones et al. 2023). Highly accurate and fast emulation is therefore needed to sufficiently exploit optimistic or standard measurements of the 21 cm signal, especially to approach the cosmic variance limit of $\sim$0.1 mK in the future (J. B. Muñoz & F.-Y. Cyr-Racine 2021). To this end, in this paper, we have presented a new emulator of the global 21 cm signal, called 21CMLSTM, which is an LSTM RNN that has exceptionally low emulation error compared to existing emulators, which are all FCNNs. 21CMLSTM owes its unprecedented accuracy to its unique ability to leverage the intrinsic (spatiotemporal) correlation of information between neighboring frequency channels in the global 21 cm signal.

In Section 2, we optimized 21CMLSTM by testing different architectures (i.e., the number of LSTM layers, activation functions, loss function, and Bi-LSTM models), data preprocessing steps (i.e., normalizations), and training configurations (i.e., number of epochs and batch sizes). A schematic diagram of the network architecture of 21CMLSTM is shown in Figure 1. In Section 3, we presented the emulation accuracy of 21CMLSTM when trained and tested on large data sets created by two different popular models of the global 21 cm signal (see Figure 2). Finally, in Section 4, we employed a representative instance of 21CMLSTM, trained on a 21CMGEM set, as the model in the likelihood of a Bayesian nested sampling analysis to fit mock signals and showed that it can be used to obtain unbiased posterior constraints.

When trained and tested on the same data as existing emulators, 21CMLSTM has an average relative rms error of $(0.22 \pm 0.03)\%$ (Figure 3), corresponding to (0.39 ± 0.05) mK, and a best trial mean error of 0.18% (top right panel of Figure 2), corresponding to 0.30 mK. 21CMLSTM therefore has a $\approx$1.6 times lower average error than the previously most accurate emulator of the global 21 cm signal, `21cmVAE` (Table 2). The maximum emulation error of 21CMLSTM is 0.82% on average, which is $\approx$two times lower than that reported for `21cmVAE`. Furthermore, 21CMLSTM has a similar emulation speed as other existing emulators when predicting one signal at a time (Table 2, Section 3.2), making it both sufficiently fast and accurate for complex, high-dimensional Bayesian parameter estimation analyses. We also examined a set of 21 cm signals created by the `ARES` model with a greater parameter variation than the 21CMGEM set (Figure 4, Section 3.3) and found, as might be expected, that 21CMLSTM produces a somewhat higher emulation error when trained on this `ARES` set (bottom right panel of Figure 2).

We obtained accurate posterior distributions when using 21CMLSTM in `MultiNest` analyses to fit mock global 21 cm signals with added observational noise levels of $\sigma_{21} = 5$ mK, $\sigma_{21} = 10$ mK, and $\sigma_{21} = 25$ mK. The full parameter posterior distributions for the $\sigma_{21} = 25$ mK and $\sigma_{21} = 5$ mK fits are presented in Figures B1 and B2, respectively, and the posterior signal realizations and residuals for all fits are shown in Figure 5. The posteriors provide a good fit to each fiducial mock signal, with the posterior mean and 1σ errors being $\approx$three times less than the respective adopted signal noise level, σ_{21} (see the bottom panel of Figure 5, Table 3, Section 4.2). The posterior mean residual consistently decreases as the signal noise level decreases, with the $\sigma_{21} = 5$ mK fit having a posterior mean relative rms error of only 0.87% (corresponding to 1.56 mk) compared to the pure emulation average error of 0.20% (corresponding to 0.33 mK), for the instance of 21CMLSTM employed. For all three noise levels tested, the posterior distributions are well converged and unbiased for three of seven parameters (f_*, V_c, and τ), and for the lowest noise level (i.e., $\sigma_{21} = 5$ mK), the posteriors become unbiased for two more parameters (f_X and ν_{min}). These results are consistent with recent findings that jointly fitting complementary summary statistics or data sets is needed to constrain certain astrophysical parameters (e.g., Y. Qin et al. 2020; A. Chatterjee et al. 2021; H. T. J. Bevins et al. 2023; J. Dorigo Jones et al. 2023; D. Breitman et al. 2024).

This work demonstrates that LSTM RNNs exploit the intrinsic correlation of adjacent frequency channels (i.e., autocorrelation) in the global 21 cm signal to perform very accurate and fast emulation of physically motivated seminumerical or semianalytical models of the signal. We have made the data sets and code publicly available on Zenodo (see footnotes 7 and 8) and GitHub (see footnote 7), respectively, so that the 21CMLSTM emulator can be used and modified by the community. In principle, 21CMLSTM could also be adapted to learn the pattern of and predict any sequential or time-series measurement, assuming sufficient data size and resolution, and subsequently be employed in Bayesian analyses. The publicly available emulator 21CMLSTM contributes to the growing body of astrophysics and cosmology research finding that, for data or measurements with intrinsic correlation over time, LSTM RNNs can perform as well as or better than FCNNs.

Acknowledgments

We thank the anonymous reviewer for the thorough feedback that improved the manuscript. We thank Christian H. Bye, Harry T. J. Bevins, and Joshua Hibbard for useful discussions. This

work utilized the Blanca condo computing resource at the University of Colorado Boulder. Blanca is jointly funded by computing users and the University of Colorado Boulder. This work was directly supported by the NASA Solar System Exploration Research Virtual Institute cooperative agreement 80ARC017M0006. We acknowledge support by NASA APRA grant award 80NSSC23K0013 and a subcontract from UC Berkeley (NASA award 80MSFC23CA015) to the University of Colorado (subcontract #00011385) for science investigations involving the LuSEE-Night lunar farside mission. J.M. was supported by an appointment to the NASA Postdoctoral Program at the Jet Propulsion Laboratory/California Institute of Technology, administered by Oak Ridge Associated Universities under contract with NASA. Part of this work was done at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004).

Software: This research relies heavily on the PYTHON (G. Van Rossum & F. L. Drake 1995) open-source community libraries NUMPY (C. R. Harris et al. 2020), MATPLOTLIB (J. D. Hunter 2007), SCIPY (P. Virtanen et al. 2020), TENSORFLOW (M. Abadi et al. 2015), and KERAS (F. Chollet et al. 2015). This research also utilized JUPYTER (T. Kluyver et al. 2016), `MultiNest` (F. Feroz et al. 2009, 2019), `21cmVAE` (C. H. Bye et al. 2022), and `globalemu` (H. T. J. Bevins et al. 2021).

Appendix A
Loss Curves for Validation and Training Sets

Figure A1 shows the distribution of MSE loss (Equation (1)) for the training and validation sets for 20 identical trials of 21CMLSTM trained on the 21CMGEM data (see Sections 2.3 and 3.1). The validation loss for each trial reaches a stable value, which indicates that the network is able to generalize to unseen signals and is not overfitting the training set, whereas increasing validation loss would indicate overfitting.

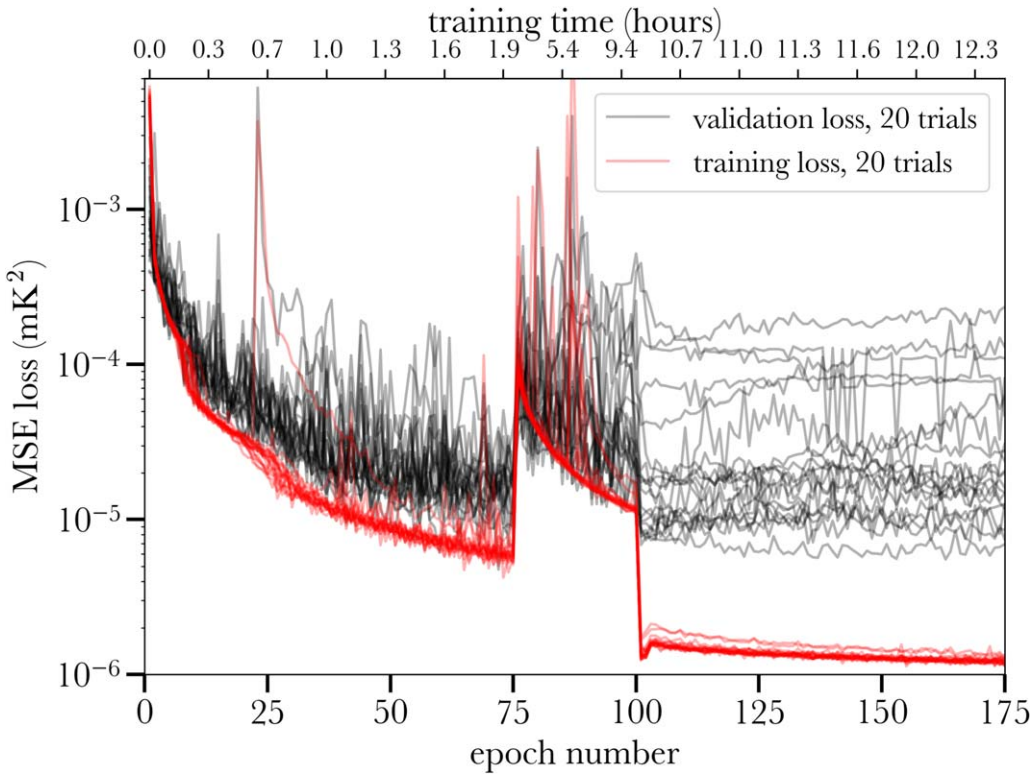

Figure A1. Loss vs. training epoch number for validation (black) and training (red) sets for 20 trials of 21CMLSTM trained on the 21CMGEM data. The top axis shows the approximate training time at each epoch. Note that the emulator is trained for 75 epochs with a batch size of 10, before and after training for 25 epochs with a batch size of one (see the batch scheduling description in Section 2.3).

Appendix B
Posterior Distributions from Fitting Mock Global 21 cm Signals with 25 mK and 5 mK Noise

In Figure B1, we present the full 1D and 2D marginalized posterior distribution for seven astrophysical parameters obtained when using 21CMLSTM in a nested sampling analysis to fit a mock global 21 cm signal randomly selected from the 21CMGEM test set with a standard observational noise level of $\sigma_{21} = 25$ mK. In Figure B2, we present the posterior distribution when fitting a different randomly selected signal fit with $\sigma_{21} = 5$ mK. The 1σ posterior signal realizations for each fit are shown in the respective panels of Figure 5. For each noise level tested, 21CMLSTM obtains unbiased posteriors for f_*, V_c, and τ, and for the optimistic noise levels tested (i.e., $\sigma_{21} = 5$ mK and 10 mK), the constraints improve for f_X and $\nu_{\min}$. See Section 4 for further details.

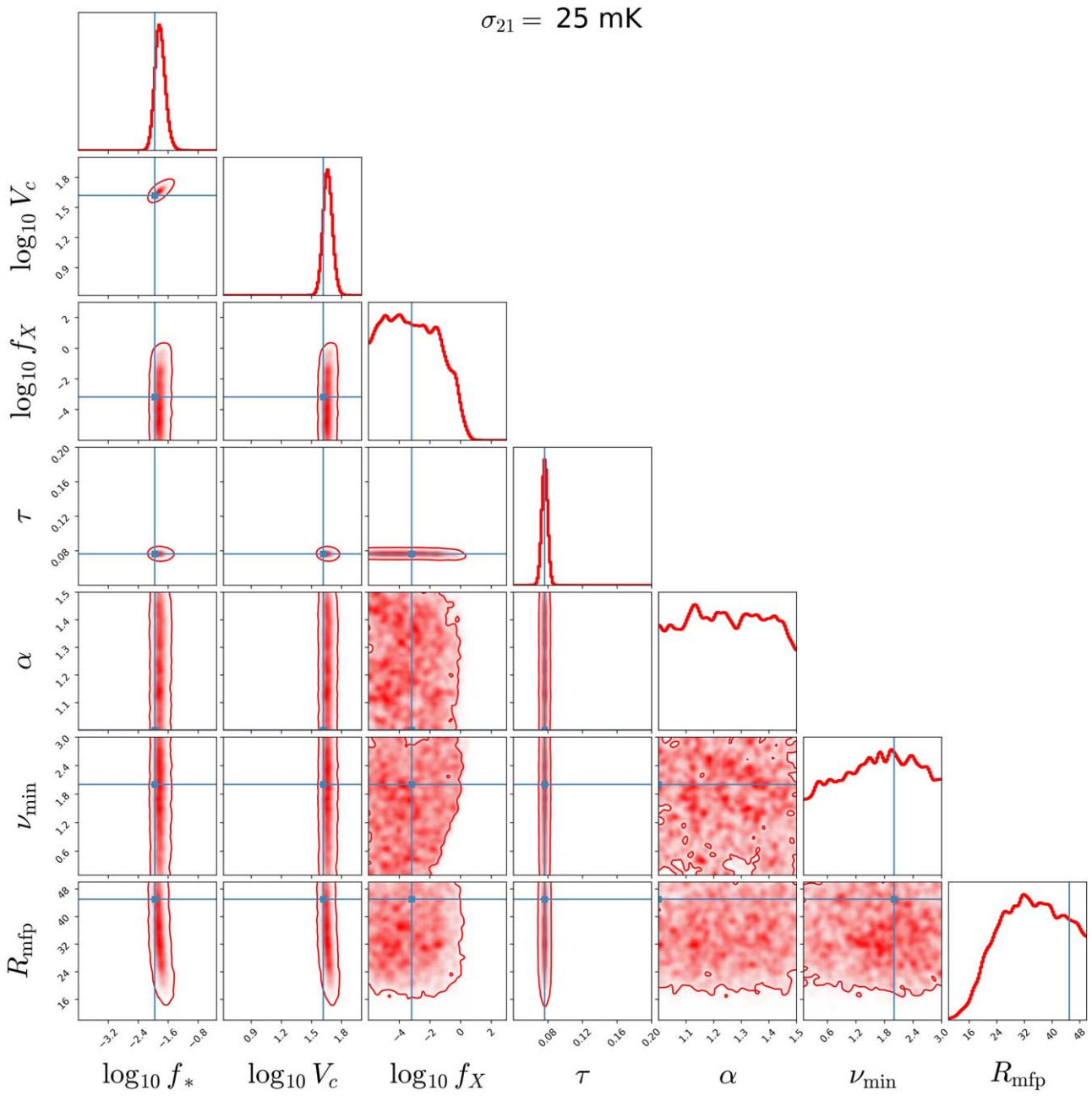

Figure B1. Marginalized 1D and 2D posterior distributions for the seven astrophysical parameters of the 21CMGEM set obtained when using 21CMLSTM to fit global 21 cm signal mock data with observational noise of $\sigma_{21} = 25$ mK (see Section 4). These parameters control the cosmic microwave background optical depth, SFE, and UV and X-ray photon production in galaxies (Table 1). The blue vertical and horizontal lines indicate the fiducial parameter values of the mock signal being fit, $\theta_0 = (f_*, V_c, f_X, \tau, \alpha, \nu_{\min}, R_{\mathrm{mfp}}) = (1.102 \times 10^{-2}, 41.534, 6.470 \times 10^{-4}, 0.076, 1, 2, 45)$, which was randomly chosen from the test set (see the right panel of Figure 5). The contour lines in the 2D histograms represent the 95% confidence levels, and density color maps are shown. The axis ranges are the full prior ranges given in Table 1.

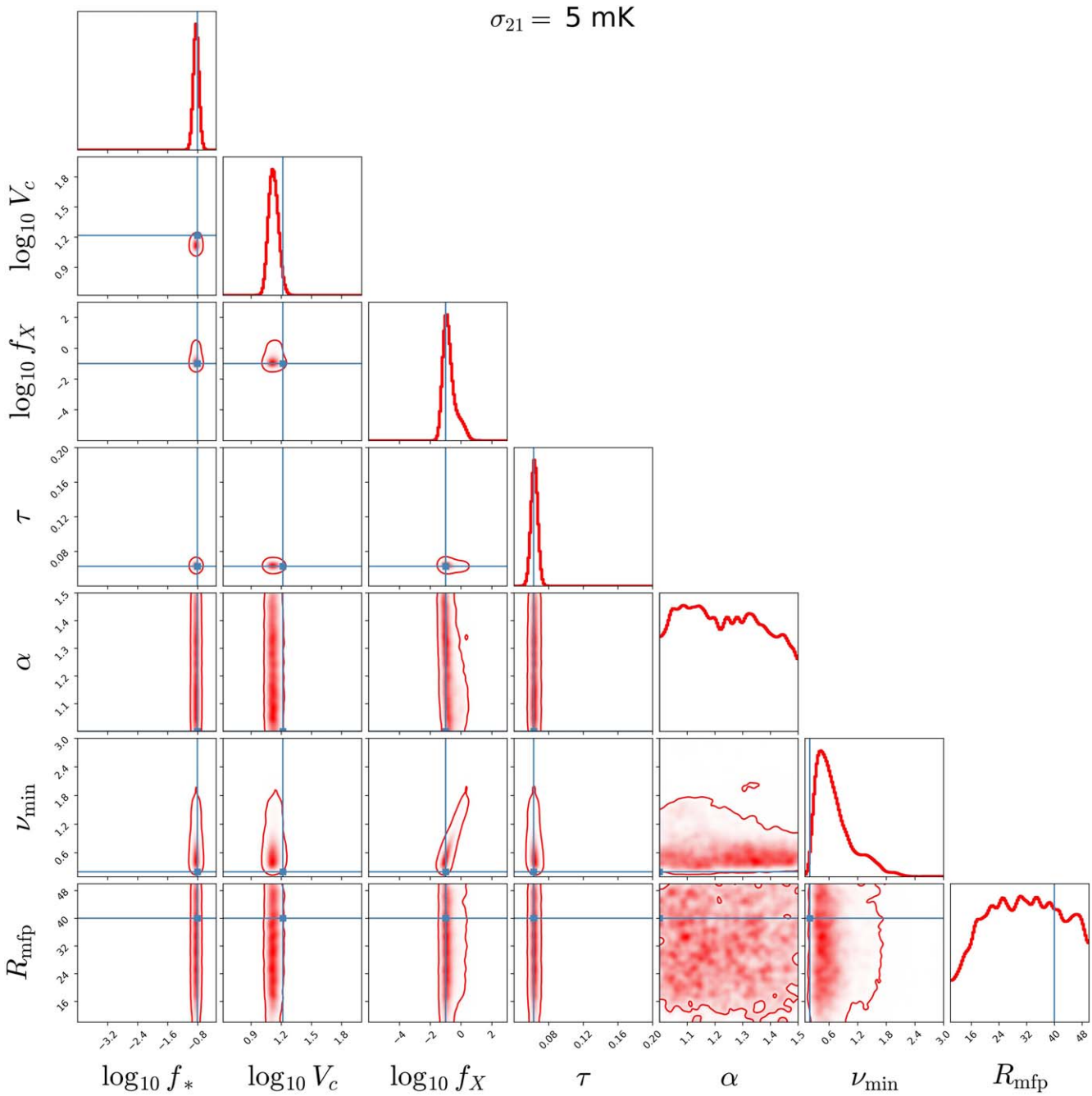

Figure B2. The same as Figure B1, but when using 21CMLSTM to fit a different randomly selected signal from the 21CMGEM test set with $\theta_0 = (f_*, V_c, f_X, \tau, \alpha, \nu_{\rm min}, R_{\rm mfp}) = (1.581 \times 10^{-1}, 16.5, 0.1, 0.0626, 1, 0.2, 40)$ and observational noise of $\sigma_{21} = 5$ mK (see the left panel of Figure 5).

ORCID iDs

J. Dorigo Jones https://orcid.org/0000-0002-3292-9784
S. M. Bahauddin https://orcid.org/0000-0003-0016-5377
D. Rapetti https://orcid.org/0000-0003-2196-6675
J. Mirocha https://orcid.org/0000-0002-8802-5581
J. O. Burns https://orcid.org/0000-0002-4468-2117

References

Abadi, M., Agarwal, A., Barham, P., et al. 2015, arXiv:1603.04467
Anstey, D., de Lera Acedo, E., & Handley, W. 2023, MNRAS, 520, 850
Ashton, G., Bernstein, N., Buchner, J., et al. 2022, NRvMP, 2, 39
Bale, S. D., Bassett, N., Burns, J. O., et al. 2023, arXiv:2301.10345
Bassett, N., Rapetti, D., Tauscher, K., et al. 2021, ApJ, 923, 33
Bera, A., Ghara, R., Chatterjee, A., Datta, K. K., & Samui, S. 2023, JApA, 44, 10
Bernardi, G., Zwart, J. T. L., Price, D., et al. 2016, MNRAS, 461, 2847
Bevins, H. T. J., de Lera Acedo, E., Fialkov, A., et al. 2022a, MNRAS, 513, 4507
Bevins, H. T. J., Fialkov, A., de Lera Acedo, E., et al. 2022b, NatAs, 6, 1473
Bevins, H. T. J., Handley, W. J., Fialkov, A., de Lera Acedo, E., & Javid, K. 2021, MNRAS, 508, 2923
Bevins, H. T. J., Handley, W. J., Lemos, P., et al. 2023, MNRAS, 526, 4613
Bevins, H. T. J., Heimersheim, S., Abril-Cabezas, I., et al. 2024, MNRAS, 527, 813
Bosman, S. E. I., Davies, F. B., Becker, G. D., et al. 2022, MNRAS, 514, 55
Bowman, J. D., Rogers, A. E. E., Monsalve, R. A., Mozdzen, T. J., & Mahesh, N. 2018, Natur, 555, 67
Bradley, R. F., Tauscher, K., Rapetti, D., & Burns, J. O. 2019, ApJ, 874, 153
Breitman, D., Mesinger, A., Murray, S. G., et al. 2024, MNRAS, 527, 9833

Buchner, J. 2023, StSur, 17, 169
Bye, C. H., Portillo, S. K. N., & Fialkov, A. 2022, ApJ, 930, 79
Chatterjee, A., Choudhury, T. R., & Mitra, S. 2021, MNRAS, 507, 2405
Chollet, F., 2015 Keras, https://github.com/fchollet/keras
Cohen, A., Fialkov, A., Barkana, R., & Monsalve, R. A. 2020, MNRAS, 495, 4845
Cohen, A., Fialkov, A., Barkana, R., & Monsalve, R. 2021, Datset for 21cmVAE, v1, Zenodo, doi:10.5281/zenodo.5084114
de Lera Acedo, E., de Villiers, D. I. L., Razavi-Ghods, N., et al. 2022, NatAs, 6, 984
Dorigo Jones, J., & Bahauddin, S., 2024 jdorigojones/21cmLSTM: 21cmLSTM Initial Release, v1.0.0, Zenodo, doi:10.5281/zenodo.13916935
Dorigo Jones, J., Rapetti, D., Mirocha, J., et al. 2023, ApJ, 959, 49
Fan, X., Strauss, M. A., Becker, R. H., et al. 2006, AJ, 132, 117
Feroz, F., & Hobson, M. P. 2008, MNRAS, 384, 449
Feroz, F., Hobson, M. P., & Bridges, M. 2009, MNRAS, 398, 1601
Feroz, F., Hobson, M. P., Cameron, E., & Pettitt, A. N. 2019, OJAp, 2, 10
Fialkov, A., Barkana, R., & Visbal, E. 2014, Natur, 506, 197
Fialkov, A., Barkana, R., Visbal, E., Tseliakhovich, D., & Hirata, C. M. 2013, MNRAS, 432, 2909
Furlanetto, S. R., Oh, S. P., & Briggs, F. H. 2006, PhR, 433, 181
Garsden, H., Greenhill, L., Bernardi, G., et al. 2021, MNRAS, 506, 5802
Gers, F. A., Schmidhuber, J., & Cummins, F. 2000, Neural Comput., 12, 2451
Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, Natur, 585, 357
HERA Collaboration, Abdurashidova, Z., Adams, T., et al. 2023, ApJ, 945, 124
Hibbard, J. J., Rapetti, D., Burns, J. O., Mahesh, N., & Bassett, N. 2023, ApJ, 959, 103
Hibbard, J. J., Tauscher, K., Rapetti, D., & Burns, J. O. 2020, ApJ, 905, 113
Hills, R., Kulkarni, G., Meerburg, P. D., & Puchwein, E. 2018, Natur, 564, E32
Hochreiter, S., & Schmidhuber, J. 1997, Neural Comput., 9, 1735
Hu, L., Chen, X., & Wang, L. 2022, ApJ, 930, 70
Huber, S., & Suyu, S. H. 2024, arXiv:2403.08029
Hunter, J. D. 2007, CSE, 9, 90
Iess, A., Cuoco, E., Morawski, F., Nicolaou, C., & Lahav, O. 2023, A&A, 669, A42
Jin, X., Yang, J., Fan, X., et al. 2023, ApJ, 942, 59
Kern, N. S., Parsons, A. R., Dillon, J. S., et al. 2020, ApJ, 888, 70
Kingma, D., & Ba, J. 2015, arXiv:1412.6980
Kluyver, T., Ragan-Kelley, B., Pérez, F., et al. 2016, in Positioning and Power in Academic Publishing: Players, Agents and Agendas, ed. F. Loizides & B. Scmidt (Amsterdam: IOS Press), 87
Kodi Ramanah, D., Arendse, N., & Wojtak, R. 2022, MNRAS, 512, 5404
Kraus, J. 1966, "System Noise" in Radio Astronomy (New York: McGraw-Hill)
LeCun, Y., Bengio, Y., & Hinton, G. 2015, Natur, 521, 436
Leeney, S. A. K., Handley, W. J., & Acedo, E. d. L. 2023, PhRvD, 108, 062006
Lemos, P., Weaverdyck, N., Rollins, R. P., et al. 2023, MNRAS, 521, 1184
Li, J. I.-H., Johnson, S. D., Avestruz, C., et al. 2024, arXiv:2407.14621
Liu, A., & Shaw, J. R. 2020, PASP, 132, 062001
Liu, H., Liu, C., Wang, J. T. L., & Wang, H. 2019, ApJ, 877, 121
Mahalanobis, P. C. 2018, Sankhya, 80, S1
Mason, C. A., Naidu, R. P., Tacchella, S., & Leja, J. 2019, MNRAS, 489, 2669
McGreer, I. D., Mesinger, A., & D'Odorico, V. 2015, MNRAS, 447, 499
Mertens, F. G., Mevius, M., Koopmans, L. V. E., et al. 2020, MNRAS, 493, 1662
Mesinger, A., Furlanetto, S., & Cen, R. 2011, MNRAS, 411, 955
Mirocha, J. 2014, MNRAS, 443, 1211
Mirocha, J., & Furlanetto, S. R. 2019, MNRAS, 483, 1980
Mirocha, J., Furlanetto, S. R., & Sun, G. 2017, MNRAS, 464, 1365
Mirocha, J., Skory, S., Burns, J. O., & Wise, J. H. 2012, ApJ, 756, 94
Monsalve, R. A., Fialkov, A., Bowman, J. D., et al. 2019, ApJ, 875, 67
Muñoz, J. B., & Cyr-Racine, F.-Y. 2021, PhRvD, 103, 023512
Murray, S. G., Bowman, J. D., Sims, P. H., et al. 2022, MNRAS, 517, 2264
Paciga, G., Chang, T.-C., Gupta, Y., et al. 2011, MNRAS, 413, 1174
Pagano, M., Sims, P., Liu, A., et al. 2024, MNRAS, 527, 5649
Pascanu, R., Mikolov, T., & Bengio, Y. 2013, in Proc. Machine Learning Research, Proc. 30th Int. Conf. on Machine Learning, 28, ed. S. Dasgupta & D. McAllester (PMLR), 1310, https://proceedings.mlr.press/v28/pascanu13.html
Pattison, J. H. N., Cavillot, J., Bevins, H. T. J., Anstey, D. J., & de Lera Acedo, E. 2024, arXiv:2408.06012
Prelogović, D., Mesinger, A., Murray, S., Fiameni, G., & Gillet, N. 2022, MNRAS, 509, 3852
Qin, Y., Mesinger, A., Park, J., Greig, B., & Muñoz, J. B. 2020, MNRAS, 495, 123
Rapetti, D., Tauscher, K., Mirocha, J., & Burns, J. O. 2020, ApJ, 897, 174
Rumelhart, D. E., Hinton, G. E., & Williams, R. J. 1986, Natur, 323, 533
Saxena, A., Meerburg, P. D., Weniger, C., de Lera Acedo, E., & Handley, W. 2024, arXiv:2403.14618
Schmit, C. J., & Pritchard, J. R. 2018, MNRAS, 475, 1213
Shaver, P. A., Windhorst, R. A., Madau, P., & de Bruyn, A. G. 1999, A&A, 345, 380
Shen, E., Anstey, D., de Lera Acedo, E., & Fialkov, A. 2022, MNRAS, 515, 4565
Sherstinsky, A. 2020, PhyD, 404, 132306
Shi, X., Chen, Z., Wang, H., et al. 2015, arXiv:1506.04214
Shi, Y., Deng, F., Xu, Y., et al. 2022, ApJ, 929, 32
Sims, P. H., Bowman, J. D., Mahesh, N., et al. 2023, MNRAS, 521, 3273
Sims, P. H., & Pober, J. C. 2020, MNRAS, 492, 22
Singh, S., Jishnu, N. T., Subrahmanyan, R., et al. 2022, NatAs, 6, 607
Singh, S., Subrahmanyan, R., Udaya Shankar, N., et al. 2018, ApJ, 858, 54
Skilling, J. 2004, in AIP Conf. Ser. 735, Bayesian Inference and Maximum Entropy Methods in Science and Engineering, ed. R. Fischer, R. Preuss, & U. V. Toussaint (Melville, NY: AIP), 395
Smith, S. L., Kindermans, P.-J., Ying, C., & Le, Q. V. 2017, arXiv:1711.00489
Staudemeyer, R. C., & Morris, E. R. 2019, arXiv:1909.09586
Sun, Z., Bobra, M. G., Wang, X., et al. 2022, ApJ, 931, 163
Tabasi, S. S., Salmani, R. V., Khaliliyan, P., & Firouzjaee, J. T. 2023, ApJ, 954, 164
Tauscher, K., Rapetti, D., & Burns, J. O. 2020, ApJ, 897, 132
Trott, C. M., Jordan, C. H., Midgley, S., et al. 2020, MNRAS, 493, 4711
Van Rossum, G., & Drake, F. L., Jr. 1995, Python Reference Manual (Amsterdam: Centrum voor Wiskunde en Informatica)
Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, NatMe, 17, 261
Visbal, E., Barkana, R., Fialkov, A., Tseliakhovich, D., & Hirata, C. M. 2012, Natur, 487, 70
Williams, R. J., & Zipser, D. 1995, Gradient-based Learning Algorithms for Recurrent Networks and their Computational Complexity (USA: L. Erlbaum Assoc. Inc.), 433
Zhang, R., Liu, Y., & Sun, H. 2020, CMAME, 369, 113226
Zheng, Y., Li, X., Yan, S., et al. 2023, MNRAS, 521, 5384
Zhu, Y., Becker, G. D., Bosman, S. E. I., et al. 2022, ApJ, 932, 76