

## Forecasting of In Situ Electron Energy Loss Spectroscopy

Nicholas R. Lewis,<sup>1</sup> Yicheng Jin,<sup>1</sup> Xiuyu Tang,<sup>1</sup> Vidit Shah,<sup>1</sup> Christina Doty,<sup>2</sup> Bethany E. Matthews,<sup>3</sup> Sarah Akers,<sup>2</sup> and Steven R. Spurgeon<sup>3,4, a)</sup>

<sup>1)</sup>*Department of Chemical Engineering, University of Washington, Seattle, Washington 98195*

<sup>2)</sup>*National Security Directorate, Pacific Northwest National Laboratory, Richland, Washington 99352*

<sup>3)</sup>*Energy and Environment Directorate, Pacific Northwest National Laboratory, Richland, Washington 99352*

<sup>4)</sup>*Department of Physics, University of Washington, Seattle, Washington 98195*

(Dated: 15 July 2022)

Forecasting models are a central part of many control systems, where high consequence decisions must be made on long latency control variables. These models are particularly relevant for emerging artificial intelligence (AI)-guided instrumentation, in which prescriptive knowledge is needed to guide autonomous decision-making. Here we describe the implementation of a long short-term memory model (LSTM) for forecasting of electron energy loss spectroscopy (EELS) data, one of the richest analytical probes of materials and chemical systems. We describe key considerations for data collection, preprocessing, training, validation, and benchmarking, showing how this approach can yield powerful predictive insight into order-disorder phase transitions. Finally, we comment on how such a model may integrate with emerging AI-guided instrumentation for powerful high-speed experimentation.

Keywords: Machine learning, long short-term memory, LSTM, forecasting, electron energy loss spectroscopy, transmission electron microscopy

---

<sup>a)</sup>Electronic mail: [steven.spurgeon@pnl.gov](mailto:steven.spurgeon@pnl.gov)

## I. INTRODUCTION

Reward-based decision-making is directly linked to our ability to accurately forecast, or anticipate, changes in a system or process. Effective forecasting is essential for many disciplines and technologies we take for granted, ranging from meteorology<sup>1</sup> to the power grid<sup>2</sup> and from stock trading<sup>3</sup> to logistics.<sup>4</sup> The recent rise of autonomous vehicles, including automobiles, drones, and spacecraft, has been propelled by advanced forecasting models deployed on high performance computing platforms.<sup>5,6</sup> Abundant low-cost computing and the proliferation of machine learning (ML) have enabled many new real-time forecasting approaches. When performed correctly, forecasting can save time, reduce cost, and guide scientific discovery by helping direct decision-making. Consequently, much of the scientific community is interested in the development and application of good forecasting models; notable examples include medicine,<sup>7,8</sup> climate science,<sup>9,10</sup> and high-energy physics.<sup>11,12</sup> However, other disciplines, such as materials science and chemistry, have been slower to adopt these approaches, often due to a lack of domain-specific analytics and control frameworks.

These issues are exemplified in the field of electron microscopy, which showcases both the challenges and opportunities for forecasting. Today’s scanning transmission electron microscopy (STEM) represents the “gold standard” for the observation of materials and chemical processes at high spatial and temporal resolution. Everything from crystal growth to battery cycling and alloy fatigue can be observed in situ, using elaborately designed stages, aberration-corrected sub-Ångström probes, and high-speed detectors.<sup>13–15</sup> While advanced hardware can easily generate large volumes of data, our ability to interpret, anticipate, and automatically act on such data is limited.<sup>16,17</sup> For many studies, both ex and in situ, we must make rapid decisions on high-latency control parameters using information from high-throughput, multimodal data streams. However, we currently lack the necessary (1) low-level control, (2) descriptive models, and (3) forecasting (prescriptive) approaches to implement more powerful decision-making.

Recently, significant progress has been made toward new microscope automation platforms (1) that allow for centralized, data-driven control of instrument operations.<sup>18–20</sup> Data can then be passed through two main kinds of descriptive models (2): those based on deeply trained ML networks fed large volumes of hand-labeled or simulated examples,<sup>21–26</sup> or those based on few-shot ML approaches utilizing sparse, canonical examples.<sup>27</sup> We have previously

demonstrated<sup>28,29</sup> the ability to conduct efficient, generalizable, and task-based automated classification via the latter approach. While such models are an important development, it is increasingly clear that we must move beyond purely descriptive models to realize truly autonomous experimentation. Specifically, we require forecasting models (3), that allow us to anticipate changes in data streams and prescriptively tune high-latency control parameters. Currently, a whole range of experiments, such as studying heating- or beam-induced phase transitions, tracking of particles and reaction fronts, as well as operando switching of ferroic and quantum materials, is difficult or impossible to conduct.<sup>30</sup> For many such studies, the experimental system (encompassing both sample and instrument) is slow to respond to changes in control parameters due to mechanical instability (movement), thermal mass (heating), and hysteresis (electric and magnetic field). Because of this latency, human-in-the-loop control is often unfeasible; once the operator has seen that something has changed, it is usually too late to implement a manual response.

Fortunately, this prediction and control problem is quite similar to those encountered in the other aforementioned domains. ML approaches are particularly well-suited to the study of higher-dimensional, noisy, or complex datasets, where latent correlations may not be immediately obvious to a human operator. A variety of time-dependent ML-based prediction approaches exist, with the most common being based on recurrent neural networks (RNNs) and gated recurrent units (GRUs).<sup>31</sup> Among the former, long short-term memory (LSTMs) are commonly used to incorporate knowledge of past experiences to model Markov-type decision processes.<sup>32</sup> LSTM models have been extensively applied to serial data, such as text, audio, and video.<sup>33</sup> Despite their prevalence, there has been surprisingly little work on the use of LSTMs in electron microscopy, with limited examples including control of scan generation<sup>34</sup> and segmentation of biological images.<sup>35</sup> Given that STEM data is acquired in serial fashion, we aim to evaluate the performance of the LSTM model for microscope data, with an eye toward practical implementation.

Here we describe an LSTM approach for forecasting of in situ electron energy loss spectroscopy (EELS) data collected in the STEM, a model we call EELSTM. We have chosen this technique because it strongly encodes local chemical state and phase, can be readily quantified using existing theoretical frameworks, and can be acquired at high speed and energy resolution.<sup>36</sup> We explore the crystalline-to-amorphous phase transition in the archetypal perovskite oxide SrTiO<sub>3</sub> (STO), utilizing the electron beam itself to drive reduction and

associated changes in core loss EELS spectra. Understanding such order-disorder phase transitions is important for emerging technologies, ranging from solid oxide fuel cells (SOFCs) to sensors in extreme environments and radiation-hard electronics.<sup>37</sup> We systematically explore data preprocessing, model architecture, hyperparameter optimization, training, and validation relative to ground truth experimental data. We show that this model has excellent predictive power and may serve as a basis for future model-predictive control approaches. Finally, we comment on the potential deployment of this model in emerging autonomous microscope systems.

## II. RESULTS AND DISCUSSION

The chief aim of the present study is to adapt existing LSTM models, such as those used for natural language processing<sup>38,39</sup> and time series prediction of physical phenomena,<sup>40–42</sup> to the task of EELS forecasting. As shown in Figure 1, the EELSTM model workflow encompasses four steps: **Data Collection**, **Preprocessing**, **Training and Validation**, and **Inference**. In Section II A, we describe the experimental setup and EELS data acquisition, including considerations for the best model performance. In Section II B, we review preprocessing strategies unique to EELS data, resulting from the data collection process, variability between experiments, and the nature of core loss data itself. In Section II C, we describe the training and validation process, including the relationship between training inputs and predictions, model transferability, and temporal correlations. Finally, in Section II D, we discuss possible error metrics and benchmark performance relative to ground truth experimental data.

### A. Data Collection

We have chosen to examine a crystalline STO sample, which will readily undergo reduction and a crystalline-to-amorphous phase transformation due to electron beam knock-on damage at 300 keV accelerating voltage. Several datasets were collected by parking the electron beam on an undamaged part of the sample and then acquiring time series spectra with a fixed dwell time of 0.08, 0.1, 0.2, 0.4, or 0.8 s px<sup>-1</sup>, while keeping all other instrument parameters constant. We observed that consistency in operating conditions (such as beam

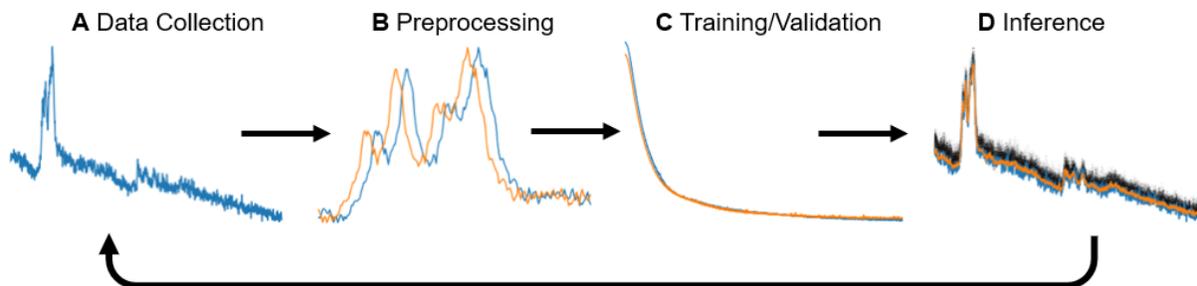


FIG. 1. **Overview of the EELSTM model workflow.** (A–D) Steps of Data Collection, Preprocessing, Training and Validation, and Inference, respectively. **Add label on bottom arrow.** **Replace labels with bold lower case letters.**

energy, dwell time, and material) between experiments is paramount, as models trained on data with specific beam parameters did not perform well on spectra acquired with differing parameters. This discrepancy arises because EELS intensity depend on dwell time and the damage rate (and hence phase transition) varies with dose conditions.

We conducted three different experiments under similar conditions to obtain the training and test datasets. Two whole experiments’ worth of data were used to construct the training dataset, while a third experiment was held back as a test set. This construction is important, because the model is able to train on data showing all stages of the phase transition; if the first portion of a single experiment were used to train—and the later portion was then utilized as the test set—the model would not accurately predict future spectra in the later phases. Finally, this construction also allows for extrapolation to new experiments, where predictions can be made independent of the progression of the phase transition.

## B. Preprocessing

We next consider data preprocessing, which includes iterative steps unique to in situ EELS data, as shown in Figure 1. Initially, we designed the model to use the raw EELS spectra and trained an LSTM model for predictions. This method resulted in an excessively long training phase, with over 30,000 epochs required for convergence. After further examination of the data and a review of experimental considerations, we identified a variety of preprocessing steps to improve training and accuracy. Figure 2 shows an overview of these

preprocessing methods and we next discuss their rationale, implementation, and effect on model predictions.

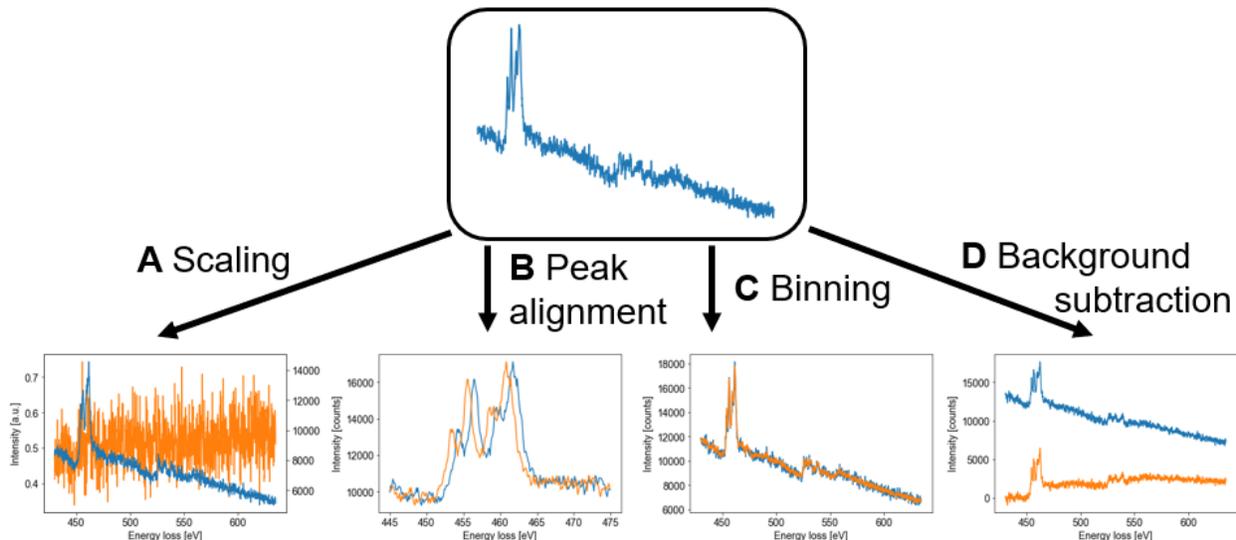


FIG. 2. **Data preprocessing.** (a–d) Strategies include scaling of spectra, peak alignment between experiments, binning of spectra to reduce noise, and background subtraction, respectively. Raw spectra are shown in blue and processed spectra are shown in orange. **Replace labels with bold lower case letters.**

### 1. *Scaling and Normalization*

The first data preprocessing required is to scale the data between 0 and 1. The primary reason for this is that LSTM networks use several zero-centered or nearly zero-centered functions, such as sigmoid and hyperbolic tangent.<sup>32</sup> The derivatives of these functions diminish greatly outside of this input range, and so training weights receive small updates based on the gradient when the input data is outside of this optimal range.<sup>43</sup> We can consider several ways of scaling the data, knowing that raw intensity counts range in the thousands to tens of thousands. The most logical, due to the inherent data structure, is a manual scaling, where the training set is scaled between 0 and 1 using the min and max values across the spectrum. Thus, the inherent link between energy bins is conserved. Conversely, we may also use the scikit-learn library `MinMaxScaler` to scale each energy bin individually between 0 and 1, so the inherent structure between energy bins is lost (this is the method

represented in Figure 2.a).

After reading in the raw data, we scaled it based on one of the training datasets, formatted the data into the sequence/output format for the LSTM and trained the model. We then called the trained model to make a prediction and unscaled the predicted result to compare to the ground truth. We assumed that the second method utilizing scikit-learn’s MinMaxScaler would preserve signal-to-noise in lower intensity regions, but our testing showed that the noise levels from the predictions were not statistically different. Most interestingly, we observed that both scaling methods yielded similar error, indicating that the channel-to-channel relationship did not need to be maintained for the model. While performance with scaled data did show an improvement over the model with raw data, the biggest benefit was faster convergence. Models were able to train approximately  $10\times$  faster, primarily due to convergence in fewer epochs. This finding demonstrates the importance of scaling data in a range where the sigmoid and activation functions have a more significant impact due to gradient-based weight updates. To evaluate the performance of this and the following preprocessing steps, we consider the mean squared error (MSE) and root mean squared error (RMSE) relative to ground truth, as will be described Section IID. As a baseline, the RMSE of the raw data before any preprocessing is 1958.3. After scaling, performance improved greatly to a RMSE value of  $295.5 \pm 40.3$  relative to raw spectra. Finally, it should be noted that, while scaling is the first strategy implemented for improved performance, it should always be done only after all other preprocessing steps have been implemented. For example, if background subtraction is implemented, scaling should only be done after the background subtraction step.

## 2. *Peak Alignment*

Because of the nature of EELS data acquisition, it is important to account for spectral shifts between experiments that might influence forecasts. While a systematic shift in core-loss edge onset is often related to oxidation state, instability in the microscope high tension system can also introduce artificial shifting. To correct for this, low-loss and core-loss data may be acquired simultaneously and the core-loss data can then be shifted to account for energy drift throughout the experiment.<sup>44</sup> However, not all instruments possess the required spectrometer hardware and shuttering between energy regimes can add overhead (slow down)

an acquisition, making this approach difficult to apply during high-speed experimentation. For simplicity, we treat the entire core-loss spectrum, aiming to minimize artificial shifts for more accurate prediction and error metrics.

In order to make a generic alignment for all future data, we used one of the timesteps from the test spectra as a reference. We utilized the peak alignment functionality of the Hyperspy Python library to align all spectra to this reference spectrum. As a result of shifting peaks, some of the data from the edges of the full spectra were lost. One of the fundamental characteristics required for model inputs is consistency between number of channels; therefore, we cropped all spectra after alignment to ensure consistent numbers of energy channels. For the data shown in Figure 2.b, the raw spectrum had 2048 channels prior to alignment. After alignment, which typically lost  $\leq 10$  channels, we cropped 74 channels from the beginning and end, yielding a final number of 1900 energy channels.

This alignment improved the RMSE between the predicted and real spectra to  $217.9 \pm 23.4$ . We consider two explanations: first, a shift in energy channels between a real and predicted spectrum leads to significant increase in error around regions of interest, such as the Ti  $L_{2,3}$  edge at  $\sim 456$  eV and O  $K$  edge at  $\sim 532$  eV. Second, the model learns trends for energy bins as distinct input features; when there is a shift between the spectra that were used to train the model and those used for prediction, we are asking for additional extrapolation. While more training datasets covering a wide range of shifts might eliminate this step, this preprocessing proved important for more limited amounts of training data.

### ***3. Binning and Smoothing***

We observed that the predicted spectrum was not able to capture the natural noise in the real spectrum, leading to an increased error between the prediction and ground truth. Multiple sources of noise exist in EELS data, including shot noise, gain noise, read-out noise, and Fano noise.<sup>45</sup> While shot and Fano noise arise prior to signal detection, they are influenced by the point spread function (PSF) of the detector and this should be considered in generalizing a predictive model to other experiments. Further, both gain and read-out noise are affected by the choice of spectrometer binning and gain correction. While some of these parameters can be fixed for a specific prediction, the intrinsic stochastic nature of noise makes it challenging to predict.

Therefore, we proposed that measured predictive performance might improve if the training data were less noisy. A typical method of reducing noise is to average spectra across several timesteps. We consider two such binning methods: “exclusive bins” and “rolling bins.” The exclusive bins method averaged every  $n$  spectra ( $n$  was typically 3–5) without any overlap; that is, we averaged  $n$  spectra, then shifted forward by  $n$  timesteps to average the next group. This approach made sure to not use any spectra more than others in the averaging, but was detrimental, given that the amount of available data was reduced by a factor of  $n$ . The rolling bins method averaged  $n$  spectra, then shifted the timestep forward by 1 to take the next average; this approach resulted in reusing all spectra  $n$  times (except for the very first and last  $n$  spectra), but was advantageous since the amount of training data was not significantly reduced (we only lose  $n$  timesteps of data).

Both methods were in fact effective in reducing noise in the datasets, as shown in Figure 2.c. However, the reduction in training data, particularly when using the exclusive bins method, actually yielded worse results (RMSE of  $400.8 \pm 154.8$ ). This result is explained by the fact that good performance must have a sufficient amount of training data for the model to predict well, and losing so much data was detrimental. To employ this approach in future studies, one would need to gather significantly more experimental data to offset the loss of training data. The rolling bins method improved the RMSE to  $241.4 \pm 63.7$  from the baseline. However, this improvement is smaller than that seen from spectrum shifting. When both approaches were combined, no significant improvement was made to the error and this strategy was discarded, since it only added to preprocessing time.

#### 4. *Background Subtraction*

Finally, we consider the natural decrease of the inelastic scattering background at higher energy losses, which primarily results from plasmon excitations that can be described by a well-known power law dependence. This behavior led to some instances where the predicted signal was shifted vertically from the actual signal, leading to inflated error between the predicted and real spectrum. To mitigate this effect on prediction error, we utilized background subtraction so that all of the signals started on a comparable baseline, as shown in Figure 2.d. It was necessary to perform background subtraction before any scaling or normalization, since the background subtraction shifted the entire baseline.

TABLE I. Changes in RMSE as a function of preprocessing.

Approach	RMSE	% Improvement
Raw	1958.3	—
Scaling and Normalization	295.5	84.9
Peak Alignment	217.9	88.9
Binning and Smoothing	400.8	79.5
Background Subtraction	392.4	80.0

In general, best practice dictates specifying the region for background subtraction to be taken before the edge of interest. This approach proved problematic for this particular study, however, because we were performing predictions on an entire spectrum, not just a single edge of interest. Most compounds contain multiple edges in a given spectrum and STO specifically contains both the Ti  $L_{2,3}$  and O  $K$  edges. For consistency, we performed the background subtraction using the region before the Ti  $L_{2,3}$  edge. After training the model on background-subtracted data, we observed decreased performance with higher variability (RMSE of  $392.4 \pm 177.4$ ). We observed that models trained on this type of data had a greater tendency to overfit. This behavior may be explained given the fact that background subtraction constrains the data to a narrower range of values, and after scaling, the data tends to not extrapolate as well to unseen data, resulting in an increased likelihood to overfit. The results highlighted the problem with using background subtraction when considering the whole spectrum, and would perhaps be more useful when utilizing different error metrics. This method would also be more suitable for modeling only a certain energy regime, where background subtraction in one region can be carried out independently from another area of interest. Therefore, background subtraction was not deemed a necessary preprocessing step for best results in the current implementation. In summary, we can see that individual preprocessing steps can improve RMSE by nearly 88.9%, as shown in Table I.

### C. Training and Validation

The generic structure of the LSTM model takes as input a sequence of time series data and outputs a prediction of a future timestep, as shown in Figure 3. We first considered two models: one that took as input a sequence of whole spectra and predicted a whole spectrum,

or an aggregate model that analyzes input and predictions channel-by-channel, followed by recombination of the individual predictions to form a whole spectrum. Given that each spectrum contained 2048 energy channels, the second method required 2048 separate models; this method was discarded primarily due to the prohibitive computational time and memory, as well as initially positive results from the first approach.

The next consideration for the model revolved around deciding the prediction horizon for the model. In time series, there is often a correlation between timesteps, and care must be taken to ensure actionable predictions. If the correlation is too high, the model may learn to simply predict the last timestep of the input sequence as the next timestep, leading to artificially inflated prediction accuracy. For example, in the “long input, short horizon” scenario in Figure 3.a with an 8 timestep input and 1 timestep horizon, the prediction appears to have an extremely high fidelity. However, this result is not significantly better than simply using the last spectrum from the input sequence as the prediction. To account for this, we performed a Pearson autocorrelation calculation between a spectrum at a given timestep and all subsequent spectra, as shown in Supplementary Figure 6. As anticipated, we observed a high correlation among timesteps proximal to each other, with a steadily decreasing correlation. After approximately 6–8 timesteps, the correlation between spectra was sufficiently low to ensure that the model would not simply learn the most recent spectrum from the input. For an EELS dwell time of 0.4 seconds, this corresponds to  $\sim 3$  seconds.

The length of the input sequence was optimized along with other model hyperparameters in the range of 3–15 timesteps, as shown in Table II. Naturally, a sufficiently long sequence of spectra must be used to establish the progression of the STO phase transformation that can predict out several timesteps. We observe poor results when a short input sequence is used, as shown in the “short input, long horizon” scenario in Figure 3.b with a 3 timestep input and 8 timestep horizon. However, we also wish to avoid excessively long input sequences to maintain sensitivity to rapid changes in the data. In the context of automation, a longer input sequence leads to more lag from the point when a control parameter is changed and the system starts gathering data, increasing the likelihood of inaccurate reaction tracking. We determined an ideal “Long input, long horizon” scenario as shown in Figure 3.c, with an 8 timestep input and 8 timestep horizon, indicating good prediction with minimal autocorrelation. Additional hyperparameter optimization was performed with the Hyperopt package from Python, which utilizes Bayesian optimization to selectively search in spaces

where performance tends to be better. While this approach does not guarantee a globally optimal hyperparameter set, it does have the advantage of searching in the most optimal sub-spaces (unless it gets stuck in a local minimum). We emphasize that further tuning of the input and horizon may be necessary depending on the exact parameters of an EELS experiment and the nature of the behavior under study.

#### D. Inference and Benchmarking

Lastly, we consider inference and benchmarking of model performance in different stages of an STO phase transition. The inference step (Figure 1.d) takes an input sequence from a new experiment (the test set), makes a prediction, and compares it to the ground truth. Results then iteratively inform data collection (such as more experiments, adjusting sampling, dose, etc.) and preprocessing steps (such as background subtraction, binning, etc.), after which the model is retrained and reevaluated. Importantly, it is possible to readily transform the output of the LSTM (in the form of a numpy array) back to a Hyperspy signal to use its built-in functionality, such as edge quantification and EELS-specific post-processing.

As already mentioned, the primary benchmarking functions used in this study are MSE and RMSE, which were selected for their straightforward interpretation. However, they are relatively simplistic, since they treat each energy channel equally; essentially, the EELS background contributes just as much to the overall error as a region of interest, such as the Ti  $L_{2,3}$  or O  $K$  edges. In light of this limitation, we also considered additional metrics to both train the model and evaluate performance, including: cosine similarity, peak MSE, and weighted MSE. Cosine similarity is a very common and effective metric for showing similarity between two vectors. The disadvantage is that our spectra all have very high cosine similarity (0.999 and greater), so it is difficult to optimize over this range. Logarithmic scaling or other strategies to inflate the difference between 0.999 and 1.0 might be useful, but in the end, it was discarded. Alternatively, we considered peak MSE as a new EELS-specific objective function, where the predicted and true signals are taken as input. Hyperspy fitting functions may then be used to determine peak location, height, and width. The MSE of those metrics between the predicted and true spectrum is the objective function. Using this approach, it is possible to simultaneously evaluate multiple spectral regions by assigning peaks of location, height, and width = 0 to make up the disparity. This approach is beneficial

because it significantly penalizes mismatched number of peaks more than simply wrong peaks. However, it completely ignores the background and so does not lead to interpretable spectra. Finally, weighted MSE is a method that specifies which regions are more important and then multiplies the error of those regions by an additional factor to lend additional penalization in the objective function.<sup>46</sup> While this approach should theoretically reward better performance in key regions of interest, we found that the overall performance was comparable to standard MSE; since this did not improve interpretability, the final evaluations used standard MSE and RMSE.

We can now evaluate the performance of the forecasting model on in situ EELS data taken at various stages of an order-disorder phase transition in STO. Figure 4 shows prediction results on a representative EELS dataset, using peak alignment and signal scaling as discussed in Section II B, and the set of hyperparameters determined in Table II. Predictions are made using an 8 timestep input, with an 8 timestep horizon, then overlaid against ground truth (raw) data. We first consider Figure 4.a, which shows the initial stages (time =  $\sim 15$  s) of the phase transition across the whole spectrum and two regions of interest, the Ti  $L_{2,3}$  and O  $K$  edges. We observe an excellent prediction across the entire spectrum (MSE = 216.7) relative to the ground truth data, with the added benefit of denoising relative to the raw experimental data. Focusing on the Ti  $L_{2,3}$ , we observe the expected crystal field splitting of the white lines into  $t_{2g}$  and  $e_g$  contributions, indicating a predominant  $Ti^{4+}$  valence state within the resolution of the measurement. Similarly, we observe expected features in the O  $K$  edge consistent with this valence state. Next, we consider a later stage in the phase transition, shown in Figure 4.b. At this time (time =  $\sim 60$  s), the sample is heavily reduced by the beam and increasingly amorphous. The substantial presence of  $Ti^{3+}$  is reflected in the increasing degeneracy of the Ti  $L_{2,3}$  edge states and merging of the features in the  $L_3$  and  $L_2$  peaks. Similarly, there is less definition and a general flattening of the O  $K$  edge features, again consistent with reduction.<sup>47</sup> Here again, we observe strong predictive capability across the full spectrum (MSE = 181.4), pointing to the ability of the model to capture the future state of a phase transition.

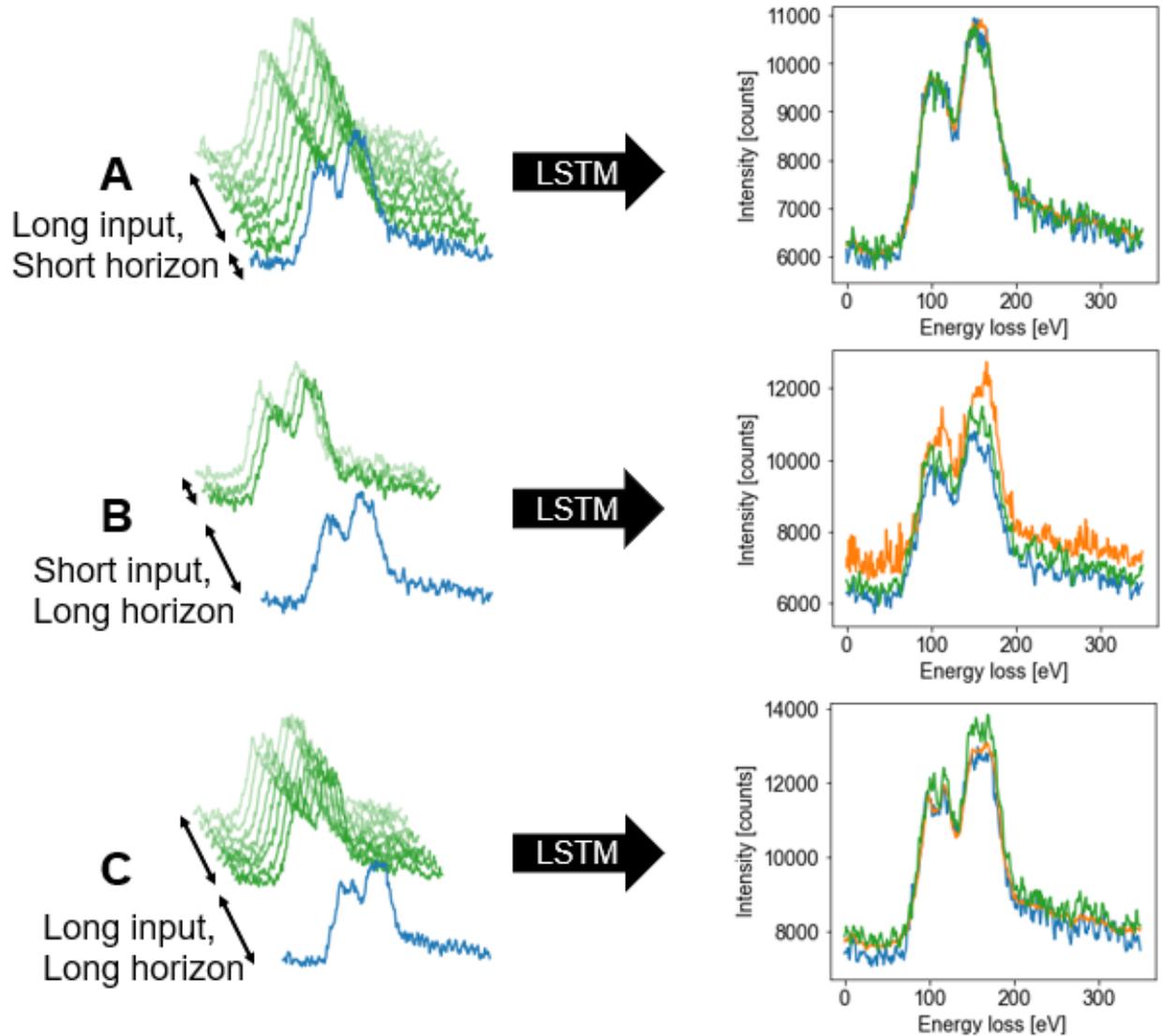


FIG. 3. **Input sequence and output forecast horizon scenarios.** (a) “Long input, short horizon” scenario with an 8 timestep input and 1 timestep horizon, demonstrating deceptively good results. (b) “Short input, long horizon” scenario with a 3 timestep input and 8 timestep horizon, demonstrating poor results. (c) “Long input, long horizon” scenario, with an 8 timestep input and 8 timestep horizon, indicating good prediction with minimal autocorrelation. The green lines represent spectra from the input sequence, the blue line is the true future spectra to predict, and the orange line is the predicted spectrum from the LSTM. Note that the plots on the right show only the last spectrum from the input sequence (in green). **Calibrate energy axis and tell me what colors are what. Replace labels with bold lower case letters.**

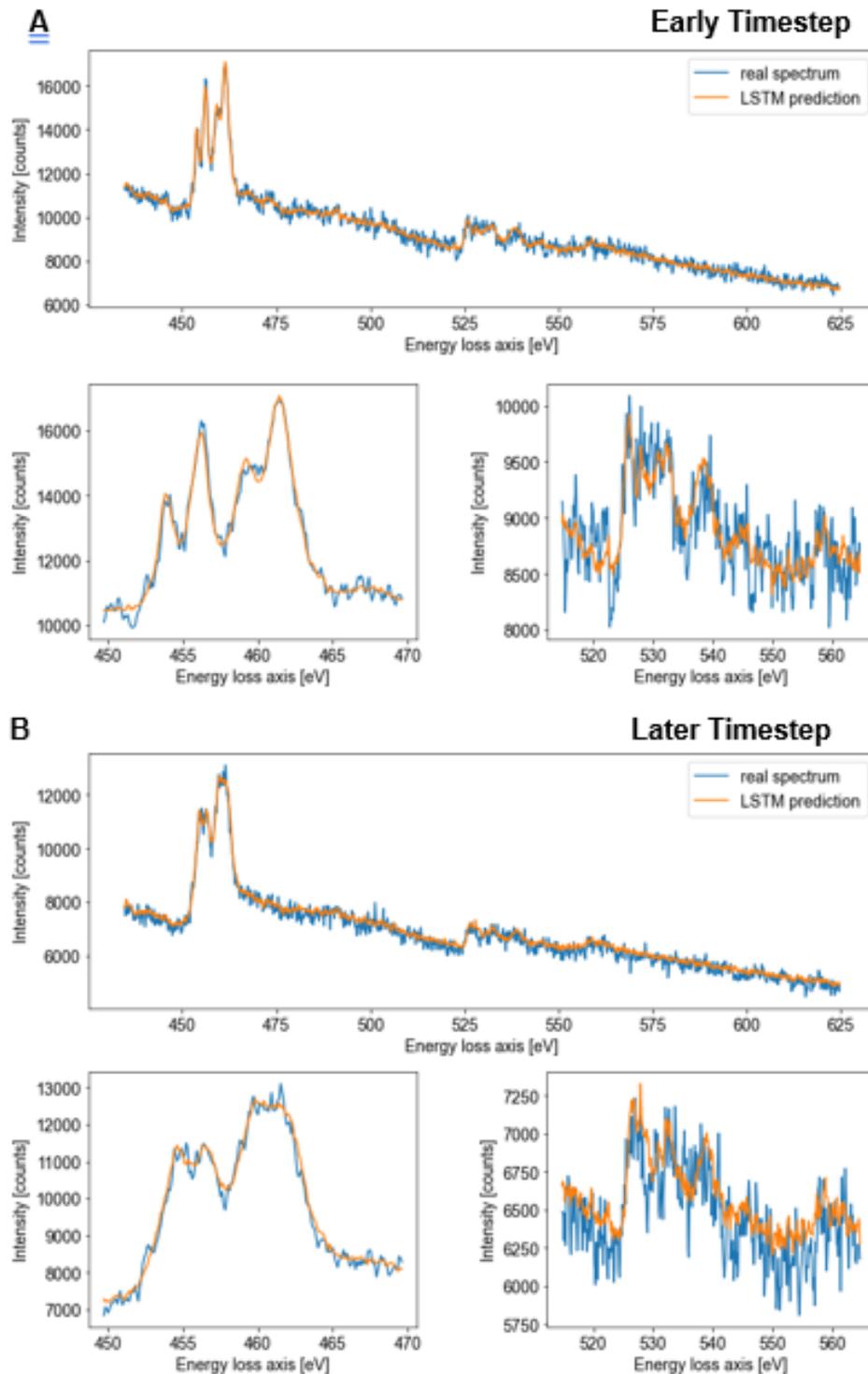


FIG. 4. **Optimized LSTM prediction of an STO order-disorder phase transition.** (a) Timestep corresponding to the start of the experiment, where the sample is crystalline and undamaged. (b) Timestep near the end of an experiment, where the sample is largely amorphous and heavily damaged. The blue line shows the real spectrum at an 8 timestep horizon, while the orange shows the LSTM model output. **Add chemical edge label and shade region in whole spectrum. Can we do a background subtraction prior to each edge (Ti L and O K)**

### III. CONCLUSIONS

We have described the implementation of an LSTM model for forecasting of EELS spectra during an in situ phase transition in STO. We find that the model possesses excellent predictive power relative to ground truth experimental data, but that there are important pre-processing and temporal correlation steps that must be considered. Moving forward, it will be important to further evaluate model accuracy against prediction horizon. It will also be necessary to explore error metrics that improve the interpretability of results and consider models that account for the physics of different beam parameters or materials. For implementation in emerging AI systems, we envision the EELSTM forecasting model should be running continuously on a rolling buffer of EELS data and implemented in model-predictive control frameworks for closed-loop feedback.<sup>18</sup> We expect that this approach will find widespread use in studies of high-speed phase transitions for fundamental studies of crystal nucleation and growth, battery cycling, mechanical deformation, and quantum behavior.

## IV. METHODS

### A. Experimental Materials and Methods

A cross-sectional STEM sample of a SrTiO<sub>3</sub> single-crystal substrate was prepared using a FEI Helios NanoLab DualBeam Focused Ion Beam (FIB) microscope and a standard lift out procedure. STEM data was collected using a probe-corrected JEOL GrandARM-300F microscope operating at 300 kV, with a convergence semi-angle of 29.7 mrad and estimated  $\sim 230$  pA probe current. EELS data was acquired using a GIF Quantum 665 spectrometer, with a spectrometer acceptance angle range of 113–273 mrad, a dispersion of 0.1 eV ch<sup>-1</sup>, and a dwell time ranging from 0.08–0.8 s px<sup>-1</sup>. Spectra were binned 130 $\times$  in the non-dispersive direction. Spectra were acquired by parking the probe on a different pristine region of the crystal for each experiment and then acquiring spectra continuously for 60–90 s.

### B. Computational Methods

Model development was done in Python 3.8. The Hyperspy 1.6.5 library was used to read and perform qualitative and quantitative analysis of EELS spectra. Numpy 1.19.4, pandas 1.2.0, scikit-learn 1.0.2, and matplotlib 3.2 were utilized for data processing, formatting, and visualization. Keras 2.4.3 and Tensorflow 2.4.1 were used for the LSTM model. Many iterations of the model were conducted using the hyperopt library, yielding the optimized hyperparameters shown in Supplementary Table II. Graphical user interface (GUI) development, as described in the Supplementary Information Figure 5, was implemented in Python 3.8. The Flask 2.0.1 and sqlalchemy 1.4.7 were used for the framework of the GUI.

## V. DATA AVAILABILITY

The EELS data used in this study are available on FigShare at <https://doi.org/10.6084/m9.figshare.20288730.v1>.

## VI. CODE AVAILABILITY

The code along with Jupyter notebooks used in this study is available on Gitlab at <https://github.com/pnnl/EELSTM>.

## VII. ACKNOWLEDGEMENTS

The authors would like to thank Dr. Jenna Pope for reviewing the manuscript. C.D., B.E.M., S.A., and S.R.S. were supported by the Chemical Dynamics Initiative/Investment, under the Laboratory Directed Research and Development (LDRD) Program at Pacific Northwest National Laboratory (PNNL). PNNL is a multi-program national laboratory operated for the U.S. Department of Energy (DOE) by Battelle Memorial Institute under Contract No. DE-AC05-76RL01830. Experimental sample preparation was performed at the Environmental Molecular Sciences Laboratory (EMSL), a national scientific user facility sponsored by the Department of Energy’s Office of Biological and Environmental Research and located at PNNL. EELS data was collected in the Radiological Microscopy Suite (RMS), located in the Radiochemical Processing Laboratory (RPL) at PNNL. N.L., Y.J., X.T., and V.S. were supported by the Data Intensive Research Enabling Clean Technology (DIRECT) National Science Foundation (NSF) National Research Traineeship (DGE-1633216), the State of Washington through the University of Washington (UW) Clean Energy Institute and the UW eScience Institute.

## VIII. AUTHOR CONTRIBUTIONS

S.A. and S.R.S. conceived and developed the project plan. N.L., Y.J., X.T., V.S., C.D., and S.A. implemented the machine learning approach. B.E.M. prepared the STEM sample and S.R.S. performed EELS analysis. All authors contributed to the writing and editing of the manuscript.

## **IX. COMPETING INTERESTS STATEMENT**

The authors declare no competing interests.

## REFERENCES

- <sup>1</sup>Jenny Cifuentes, Geovanny Marulanda, Antonio Bello, and Javier Reneses. Air Temperature Forecasting Using Machine Learning Techniques: A Review. *Energies*, 13(16):4215, aug 2020. URL: <https://www.mdpi.com/1996-1073/13/16/4215>, doi: [10.3390/en13164215](https://doi.org/10.3390/en13164215).
- <sup>2</sup>Eliana Vivas, Héctor Allende-Cid, and Rodrigo Salas. A Systematic Review of Statistical and Machine Learning Methods for Electrical Power Forecasting with Reported MAPE Score. *Entropy*, 22(12):1412, dec 2020. URL: <https://www.mdpi.com/1099-4300/22/12/1412>, doi: [10.3390/e22121412](https://doi.org/10.3390/e22121412).
- <sup>3</sup>Sotirios P. Chatzis, Vassilis Siakoulis, Anastasios Petropoulos, Evangelos Stavroulakis, and Nikos Vlachogiannakis. Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert Syst. Appl.*, 112:353–371, dec 2018. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0957417418303798>, doi: [10.1016/j.eswa.2018.06.032](https://doi.org/10.1016/j.eswa.2018.06.032).
- <sup>4</sup>Kalliopi Tsolaki, Thanasis Vafeiadis, Alexandros Nizamis, Dimosthenis Ioannidis, and Dimitrios Tzovaras. Utilizing machine learning on freight transportation and logistics applications: A review. *ICT Express*, feb 2022. URL: <https://linkinghub.elsevier.com/retrieve/pii/S2405959522000200>, doi: [10.1016/j.icte.2022.02.001](https://doi.org/10.1016/j.icte.2022.02.001).
- <sup>5</sup>Wilko Schwarting, Javier Alonso-Mora, and Daniela Rus. Planning and Decision-Making for Autonomous Vehicles. *Annu. Rev. Control. Robot. Auton. Syst.*, 1(1):187–210, may 2018. URL: <https://www.annualreviews.org/doi/10.1146/annurev-control-060117-105157>, doi: [10.1146/annurev-control-060117-105157](https://doi.org/10.1146/annurev-control-060117-105157).
- <sup>6</sup>Francisca Rosique, Pedro J. Navarro, Carlos Fernández, and Antonio Padilla. A Systematic Review of Perception System and Simulators for Autonomous Vehicles Research. *Sensors*, 19(3):648, feb 2019. URL: <http://www.mdpi.com/1424-8220/19/3/648>, doi: [10.3390/s19030648](https://doi.org/10.3390/s19030648).
- <sup>7</sup>Gopi Battineni, Getu Gamo Sagaro, Nalini Chinatalapudi, and Francesco Amenta. Applications of Machine Learning Predictive Models in the Chronic Disease Diagnosis. *J. Pers. Med.*, 10(2):21, mar 2020. URL: <https://www.mdpi.com/2075-4426/10/2/21>, doi: [10.3390/jpm10020021](https://doi.org/10.3390/jpm10020021).

- <sup>8</sup>Samuel Lalmuanawma, Jamal Hussain, and Lalrinfela Chhakchhuak. Applications of machine learning and artificial intelligence for Covid-19 (SARS-CoV-2) pandemic: A review. *Chaos, Solitons & Fractals*, 139:110059, oct 2020. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0960077920304562>, doi:10.1016/j.chaos.2020.110059.
- <sup>9</sup>Parag Ghorpade, Aditya Gadge, Akash Lende, Hitesh Chordiya, Gita Gosavi, Asima Mishra, Basavaraj Hooli, Yashwant S. Ingle, and Nuzhat Shaikh. Flood Forecasting Using Machine Learning: A Review. In *2021 8th Int. Conf. Smart Comput. Commun.*, pages 32–36. IEEE, jul 2021. URL: <https://ieeexplore.ieee.org/document/9528099/>, doi:10.1109/ICSCC51209.2021.9528099.
- <sup>10</sup>Pejman Tahmasebi, Serveh Kamrava, Tao Bai, and Muhammad Sahimi. Machine learning in geo- and environmental sciences: From small to large scale. *Adv. Water Resour.*, 142:103619, aug 2020. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0309170820301366>, doi:10.1016/j.advwatres.2020.103619.
- <sup>11</sup>Peter W. Hatfield, Jim A. Gaffney, Gemma J. Anderson, Suzanne Ali, Luca Antonelli, Suzan Başğmez du Pree, Jonathan Citrin, Marta Fajardo, Patrick Knapp, Brendan Kettle, Bogdan Kustowski, Michael J. MacDonald, Derek Mariscal, Madison E. Martin, Taisuke Nagayama, Charlotte A. J. Palmer, J. Luc Peterson, Steven Rose, J J Ruby, Carl Shneider, Matt J. V. Streeter, Will Trickey, and Ben Williams. The data-driven future of high-energy-density physics. *Nature*, 593(7859):351–361, may 2021. URL: <http://www.nature.com/articles/s41586-021-03382-w>, doi:10.1038/s41586-021-03382-w.
- <sup>12</sup>Alexander Radovic, Mike Williams, David Rousseau, Michael Kagan, Daniele Bonacorsi, Alexander Himmel, Adam Aurisano, Kazuhiro Terao, and Taritree Wongjirad. Machine learning at the energy and intensity frontiers of particle physics. *Nature*, 560(7716):41–48, aug 2018. URL: <http://www.nature.com/articles/s41586-018-0361-2>, doi:10.1038/s41586-018-0361-2.
- <sup>13</sup>Chao Zhang, Konstantin L. Firestein, Joseph F. S. Fernando, Dumindu Siriwardena, Joel E. Treifeldt, and Dmitri Golberg. Recent Progress of In Situ Transmission Electron Microscopy for Energy Materials. *Adv. Mater.*, 1904094:1904094, sep 2019. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201904094>, doi:10.1002/adma.201904094.
- <sup>14</sup>Haimei Zheng, Ying Shirley Meng, and Yimei Zhu. Frontiers of in situ electron microscopy. *MRS Bull.*, 40(1):12–18, jan 2015. URL: <http://www.>

- [journals.cambridge.org/abstract{ }S0883769414003054https://www.cambridge.org/core/product/identifier/S0883769414003054/type/journal{ }article](https://www.cambridge.org/core/product/identifier/S0883769414003054/type/journal-article), doi:10.1557/mrs.2014.305.
- <sup>15</sup>Mitra L. Taheri, Eric A. Stach, Ilke Arslan, P. A. Crozier, Bernd C. Kabius, Thomas LaGrange, Andrew M. Minor, Seiji Takeda, Mihaela Tanase, Jakob B. Wagner, and Renu Sharma. Current status and future directions for in situ transmission electron microscopy. *Ultramicroscopy*, 170:86–95, 2016. URL: <http://dx.doi.org/10.1016/j.ultramic.2016.08.007>, doi:10.1016/j.ultramic.2016.08.007.
- <sup>16</sup>Eric Stach, Brian DeCost, A Gilad Kusne, Jason Hattrick-Simpers, Keith A Brown, Kristofer G Reyes, Joshua Schrier, Simon Billinge, Tonio Buonassisi, Ian Foster, Carla P Gomes, John M Gregoire, Apurva Mehta, Joseph Montoya, Elsa Olivetti, Chiwoo Park, Eli Rotenberg, Semion K Saikin, Sylvia Smullin, Valentin Stanev, and Benji Maruyama. Autonomous experimentation systems for materials development: A community perspective. *Matter*, 4(9):2702–2726, sep 2021. URL: <https://doi.org/10.1016/j.matt.2021.06.036><https://linkinghub.elsevier.com/retrieve/pii/S2590238521003064>, doi:10.1016/j.matt.2021.06.036.
- <sup>17</sup>Steven R Spurgeon, Colin Ophus, Lewys Jones, Amanda Petford-Long, Sergei V Kalinin, Matthew J Olszta, Rafal E. Dunin-Borkowski, Norman Salmon, Khalid Hattar, Weichang D Yang, Renu Sharma, Yingge Du, Ann Chiaramonti, Haimei Zheng, Edgar C Buck, Libor Kovarik, R Lee Penn, Dongsheng Li, Xin Zhang, Mitsuhiro Murayama, and Mitra L. Taheri. Towards data-driven next-generation transmission electron microscopy. *Nat. Mater.*, 20(3):274–279, mar 2021. URL: <http://dx.doi.org/10.1038/s41563-020-00833-z><http://www.nature.com/articles/s41563-020-00833-z>, doi:10.1038/s41563-020-00833-z.
- <sup>18</sup>Matthew Olszta, Derek Hopkins, Kevin R. Fiedler, Marjolein Oostrom, Sarah Akers, and Steven R. Spurgeon. An Automated Scanning Transmission Electron Microscope Guided by Sparse Data Analytics. *Microsc. Microanal.*, pages 1–11, jun 2022. URL: <http://arxiv.org/abs/2109.14772><https://www.cambridge.org/core/product/identifier/S1431927622012065/type/journal-article>, arXiv:2109.14772, doi:10.1017/S1431927622012065.
- <sup>19</sup>Yongtao Liu, Kyle P. Kelley, Rama K. Vasudevan, Hiroshi Funakubo, Maxim A. Ziatdinov, and Sergei V. Kalinin. Experimental discovery of structure–property rela-

- tionships in ferroelectric materials via active learning. *Nat. Mach. Intell.*, 4(4):341–350, apr 2022. URL: <https://www.nature.com/articles/s42256-022-00460-0>, doi: [10.1038/s42256-022-00460-0](https://doi.org/10.1038/s42256-022-00460-0).
- <sup>20</sup>Sergei V. Kalinin, Maxim Ziatdinov, Jacob Hinkle, Stephen Jesse, Ayana Ghosh, Kyle P. Kelley, Andrew R. Lupini, Bobby G. Sumpter, and Rama K. Vasudevan. Automated and Autonomous Experiments in Electron and Scanning Probe Microscopy. *ACS Nano*, 15(8):12604–12627, aug 2021. URL: <https://pubs.acs.org/doi/10.1021/acsnano.1c02104>, arXiv:2103.12165, doi:10.1021/acsnano.1c02104.
- <sup>21</sup>Catherine K. Groschner, Christina Choi, and Mary C. Scott. Machine Learning Pipeline for Segmentation and Defect Identification from High-Resolution Transmission Electron Microscopy Data. *Microsc. Microanal.*, 27(3):549–556, 2021. arXiv:2001.05022, doi: [10.1017/S1431927621000386](https://doi.org/10.1017/S1431927621000386).
- <sup>22</sup>Robbie Sadre, Colin Ophus, Anastasiia Butko, and Gunther H. Weber. Deep Learning Segmentation of Complex Features in Atomic-Resolution Phase-Contrast Transmission Electron Microscopy Images. *Microsc. Microanal.*, 27(4):804–814, 2021. arXiv:2012.05322, doi:10.1017/S1431927621000167.
- <sup>23</sup>James P. Horwath, Dmitri N. Zakharov, Rémi Mégret, and Eric A. Stach. Understanding important features of deep learning models for segmentation of high-resolution transmission electron microscopy images. *npj Comput. Mater.*, 6(1):108, dec 2020. URL: <http://dx.doi.org/10.1038/s41524-020-00363-x>, doi:10.1038/s41524-020-00363-x.
- <sup>24</sup>Chenyu Zhang, Jie Feng, Luis Rangel DaCosta, and Paul.M. Voyles. Atomic resolution convergent beam electron diffraction analysis using convolutional neural networks. *Ultramicroscopy*, 210(October 2019):112921, mar 2020. URL: <https://doi.org/10.1016/j.ultramic.2019.112921><https://linkinghub.elsevier.com/retrieve/pii/S0304399119303511>, doi:10.1016/j.ultramic.2019.112921.
- <sup>25</sup>Weizong Xu and J.M. LeBeau. A deep convolutional neural network to analyze position averaged convergent beam electron diffraction patterns. *Ultramicroscopy*, 188:59–69, may 2018. URL: <http://arxiv.org/abs/1708.00855><https://linkinghub.elsevier.com/retrieve/pii/S0304399117303340>, arXiv:1708.00855, doi:10.1016/j.ultramic.2018.03.004.

- <sup>26</sup>Jacob Madsen, Pei Liu, Jens Kling, Jakob Birkedal Wagner, Thomas Willum Hansen, Ole Winther, and Jakob Schiøtz. A Deep Learning Approach to Identify Local Structures in Atomic-Resolution Transmission Electron Microscopy Images. *Adv. Theory Simulations*, 1(8):1800037, aug 2018. URL: <http://doi.wiley.com/10.1002/adts.201800037>, [arXiv:1802.03008](https://arxiv.org/abs/1802.03008), [doi:10.1002/adts.201800037](https://doi.org/10.1002/adts.201800037).
- <sup>27</sup>Kevin Kaufmann, Hobson Lane, Xiao Liu, and Kenneth S. Vecchio. Efficient few-shot machine learning for classification of EBSD patterns. *Sci. Rep.*, 11(1):8172, dec 2021. URL: <https://doi.org/10.1038/s41598-021-87557-5><http://www.nature.com/articles/s41598-021-87557-5>, [doi:10.1038/s41598-021-87557-5](https://doi.org/10.1038/s41598-021-87557-5).
- <sup>28</sup>Christina Doty, Shaun Gallagher, Wenqi Cui, Wenya Chen, Shweta Bhushan, Marjolein Oostrom, Sarah Akers, and Steven R. Spurgeon. Design of a graphical user interface for few-shot machine learning classification of electron microscopy data. *Comput. Mater. Sci.*, 203:111121, feb 2022. URL: <http://arxiv.org/abs/2107.10387><https://linkinghub.elsevier.com/retrieve/pii/S0927025621007874>, [arXiv:2107.10387](https://arxiv.org/abs/2107.10387), [doi:10.1016/j.commatsci.2021.111121](https://doi.org/10.1016/j.commatsci.2021.111121).
- <sup>29</sup>Sarah Akers, Elizabeth Kautz, Andrea Trevino-Gavito, Matthew Olszta, Bethany E. Matthews, Le Wang, Yingge Du, and Steven R. Spurgeon. Rapid and flexible segmentation of electron microscopy data using few-shot machine learning. *npj Comput. Mater.*, 7(1):187, dec 2021. URL: <https://www.nature.com/articles/s41524-021-00652-z>, [doi:10.1038/s41524-021-00652-z](https://doi.org/10.1038/s41524-021-00652-z).
- <sup>30</sup>Lei Yu, Bethany M. Hudak, Ahamed Ullah, Melonie P. Thomas, Chloe C. Porter, Ayanthi Thisera, Rose H. Pham, Manisha De Alwis Goonatilleke, and Beth S. Guiton. Unveiling the Microscopic Origins of Phase Transformations: An in Situ TEM Perspective. *Chem. Mater.*, 32(2):639–650, jan 2020. URL: <https://pubs.acs.org/doi/10.1021/acs.chemmater.9b03360>, [doi:10.1021/acs.chemmater.9b03360](https://doi.org/10.1021/acs.chemmater.9b03360).
- <sup>31</sup>Jeffrey M. Ede. Deep learning in electron microscopy. *Mach. Learn. Sci. Technol.*, 2(1):011004, mar 2021. URL: <http://arxiv.org/abs/2009.08328><https://iopscience.iop.org/article/10.1088/2632-2153/abd614>, [arXiv:2009.08328](https://arxiv.org/abs/2009.08328), [doi:10.1088/2632-2153/abd614](https://doi.org/10.1088/2632-2153/abd614).
- <sup>32</sup>Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Comput.*, 31(7):1235–1270, jul 2019. URL: <https://direct.mit.edu/neco/article/31/7/1235-1270/8500>, [doi:10.1162/NECO-2018-0111](https://doi.org/10.1162/NECO-2018-0111).

- [10.1162/neco\\_a\\_01199](https://doi.org/10.1162/neco_a_01199).
- <sup>33</sup>Connor Robertson, Jared L. Wilmoth, Scott Retterer, and Miguel Fuentes-Cabrera. Performing Video Frame Prediction of Microbial Growth with a Recurrent Neural Network. pages 1–15, 2022. URL: <http://arxiv.org/abs/2205.05810>, [arXiv:2205.05810](https://arxiv.org/abs/2205.05810).
- <sup>34</sup>Jeffrey M. Ede. Adaptive Partial Scanning Transmission Electron Microscopy with Reinforcement Learning. *arXiv*, (1), apr 2020. URL: <http://arxiv.org/abs/2004.02786>, [arXiv:2004.02786](https://arxiv.org/abs/2004.02786), [doi:10.48550/arXiv.2004.02786](https://doi.org/10.48550/arXiv.2004.02786).
- <sup>35</sup>Marijn F Stollenga, Wonmin Byeon, Marcus Liwicki, and Jürgen Schmidhuber. Parallel Multi-Dimensional LSTM, With Application to Fast Biomedical Volumetric Image Segmentation. In C Cortes, N Lawrence, D Lee, M Sugiyama, and R Garnett, editors, *Adv. Neural Inf. Process. Syst.*, volume 28. Curran Associates, Inc., 2015. URL: <https://proceedings.neurips.cc/paper/2015/file/d43ab110ab2489d6b9b2caa394bf920f-Paper.pdf>.
- <sup>36</sup>S.R. Spurgeon and S.A. Chambers. Atomic-Scale Characterization of Oxide Interfaces and Superlattices Using Scanning Transmission Electron Microscopy. In *Encycl. Interfacial Chem.*, pages 38–48. Elsevier, New York, 2018. URL: <http://linkinghub.elsevier.com/retrieve/pii/B978012409547212877X><https://linkinghub.elsevier.com/retrieve/pii/B978012409547212877X>, [doi:10.1016/B978-0-12-409547-2.12877-X](https://doi.org/10.1016/B978-0-12-409547-2.12877-X).
- <sup>37</sup>Steven R Spurgeon. Order-disorder behavior at thin film oxide interfaces. *Curr. Opin. Solid State Mater. Sci.*, 24(6):100870, dec 2020. URL: <https://doi.org/10.1016/j.cossms.2020.100870><https://linkinghub.elsevier.com/retrieve/pii/S1359028620300681>, [doi:10.1016/j.cossms.2020.100870](https://doi.org/10.1016/j.cossms.2020.100870).
- <sup>38</sup>Lirong Yao and Yazhuo Guan. An Improved LSTM Structure for Natural Language Processing. *Proceedings of 2018 IEEE International Conference of Safety Produce Informatization, IICSPI 2018*, pages 565–569, apr 2019. [doi:10.1109/IICSPI.2018.8690387](https://doi.org/10.1109/IICSPI.2018.8690387).
- <sup>39</sup>Shalini Ghosh, Oriol Vinyals, Brian Strope, Scott Roy, Tom Dean, and Larry Heck. Contextual LSTM (CLSTM) models for Large scale NLP tasks. feb 2016. URL: <https://arxiv.org/abs/1602.06291v2>, [arXiv:1602.06291](https://arxiv.org/abs/1602.06291), [doi:10.48550/arxiv.1602.06291](https://doi.org/10.48550/arxiv.1602.06291).
- <sup>40</sup>Seong Hyeon Park, Byeongdo Kim, Chang Mook Kang, Chung Choo Chung, and Jun Won Choi. Sequence-to-Sequence Prediction of Vehicle Trajectory via LSTM Encoder-Decoder Architecture. *IEEE Intelligent Vehicles Symposium, Proceedings*, 2018-June:1672–1678,

- oct 2018. [arXiv:1802.06338](https://arxiv.org/abs/1802.06338), [doi:10.1109/IVS.2018.8500658](https://doi.org/10.1109/IVS.2018.8500658).
- <sup>41</sup>Yuting Yang, Junyu Dong, Xin Sun, Estanislau Lima, Quanquan Mu, and Xinhua Wang. A CFCC-LSTM Model for Sea Surface Temperature Prediction. *IEEE Geoscience and Remote Sensing Letters*, 15(2):207–211, feb 2018. [doi:10.1109/LGRS.2017.2780843](https://doi.org/10.1109/LGRS.2017.2780843).
- <sup>42</sup>Sujan Ghimire, Ravinesh C. Deo, Hua Wang, Mohanad S. Al-Musaylh, David Casillas-Pérez, and Sancho Salcedo-Sanz. Stacked LSTM Sequence-to-Sequence Autoencoder with Feature Selection for Daily Solar Radiation Prediction: A Review and New Modeling Results. *Energies* 2022, Vol. 15, Page 1061, 15(3):1061, jan 2022. URL: <https://www.mdpi.com/1996-1073/15/3/1061/html><https://www.mdpi.com/1996-1073/15/3/1061>, [doi:10.3390/EN15031061](https://doi.org/10.3390/EN15031061).
- <sup>43</sup>Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, nov 1997. [doi:10.1162/NECO.1997.9.8.1735](https://doi.org/10.1162/NECO.1997.9.8.1735).
- <sup>44</sup>Alexander Gubbens, Melanie Barfels, Colin Trevor, Ray Twesten, Paul Mooney, Paul Thomas, Nanda Menon, Bernd Kraus, Chengye Mao, and Brian McGinn. The GIF Quantum, a next generation post-column imaging energy filter. *Ultramicroscopy*, 110(8):962–970, jul 2010. URL: <http://dx.doi.org/10.1016/j.ultramic.2010.01.009><http://linkinghub.elsevier.com/retrieve/pii/S0304399110000124>, [doi:10.1016/j.ultramic.2010.01.009](https://doi.org/10.1016/j.ultramic.2010.01.009).
- <sup>45</sup>James L. Hart, Andrew C. Lang, Asher C. Leff, Paolo Longo, Colin Trevor, Ray D. Twesten, and Mitra L. Taheri. Direct Detection Electron Energy-Loss Spectroscopy: A Method to Push the Limits of Resolution and Sensitivity. *Sci. Rep.*, 7(1):8243, 2017. URL: <http://www.nature.com/articles/s41598-017-07709-4>, [doi:10.1038/s41598-017-07709-4](https://doi.org/10.1038/s41598-017-07709-4).
- <sup>46</sup>Nicholas R. Lewis, John D. Hedengren, and Eric L. Haseltine. Hybrid Dynamic Optimization Methods for Systems Biology with Efficient Sensitivities. *Processes* 2015, Vol. 3, Pages 701-729, 3(3):701–729, sep 2015. URL: <https://www.mdpi.com/2227-9717/3/3/701/html><https://www.mdpi.com/2227-9717/3/3/701>, [doi:10.3390/PR3030701](https://doi.org/10.3390/PR3030701).
- <sup>47</sup>Steven R. Spurgeon, Tiffany C. Kaspar, Vaithiyalingam Shutthanandan, Jonathan Gigax, Lin Shao, and Michel Sassi. Asymmetric Lattice Disorder Induced at Oxide Interfaces. *Adv. Mater. Interfaces*, 7(8):1901944, apr 2020. URL: <http://arxiv.org/abs/1904.05932><https://onlinelibrary.wiley.com/doi/abs/10.1002/admi.201901944>, [arXiv:1904.05932](https://arxiv.org/abs/1904.05932), [doi:10.1002/admi.201901944](https://doi.org/10.1002/admi.201901944).

## X. SUPPLEMENTARY INFORMATION

### A. Optimized Hyperparameters

TABLE II. Optimized hyperparameters for the EELSTM model.

Hyperparameter	Search Space	Optimal Value
Learning Rate	$[\cdot 10^{-6}, \cdot 10^{-1}]$	$\cdot 10^{-4}$
LSTM Layers	[1,4]	3
Nodes in Layer	[64,512]	256
Dropout	[0.01,0.3]	0.2
Input sequence window	[3,15]	8
Batch size	[20,100]	40

## B. Graphical User Interface

The model described in the main text is linked to a graphical user interface (GUI) built using the Python Flask web framework, shown in Figure 5. This interface provides real-time visualization of raw data and model forecasts. The GUI includes four steps: (1) **Data Import**, (2) **Spectrum Visualization and Model Selection**, (3) **Forecast Display**, and (4) **Forecast History**. In (1) the user selects an appropriate EELS spectrum image (SI) data in the native Gatan \*.dm4 format. This data should contain the EELS datacube in a time-series. The backend of the GUI uses HyperSpy to parse the individual EELS spectra. After selecting the spectrum, the user is sent to (2), where they can visualize the SI time series dynamically via a slider bar. They can also input the dwell time and model parameters. After submitting the desired parameters, the user is presented with a (3) forecast, input sequences, ground truth data (if available), and MSE. Finally, (4) catalogs the history of prior forecasts, allowing a user to easily revisit previous spectra and results.

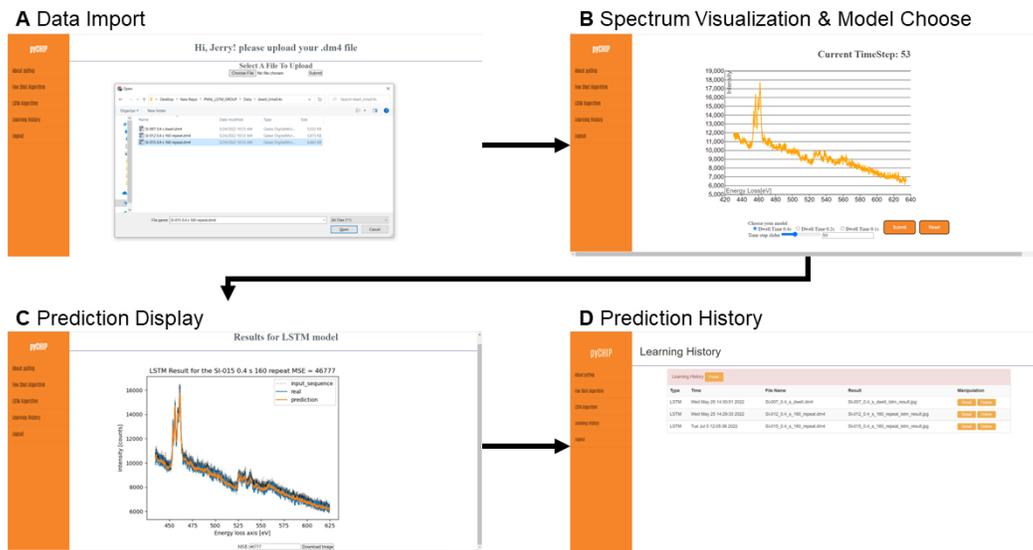


FIG. 5. **Graphical user interface (GUI) for interacting with the EELSTM model.** (a–d) Usage steps include Data Import, Spectrum Visualization and Model Selection, Forecast Display, and Forecast History, respectively. **Replace letters in figure with lower case bold.**

### C. Autocorrelation

In some cases, the last input in a sequence is a better predictor of future sequences than an LSTM model. In light of this, autocorrelation calculations between the EELS spectra inform how many timesteps in advance must be predicted to ensure actionable predictions. Figure 6 shows the result of the autocorrelation calculation from the dataset shown in the main text. The shaded area represents number of input timesteps where the autocorrelation value falls into an acceptable range, whereas the blue dots represent the autocorrelation values for each input timestep. The smaller the autocorrelation value, the less likely that the model copies seasonal trends into its prediction and the more accurate the prediction will be. As shown in this plot, the minimum preferred number of input timestep is 6 for our LSTM model, and the autocorrelation value converges to zero starting at 10 timesteps. This also verifies that our data are not subject to explicit autocorrelation.

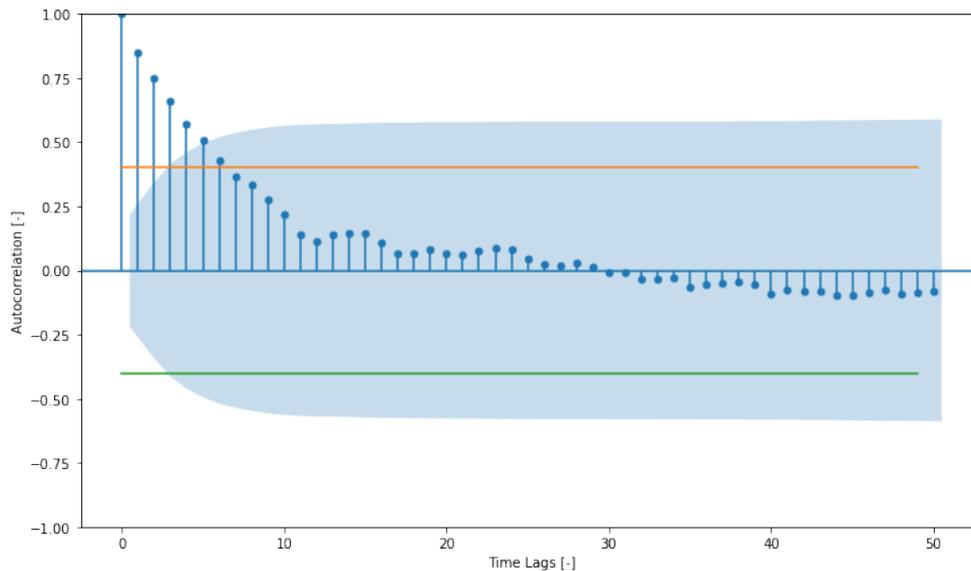


FIG. 6. **Autocorrelation plot from the dataset shown in the main text.** The shaded area represents the 95 % confidence interval. The horizontal lines depict an autocorrelation of  $\pm 0.4$ , which is the preferred maximum range for our model. Both the autocorrelation and the time axes are unitless.