ARTICLE OPEN

Check for updates

Forecasting of in situ electron energy loss spectroscopy

Nicholas R. Lewis[1], Yicheng Jin[1], Xiuyu Tang[1], Vidit Shah[1], Christina Doty[2], Bethany E. Matthews [3], Sarah Akers [2] and Steven R. Spurgeon [3,4✉]

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 in situ 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.

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[1] to the power grid[2] and from stock trading[3] to logistics[4]. The recent rise of autonomous vehicles, including automobiles, drones, and spacecraft, has been propelled by advanced forecasting models deployed on high-performance computing platforms[5,6]. Abundant low-cost computing and the proliferation of machine learning (ML) have enabled many 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[7,8], climate science[9,10], and high-energy physics[11,12]. 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[13–15]. While advanced hardware can easily generate large volumes of data, our ability to interpret, anticipate, and automatically act on such data is limited[16,17]. 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 low-level control, descriptive models, and forecasting (prescriptive) approaches to implement more powerful decision-making.

Recently, significant progress has been made toward microscope automation platforms that allow for centralized, data-driven control of instrument operations[18–20]. Collected data is then typically passed through two main kinds of descriptive models: those based on neural networks fed large volumes of hand-labeled or simulated examples[21–26], or those based on few-shot approaches utilizing sparse, canonical examples[27]. We have previously demonstrated[28,29] 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 that enable us to anticipate changes in data streams.

Currently, a wide range of experiments, such as studying heating- or beam-induced phase transitions, tracking of particles and reaction fronts, or operando switching of ferroic and quantum materials, is difficult or impossible to conduct[30]. In situ electron energy loss spectroscopy (EELS), in particular, is one of the highest-resolution chemical analysis techniques in the STEM, but it is prone to artifacts and latency. This technique measures the inelastic energy loss experienced by an incident electron probe upon interaction with the atoms in a material. The energy loss results from primary electron interactions with weakly bound outer-shell electrons, as well as inner-shell ionization. Using this technique, it is possible to probe the local density of unoccupied states at sub-nanometer spatial resolution, pro- viding a powerful means to chemically fingerprint a material during phase transformations. However, 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

[1]Department of Chemical Engineering, University of Washington, Seattle, WA 98195, USA. [2]National Security Directorate, Pacific Northwest National Laboratory, Richland, WA 99352, USA. [3]Energy and Environment Directorate, Pacific Northwest National Laboratory, Richland, WA 99352, USA. [4]Department of Physics, University of Washington, Seattle, WA 98195, USA. ✉email: steven.spurgeon@pnnl.gov
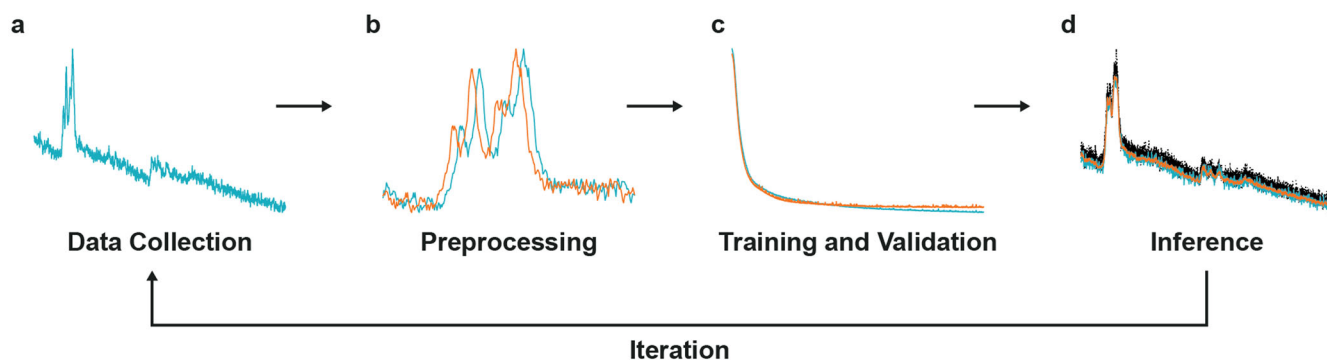
Fig. 1 Overview of the EELSTM model workflow. a–d Steps include data collection, preprocessing, training and validation, and inference, respectively.

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, such recurrent neural networks (RNNs), gated recurrent units (GRUs)[31], and, more recently, transformer models[32,33]. Among the former, long short-term memory (LSTMs) are commonly used to incorporate knowledge of past experiences to model Markov-type decision processes[34]. LSTM models have been extensively applied to serial data, such as text, audio, and video[35], and to materials science problems such as prediction of switching in ferroelectrics[36]. Another common time-series model is autoregressive integrated moving average (ARIMA);[37] however, this type of model is built around the assumption that future values resemble past trends with a type of periodic behavior, such as in financial forecasts.

Since the nature of this problem is often irreversible change imparted to a sample, the advantages of ARIMA models are poorly suited. Based on the shortcomings of other models, and recent successes with LSTMs used to predict time series on other types of STEM data[38], we chose to focus on LSTMs to determine suitability for this type of problem. 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[39] and segmentation of biological images[40]. Given that in situ STEM data are acquired in serial fashion, we aim to evaluate the performance of LSTM for microscope data, with an eye toward practical implementation.

Here we describe an LSTM approach for forecasting of in situ 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[41]. Prior work has also demonstrated the potential for real-time ML-based denoising and classification of in situ EELS data[42]. We explore the crystalline-to-amorphous phase transition in the archetypal perovskite oxide $SrTiO_3$ (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[43]. We systematically explore data preprocessing, model architecture, hyperparameter optimization, training, and validation relative to ground truth experimental data. We emphasize that the choice of parameters provided is not exhaustive, but instead is representative of a common beam-induced phase transition in complex oxides. Nonetheless, we show that this model has good 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 and provide our code to spur adoption of this approach.

RESULTS AND DISCUSSION

The chief aim of the present study is to adapt existing LSTM models, such as those used for natural language processing[44,45] and time series prediction of physical phenomena[46–48], to the task of EELS forecasting. As shown in Fig. 1, the EELSTM model workflow encompasses four steps: *Data Collection, Preprocessing, Training and Validation,* and *Inference*. In Section "Data Collection", we describe the experimental setup and EELS data acquisition, including considerations for the best model performance. In Section "Preprocessing", we review preprocessing strategies specific to EELS data, resulting from the data collection process, variability between experiments, and the nature of core-loss data itself. In Section "Training and Validation", we describe the training and validation process, including the relationship between training inputs and predictions, model transferability, and temporal correlations. Lastly, in Section "Inference and Benchmarking", we discuss possible error metrics and benchmark performance relative to ground truth experimental data.

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^{-1}, while keeping all other instrument parameters constant. We observed that consistency in operating conditions (such as beam energy, dwell time/dose, and sample configuration) 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 depends on dwell time and the damage rate (and associated phase transition) varies with dose conditions. We conducted three different experiments under similar conditions to obtain the training and test datasets. Two experiments' worth of data were used to construct the training dataset, while a third experiment was held back as a test set. Each experiment contained ~160 spectra, ranging from the first spectrum with a fresh sample to the final degraded sample spectrum; each spectrum contained 2048 energy channels. The plots and MSE values presented in the main text used a dwell time of 0.4 s px^{-1}, while those shown in the Supplementary used a dwell time of 0.8 s px^{-1}. This train/test split 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
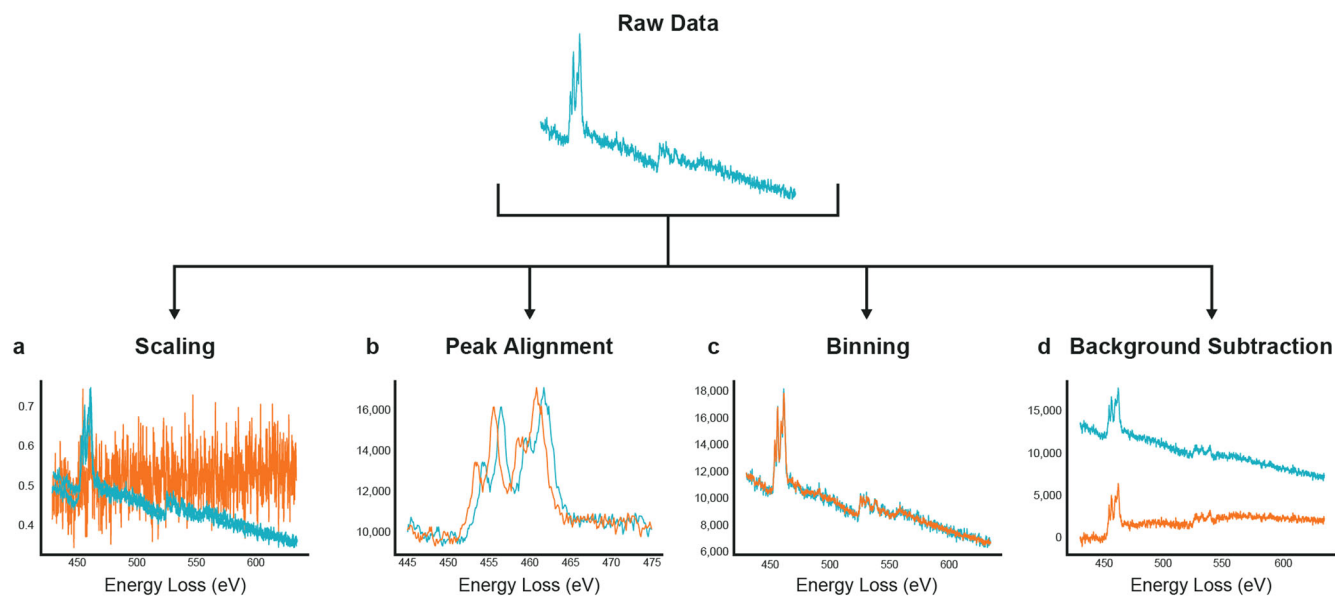
Raw Data

Fig. 2 **Data preprocessing strategies. 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.

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. This construction also allows for extrapolation to future experiments, where predictions can be made independent of the progression of the phase transition. We note that such calibration runs are important to ensure successful model forecasting, as we discuss below.

Preprocessing

We next consider data preprocessing, which includes iterative steps specific to in situ EELS data, as shown in Fig. 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 (domain) considerations, we identified a variety of preprocessing strategies to improve training and accuracy. Figure 2 shows an overview of these preprocessing strategies and we next discuss their rationale, implementation, and effect on model predictions.

Scaling. The first data preprocessing step 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[34]. The derivatives of these functions diminish greatly outside of this input range. As a result, training weights receive small updates based on the gradient when the input data is outside of [0, 1][49]. We can consider several ways to scale the data, knowing that raw intensity counts range from the thousands to tens of thousands. The most logical, due to the inherent data structure, is a min-max scaling where the training set is scaled between 0 and 1 using the minimum and maximum 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 Fig. 2a). This latter approach is desirable, since it ensures that a given energy bin spans the whole range of [0,1], rather than only a fractional range that would potentially introduce unintended sensitivities to model weights.

The raw data were scaled using the minimum and maximum values of one of the training datasets and formatted into the sequence/output format expected by the LSTM. We then passed the formatted data through 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. Interestingly, we observed that both scaling methods yielded similar errors, indicating that the channel-to-channel relationship did not need to be maintained for the model to perform optimally. 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× 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 "Inference and Benchmarking". 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 ± 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.

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[50]. 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 use one of the timesteps from the test spectra as a reference. We utilize the peak alignment functionality of the Hyperspy Python library to align all spectra to this reference spectrum[51]. A hard limit for spectral shifts can also be applied in the Hyperspy package to account for spurious energy shifts due to high-tension instability, while minimizing the overall realignment of spectra. 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 Fig. 2b, the raw spectrum had 2048 channels prior to alignment. After alignment, which typically lost ≤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 ± 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 ~456 eV and O K edge at ~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 strategy proved important for more limited amounts of training data.

Binning. 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[52]. 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 Fig. 2c. However, the reduction in training data, particularly when using the exclusive bins method, actually yielded worse results (RMSE of 400.8 ± 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 ± 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.

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 Fig. 2d. It was necessary to perform background subtraction before any scaling, since the background subtraction shifted the entire baseline.

Best EELS practice dictates specifying the region for background subtraction to be taken before the edge of interest[53,54]. This approach proved problematic for this particular study, since we aimed to predict the RMSE of 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. We emphasize this done only to evaluate the accuracy of the prediction and that individual, raw spectra with background subtraction prior to each edge should be conducted for EELS quantification. After training the model on background-subtracted data, we observed decreased performance with higher variability (RMSE of 392.4 ± 177.4). We also observed that models trained on this type of data had a greater tendency to overfit. This behavior may be explained by the fact that background subtraction constrains the data to a narrower range of values, and after scaling, the model 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 this approach would perhaps be more useful when utilizing different error metrics. This approach 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 to achieve the best results in the current implementation. In summary, we can see that individual preprocessing steps can improve RMSE by nearly 88.9% relative to using raw spectra, as shown in Table 1.

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 Fig. 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 analyzed input and

Table 1. Changes in RMSE with preprocessing strategy.

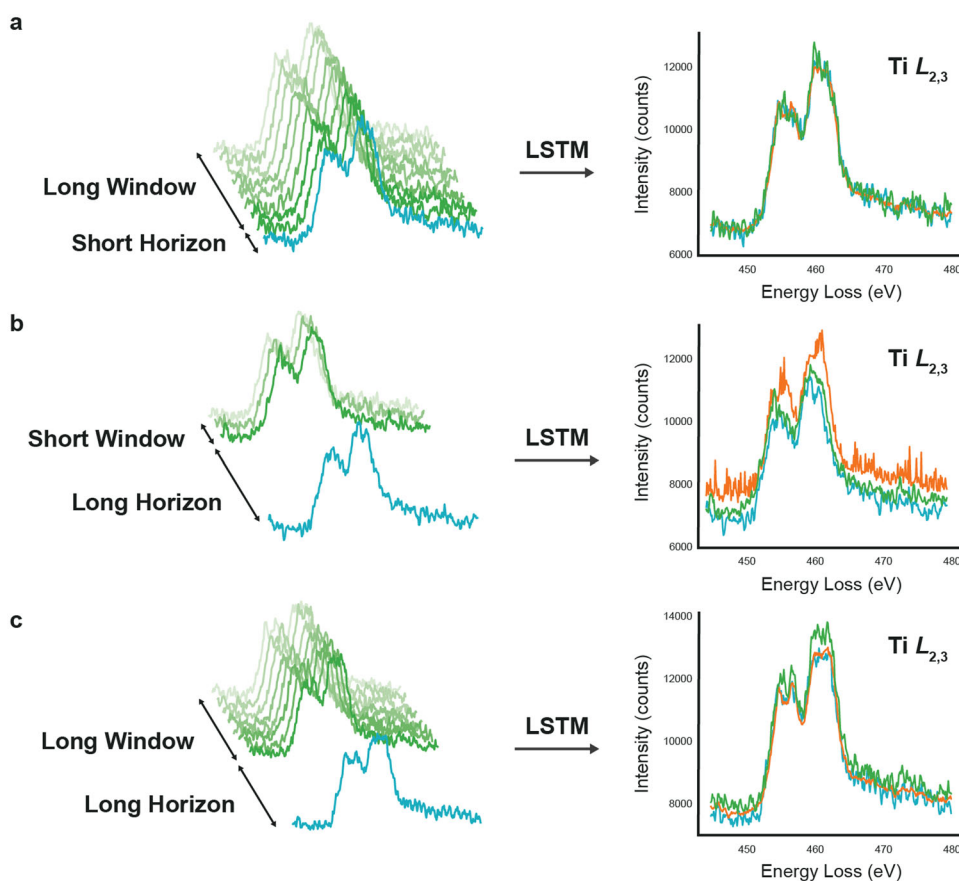| Strategy	RMSE	Improvement (%)
Raw	1958.3	—
Scaling	295.5	84.9
Peak Alignment	217.9	88.9
Binning	400.8	79.5
Background Subtraction	392.4	80.0

Fig. 3 Input window and output horizon forecast scenarios. a "Long window, short horizon" scenario with an eight timestep window and one timestep horizon, demonstrating deceptively good results due to autocorrelation. **b** "Short window, long horizon" scenario with a 3 timestep window and 8 timestep horizon, demonstrating poor results. **c** "Long window, long horizon" scenario, with an 8 timestep window and 8 timestep horizon, indicating good prediction with minimal autocorrelation. The green lines indicate spectra from the input window, the blue lines indicate true future spectra to predict, and the orange line indicates spectra predicted by the LSTM. Note that the plots on the right show only the last spectrum from the input window (in green).

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 its prohibitive computational time and memory, as well as the positive results using the first approach.

The next consideration for the model revolved around deciding on a prediction horizon. In time series data, 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 window as the next timestep, leading to artificially inflated prediction accuracy. For example, in the "long window, short horizon" scenario in Fig. 3a, with an 8 timestep window and 1 timestep horizon, the prediction appears to have extremely high fidelity. However, this result is not significantly better than simply using the last spectrum from the input window as the prediction. To account for this possibility, we performed a Pearson autocorrelation calculation between a spectrum at a given timestep and all subsequent spectra, as shown in Supplementary Fig. 1. As anticipated, we observed a high correlation among timesteps proximal to each other, with a steadily de- creasing correlation at a further horizon. After ~6–8 timesteps, the correlation between spectra was statistically insignificant to ensure that the model would not simply learn the most recent spectrum from the input regardless of the qualitative similarity, as shown in Supplementary Fig. 1. For an EELS dwell time of 0.4 s px^{-1}, this corresponds to ~3 s and is an important consideration for any practical usage.

The length of the input window was optimized along with other model hyperparameters in the range of 3–15 timesteps, as shown in Supplementary Table 1. Naturally, a sufficiently long sequence of spectra must be used to establish the progression of the STO phase transformation, which can then be used to predict future timesteps. We observe poor results when a short input window is used, as shown in the "short window, long horizon" scenario in Fig. 3b with a 3 timestep window and 8 timestep horizon. However, we also wish to avoid excessively long input windows to maintain sensitivity to rapid changes in the data. In the context of automation, a longer input window 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 window, long horizon" scenario as shown in Fig. 3c, with an 8 timestep window and 8 timestep horizon, indicating good prediction with minimal autocorrelation. Additional hyperparameter optimization was performed with the Hyperopt Python package, 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 window and horizon may be necessary depending on the exact parameters of an EELS experiment and the nature of the dynamic behavior under study.

Inference and benchmarking

Lastly, we consider inference and benchmarking of model performance at different stages of an STO phase transition. The inference step (Fig. 1d) takes an input window 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 an 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), 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 an 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[55]. 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. While MSE and RMSE have limitations, they are directly interpretable: a RMSE of 200 indicates that, on average, each energy channel is 200 intensity counts off, whether it is in a region of interest or background. Using these metrics also allows us to more readily compare values with future research, including different forecasting models, materials with different regions of interest, or cropped spectra. The error values reported throughout this study represent error values between the ground truth raw data and the model prediction *after* any post-processing is done to revert the prediction back to raw data format, such as the inverse scaling transform.

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 "Preprocessing," and the set of hyperparameters determined in Supplementary Table 1. Predictions are made using an 8 timestep window, with an 8 timestep horizon, and then overlaid against ground truth (raw) data. We first consider Fig. 4a, which shows the initial stages ($t \approx 15$ s) of the phase transition across the whole spectrum and two regions of interest, the Ti $L_{2,3}$ and O K edges. We note that because both

edges are collected simultaneously, relative chemical shifts can still be accurately measured. We observe a good 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[56]. Next, we consider a later stage in the phase transition, shown in Fig. 4b. At this time ($t \approx 60$ 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[57]. Here again, we observe strong predictive capability across the full spectrum (MSE = 181.4), pointing to the ability of the model to effectively capture the future state of a phase transition.

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 good predictive power relative to ground truth experimental data, but that there are important pre-processing strategies and forecast parameters 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. We may envision other scenarios in which this model may be useful. For example, a portion of a phase transition may be triggered and then stopped to minimize sample changes, while the model may be used to predict the completed state of the reaction. Alternatively, calibration using a sacrificial part of the sample would allow us to define control envelopes for a self-driving experiment to minimize unnecessary repetition and capture underlying mechanisms.

As already described, a central challenge of in situ electron microscopy is the ability to anticipate and respond to changing noisy and high-velocity data. Forecasting models of the kind shown in this work will find important usage in emerging self-driving microscope platforms. 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[18]. The ability to run these models in real-time and predict a future state of a chemical reaction will allow for optimization of experimental parameters, such as beam electron dose, sampling, and current, which are captured metadata alongside the EELS spectra. Such feedback from the model forecast may then be implemented in emerging closed-loop instrument controllers that can respond faster than any human. In turn, these capabilities will realize richer, more accurate studies of fundamental phase transitions for fundamental studies of crystal nucleation and growth, battery cycling, mechanical testing, and quantum behavior.

METHODS

Experimental materials and methods

A cross-sectional STEM sample of a $SrTiO_3$ single-crystal substrate was prepared using a FEI Helios NanoLab DualBeam Focused Ion Beam (FIB) microscope and a standard lift-out procedure. STEM data were collected using a probe-corrected JEOL GrandARM-300F microscope operating at 300 kV, with a convergence semi-angle of 29.7 mrad and estimated ~230 pA probe current. EELS data was acquired using a GIF Quantum
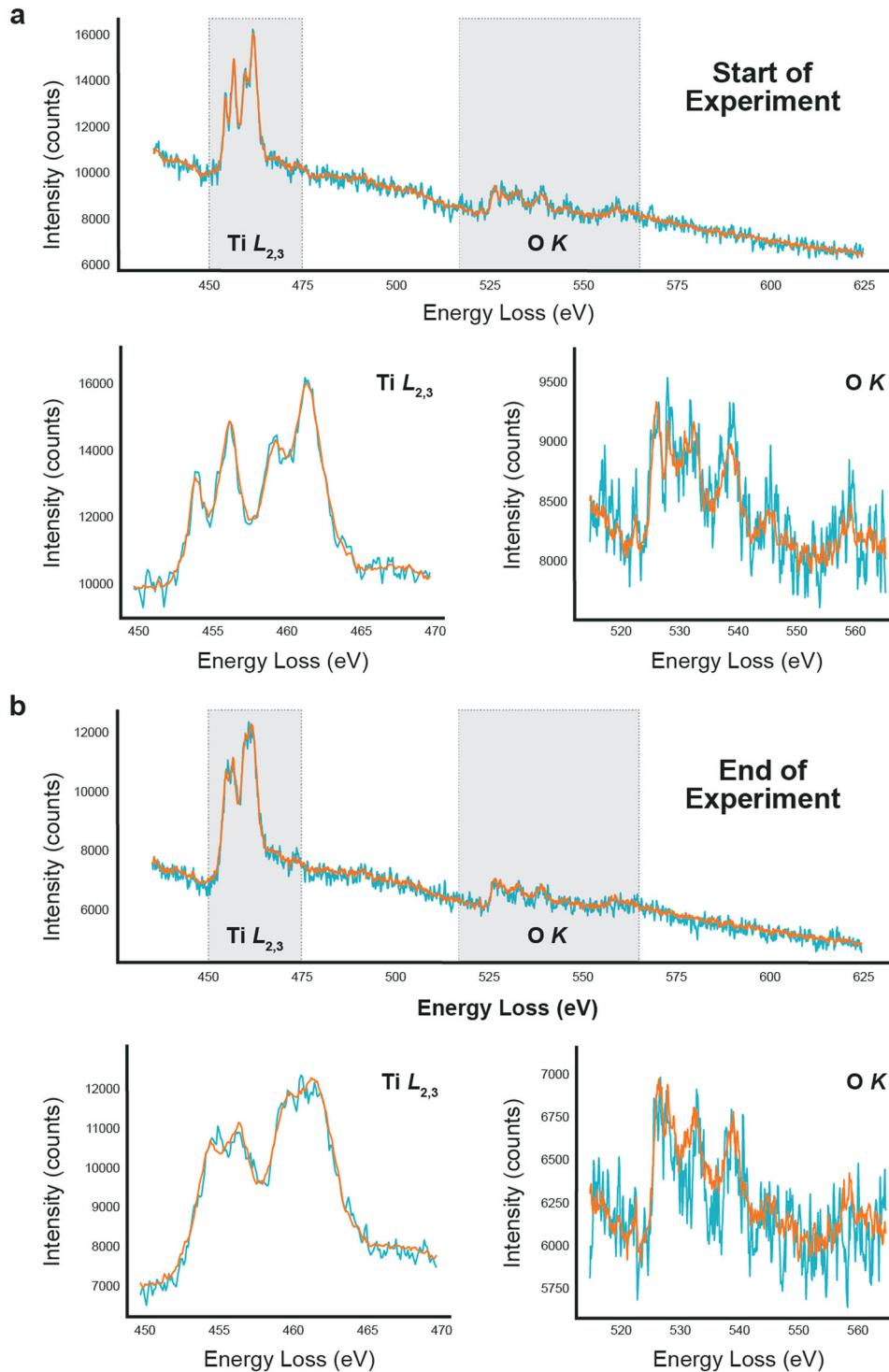
Fig. 4 Optimized LSTM prediction of an STO order-disorder phase transition. a Timestep near the start of the experiment ($t \approx 15$ s), where the sample is crystalline and nearly fully oxidized. **b** Timestep near the end of an experiment ($t \approx 60$ s), where the sample is largely amorphous and heavily reduced. The blue line indicates the real spectrum at an 8 timestep horizon, while the orange indicates the LSTM model output. Shaded regions of each full spectrum indicate the Ti $L_{2,3}$ and O K edges.

665 spectrometer, with a spectrometer acceptance angle range of 113–273 mrad, a dispersion of 0.1 eV ch^{-1}, and a dwell time ranging from 0.08–0.8 s px^{-1}. Spectra were binned 130× 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.

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 1. Graphical user interface (GUI) devel- opment, as described in Supplementary Fig. 3, was implemented in Python 3.8. The Flask 2.0.1 and sqlalchemy 1.4.7 packages were used for the framework of the GUI.

SUPPLEMENTARY AVAILABILITY

Supplementary containing details on hyperparameters, autocorrelation analysis, analysis of a different dose rate, and the GUI is available.

DATA AVAILABILITY

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

CODE AVAILABILITY

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

REFERENCES

1. Cifuentes, J., Marulanda, G., Bello, A. & Reneses, J. Air temperature forecasting using machine learning techniques: a review. *Energies* **13**, 4215 (2020).
2. Vivas, E., Allende-Cid, H. & Salas, R. A systematic review of statistical and machine learning methods for electrical power forecasting with reported mape score. *Entropy* **22**, 1412 (2020).
3. Chatzis, S. P., Siakoulis, V., Petropoulos, A., Stavroulakis, E. & Vlachogiannakis, N. Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert Syst. Appl.* **112**, 353–371 (2018).
4. Tsolaki, K., Vafeiadis, T., Nizamis, A., Ioannidis, D. & Tzovaras, D. Utilizing machine learning on freight transportation and logistics applications: a review. *ICT Express.* https://doi.org/10.1016/j.icte.2022.02.001 (2022).
5. Schwarting, W., Alonso-Mora, J. & Rus, D. Planning and decision-making for autonomous vehicles. *Annu. Rev. Control Robot. Auton. Syst.* **1**, 187–210 (2018).
6. Rosique, F., Navarro, P. J., Fern´andez, C. & Padilla, A. A systematic review of perception system and simulators for autonomous vehicles research. *Sensors* **19**, 648 (2019).
7. Battineni, G., Sagaro, G. G., Chinatalapudi, N. & Amenta, F. Applications of machine learning predictive models in the chronic disease diagnosis. *J. Pers. Med.* **10**, 21 (2020).
8. Lalmuanawma, S., Hussain, J. & Chhakchhuak, L. Applications of machine learning and artificial intelligence for covid-19 (sars-cov-2) pandemic: a review. *Chaos Solitons Fractals* **139**, 110059 (2020).
9. Ghorpade, P. et al. Flood forecasting using machine learning: a review. In: *Proc. 8th International Conference on Smart Computing and Communications (ICSCC)* 32–36 (IEEE, 2021).
10. Tahmasebi, P., Kamrava, S., Bai, T. & Sahimi, M. Machine learning in geo- and environmental sciences: from small to large scale. *Adv. Water Resour.* **142**, 103619 (2020).
11. Hatfield, P. W. et al. The data-driven future of high-energy-density physics. *Nature* **593**, 351–361 (2021).
12. Radovic, A. et al. Machine learning at the energy and intensity frontiers of particle physics. *Nature* **560**, 41–48 (2018).
13. Zhang, C. et al. Recent progress of in situ transmission electron microscopy for energy materials. *Adv. Mater.* **1904094**, 1904094 (2019).
14. Zheng, H., Meng, Y. S. & Zhu, Y. Frontiers of in situ electron microscopy. *MRS Bull.* **40**, 12–18 (2015).
15. Taheri, M. L. et al. Current status and future directions for in situ transmission electron microscopy. *Ultramicroscopy* **170**, 86–95 (2016).
16. Stach, E. et al. Autonomous experimentation systems for materials development: a community perspective. *Matter* **4**, 2702–2726 (2021).
17. Spurgeon, S. R. et al. Towards data-driven next-generation transmission electron microscopy. *Nat. Mater.* **20**, 274–279 (2021).
18. Olszta, M. et al. An automated scanning transmission electron microscope guided by sparse data analytics. *Microsc. Mircroanal.* **28**, 1611–1621 (2022).
19. Liu, Y. et al. Experimental discovery of structure–property relationships in ferroelectric materials via active learning. *Nat. Mach. Intell.* **4**, 341–350 (2022).
20. Kalinin, S. V. et al. Automated and autonomous experiments in electron and scanning probe microscopy. *ACS Nano* **15**, 12604–12627 (2021).
21. Groschner, C. K., Choi, C. & Scott, M. C. Machine learning pipeline for segmentation and defect identification from high-resolution transmission electron microscopy data. *Microsc. Mircroanal.* **27**, 549–556 (2021).
22. Sadre, R., Ophus, C., Butko, A. & Weber, G. H. Deep learning segmentation of complex features in atomic-resolution phase-contrast transmission electron microscopy images. *Microsc. Mircroanal.* **27**, 804–814 (2021).
23. Horwath, J. P., Zakharov, D. N., M´egret, R. & Stach, E. A. Understanding important features of deep learning models for segmentation of high-resolution transmission electron microscopy images. *NPJ Comput. Mater.* **6**, 108 (2020).
24. Zhang, C., Feng, J., DaCosta, L. R. & Voyles, P. Atomic resolution convergent beam electron diffraction analysis using convolutional neural networks. *Ultramicroscopy* **210**, 112921 (2020).
25. Xu, W. & LeBeau, J. A deep convolutional neural network to analyze position averaged convergent beam electron diffraction patterns. *Ultramicroscopy* **188**, 59–69 (2018).
26. Madsen, J. et al. A deep learning approach to identify local structures in atomic-resolution transmission electron microscopy images. *Adv. Theory Simul.* **1**, 1800037 (2018).
27. Kaufmann, K., Lane, H., Liu, X. & Vecchio, K. S. Efficient few-shot machine learning for classification of EBSD patterns. *Sci. Rep.* **11**, 8172 (2021).
28. Doty, C. et al. Design of a graphical user interface for few-shot machine learning classification of electron microscopy data. *Comput. Mater. Sci.* **203**, 111121 (2022).
29. Akers, S. et al. Rapid and flexible segmentation of electron microscopy data using few-shot machine learning. *NPJ Comput. Mater.* **7**, 187 (2021).
30. Yu, L. et al. Unveiling the microscopic origins of phase transformations: An in situ term perspective. *Chem. Mater.* **32**, 639–650 (2020).
31. Ede, J. M. Deep learning in electron microscopy. *Mach. Learn. Sci. Technol.* **2**, 011004 (2021).
32. Lim, B. & Zohren, S. Time-series forecasting with deep learning: a survey. *Philos. Trans. R. Soc. A* **379**, 20200209 (2021).
33. Zeng, A., Chen, M., Zhang, L. & Xu, Q. Are transformers effective for time series forecasting? Preprint at: http://arxiv.org/abs/2205.13504 (2022).
34. Yu, Y., Si, X., Hu, C. & Zhang, J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **31**, 1235–1270 (2019).
35. Robertson, C., Wilmoth, J. L., Retterer, S. & Fuentes-Cabrera, M. Performing video frame prediction of microbial growth with a recurrent neural network. Preprint available at: https://arxiv.org/abs/2205.05810 (2022).
36. Agar, J. C. et al. Revealing ferroelectric switching character using deep recurrent neural networks. *Nat. Commun.* **10**, 4809 (2019).
37. Siami-Namini, S., Tavakoli, N. & Namin, A. S. A comparison of ARIMA and lSTM in forecasting time series. In *Proc. 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 1394–1401 (IEEE, 2018).
38. Fu, W. et al. Deep-learning-based prediction of nanoparticle phase transitions during in situ transmission electron microscopy. Preprint available at: https://arxiv.org/abs/2205.11407 (2022).
39. Ede, J. M. Adaptive partial scanning transmission electron microscopy with reinforcement learning. *Mach. Learn.: Sci. Technol.* **2**, 045011 (2021).
40. Stollenga, M. F., Byeon, W., Liwicki, M. & Schmidhuber, J. Parallel multi-dimensional LSTM, with application to fast biomedical volumetric image segmentation. 28 (Curran Associates, Inc., 2015).
41. Spurgeon, S. & Chambers, S. *Atomic-Scale Characterization of Oxide Interfaces and Superlattices Using Scanning Transmission Electron Microscopy* (Elsevier, 2018).
42. Pate, C. M., Hart, J. L. & Taheri, M. L. Rapideels: machine learning for denoising and classification in rapid acquisition electron energy loss spectroscopy. *Sci. Rep.* **11**, 19515 (2021).
43. Spurgeon, S. R. Order-disorder behavior at thin film oxide interfaces. *Curr. Opin. Solid State Mater. Sci.* **24**, 100870 (2020).
44. Yao, L. & Guan, Y. An improved LSTM structure for natural language processing. In: *Proceedings of the IEEE International Conference of Safety Produce Informatization*, IICSPI 2018 565–569 (2019).
45. Ghosh, S. et al. Contextual LSTM (CLSTM) models for Large scale NLP tasks. Preprint available at: https://arxiv.org/abs/1602.06291v2 (2016).
46. Park, S. H., Kim, B., Kang, C. M., Chung, C. C. & Choi, J. W. Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture. In *Proc. IEEE Intelligent Vehicles Symposium* 2018, 1672–1678 (2018).
47. Yang, Y. et al. A CFCC-LSTM model for sea surface temperature prediction. *IEEE Trans. Geosci. Remote Sens.* **15**, 207–211 (2018).
48. Ghimire, S. et al. Stacked LSTM sequence-to-sequence autoencoder with feature selection for daily solar radiation prediction: a review and new modeling results. *Energies* **15**, 1061 (2022).

49. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997).
50. Gubbens, A. et al. The gif quantum, a next-generation post-column imaging energy filter. *Ultramicroscopy* **110**, 962–970 (2010).
51. de la Peña, F. et al. Hyperspy/hyperspy: Release v1.7.1 (2022).
52. Hart, J. L. et al. Direct detection electron energy-loss spectroscopy: a method to push the limits of resolution and sensitivity. *Sci. Rep.* **7**, 8243 (2017).
53. Egerton, R. F. Electron Energy-Loss Spectroscopy in the Electron Microscope (Springer US, 2012).
54. Egerton, R. & Malac, M. Improved background-fitting algorithms for ionization edges in electron energy-loss spectra. *Ultramicroscopy* **92**, 47–56 (2002).
55. Lewis, N. R., Hedengren, J. D. & Haseltine, E. L. Hybrid dynamic optimization methods for systems biology with efficient sensitivities. *Process* **3**, 701–729 (2015).
56. Mosk, A. et al. Atomic-scale imaging of nanoengineered oxygen vacancy profiles in SrTiO$_3$. *Nature* **430**, 657–661 (2004).
57. Spurgeon, S. R. et al. Asymmetric lattice disorder induced at oxide interfaces. *Adv. Mater. Interfaces* **7**, 1901944 (2020).

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 ML 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.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41524-022-00940-2.

Correspondence and requests for materials should be addressed to Steven R. Spurgeon.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.