Versatile domain mapping of scanning electron nanobeam diffraction datasets utilising variational autoencoders

A. Bridger1,2,3, W. I. F. David1,2, T. J. Wood4, M. Danaie5,6,7, and K. T. Butler4,5,6

Characterisation of structure across the nanometre scale is key to bridging the gap between the local atomic environment and macro-scale and can be achieved by means of scanning electron nanobeam diffraction (SEND). As a technique, SEND allows for a broad range of samples, due to being relatively tolerant of specimen thickness with low electron dosage. This, coupled with the capacity for automation of data collection over wide areas, allows for statistically representative probing of the microstructure. This paper outlines a versatile, data-driven approach for producing domain maps, and a statistical approach for assessing their applicability. The workflow utilises a Variational AutoEncoder to identify the sources of variance in the diffraction signal, and this, in combination with clustering techniques, is used to produce domain maps. This approach is agnostic to domain crystallinity, requires no prior knowledge of crystal structure, and does not require simulation of a library of expected diffraction patterns.

npj Computational Materials (2023) 9:14; https://doi.org/10.1038/s41524-022-00960-y

INTRODUCTION
The drive of material science towards improving the physical properties of a given application is universally aided by improving the fidelity with which materials can be characterised, modelled and understood, across length scales, from nanometres to centimetres. Scanning electron nanobeam diffraction (SEND) can offer a fast and convenient approach for collecting structural information on the nanometre scale. While a wealth of insight into the bulk structure can be obtained through techniques such as powder X-ray diffraction (PXRD) and powder neutron diffraction (PND), and into the immediate local structure using techniques such as high-resolution scanning transmission electron microscopy (HR-STEM), SEND can provide a solution to bridge these length scales and produce a more complete structural model. SEND, as a technique, is becoming increasingly accessible alongside advances in fast electron detectors and the technique lends itself well to microscope automation. This means it can cover areas in the micrometres length scale and hence, while still capturing microstructural detail not accessible via PXRD, provide statistically significant insight into the crystallographic defects and morphology of the phases present in the sample.

Utilising the 4D scanning transmission electron microscopy (4D-STEM) data collection approach, SEND combines this with a minimised probe convergence semi-angle to obtain distinct Bragg peaks within the diffraction patterns. The advantages that SEND technique offers make it a versatile approach for studying a wide variety of materials. The primary advantages of SEND technique are; the relaxation of sample thickness requirements (relative to HR-STEM) resulting in data collection from much wider areas of the sample, which in turn offers better statistics of the microstructure; and the ability to operate the microscope in a stable configuration, without requiring the frequent aberration tuning common in HR-STEM measurements, which permits for longer and more autonomous data collection. Given the relatively large probe size, resulting from the optical setup in SEND, data collection is normally done at a low to medium magnification range. This leads to a reduced electron dose subtended on the sample, making this technique particularly suitable for the study of beam-sensitive systems, e.g. organic perovskites and metal-organic frameworks (MOFs). Due to these factors, SEND data acquisition is well suited to the task of domain mapping, which is typically achieved through crystallographic orientation mapping. Domain mapping, broadly speaking, is the task of identifying regions which share distinguishing properties and grouping them together. Producing these domain maps for SEND data allows for the aggregation of all the diffraction information available within the entire domain into a single, signal-boosted diffraction pattern. Working with these signal-boosted patterns rather than the individual patterns is advantageous for studying more subtle features like superstructure and can aid pattern indexing, as the signal-boosted patterns typically exhibit additional resolvable Bragg peaks, and improved peak definition. If the atomic structure present within the mapped domains can be identified (through direct treatment of diffraction data or combination with complementary techniques), then the statistical nature of SEND technique could provide a link between local structure and the experimental results of the bulk techniques.

For a given dataset, no single, ground-truth domain map exists which is inherent to that sample. Experimentally, there will always be multiple sources of variation between patterns, due to the statistical nature of the sampling. As the domain map seeks to group regions of shared properties, the returned domain map for a sample must depend on what property similarities are relevant to the specific experiment. For a given sample, it is therefore plausible to seek different domain maps, depending on what specific factors are of interest. As an example, given a SEND dataset taken from a thin foil sample with small precipitates dispersed within it, one might only be interested in the locations on the foil where these precipitates have aggregated and the structural and orientation factors present in the matrix that drives precipitation in these locations. Alternatively, for the exact same dataset, one might instead be interested in the precipitate
structure, and their orientation relationship with the matrix, in which case, a domain map which identifies differences in bulk/matrix lattice mean crystal structure and relative orientation would be desired. Thus, the goal of the experiment will always enforce bias onto which factors of variance are relevant and this should be considered when separating domains.

Research into automated crystallographic orientation mapping (ACOM)\textsuperscript{15,16} has employed SEND data collection to tackle the task of domain mapping. These techniques rely on prior knowledge of possible underlying crystal structures in order to classify the nature of individual diffraction patterns and then aggregate these into maps. For ACOM, a library of possible zone-axis orientations are generated and the individual patterns are correlated to determine the most plausible match. This approach works well when there is prior knowledge of models which can replicate all the desired variations within the experimental diffraction data collected. Unfortunately, this is frequently not the case. In many situations, the simulations obtained from the models can differ from the experiment due to factors which are absent, overlooked or incorrect. Example factors include nuanced superstructure\textsuperscript{17,18}; overlapping lattice motifs\textsuperscript{19}; non-systematic dynamical effects; or simply a lack of structural insight. Non-systematic dynamical effects usually cannot be fitted and so attempts must be made to mitigate them experimentally\textsuperscript{20}. These simulation approaches are also unable to distinguish amorphous regions within the sample from vacuum or thick samples.

This work reports a structurally agnostic, data-driven workflow for domain mapping, which requires no prior knowledge of possible crystal structures and is equally as effective for identifying amorphous regions with little to no crystalline diffraction character. Techniques based on feature extraction with non-negative matrix factorisation (NMF)\textsuperscript{21} and hierarchical k-means clustering to group similar diffraction signal\textsuperscript{22} have recently been reported. The workflow, discussed within, leverages a variational autoencoder (VAE) to identify the variance and similarity within the diffraction data. VAEs have recently shown great promise for processing and understanding large scientific datasets, extracting useful representations of data\textsuperscript{23–25}. For this workflow, the VAE is used in tandem with NMF to produce domain maps. The VAE simultaneously improves the signal-to-noise ratio and uses similarities between 2D diffraction patterns to optimally cluster regions in the low dimensional manifold of the VAE latent-space. This approach means that the factors determining the segregation are tangible and interpretable, allowing for simple alignment of the bias in the output domain map with the experimental goals. Additionally, this approach produces a confidence map of the domain assignments, which means that the experimentalist can manually investigate regions of ambiguity and, if desired, update the mapping process.

This approach combines the convenience and rigour of automated mapping, with the flexibility of human-in-the-loop steering of the desired outcomes. This report begins by outlining the workflow as shown in Fig. 1. The paper then demonstrates the procedure, first on a synthetic dataset, which allows for quantifiable comparison with ground truth, then on a diverse set of real experimental datasets. It is shown that this method can produce meaningful domain maps in different scenarios where different applications are required. The code needed to run this method is available in an open repository\textsuperscript{26} with the intention that the flexibility of this approach will make it a valuable tool for the application of SEND to gain statistically relevant insights into composition-structure-property relationships in materials.

Fig. 1 Overview of proposed workflow. The motivation of each step can be most straightforwardly understood by working backwards from the final domain map.
RESULTS AND DISCUSSION

Workflow—overview

The input SEND dataset is in the form of a 4D data array (x, y | u, v): two dimensions corresponding to the x, y coordinates of the probe position in the real space plane; and two dimensions corresponding to u, v coordinates of the pixelated detector recording diffraction data intensities in reciprocal space. This dataset is then pre-processed and used to train a VAE. The VAE is constructed in two parts, encoder and decoder, which are trained in tandem to produce the identity transformation (i.e. take the diffusion signal as input and return the same signal as output), passing through an n-dimensional latent-space (where n < < number of detector pixels). Thus the VAE learns to produce a compressed representation of the data. Once trained, the two halves can be used independently, to convert the diffusion signal into low dimensional coordinates (encoder) or low dimensional coordinates into diffusion signal (decoder).

As standard, this workflow uses two dimensions for its low-dimensional coordinates. The properties of both the encoder and decoder are leveraged by this workflow, but in the first case, the encoder was used to convert the pre-processed patterns of the input data into a collection of points within a 2D plane (green section to yellow section in Fig. 1). As a result of the constraints of the tandem training of the components of the VAE, the 2D representation outputted from the encoder will group patterns of similar diffraction character proximally within this latent-space. Directly clustering these points based on 2D coordinates alone is non-trivial. While similar patterns will be encoded near each other, a point being nearby is not guaranteed to be from a similar pattern. This can lead to ambiguity in identifying the boundaries between regions from point locations alone.

In order to most effectively and efficiently cluster patterns into separate domains, the workflow uses a four-step process, which is expanded upon later: (i) estimate where the boundaries of domain regions could be in the latent-space; (ii) estimate where a representative pattern of each domain is located; (iii) use the structural similarity index (SSI) of representative patterns and decoded boundary patterns to define the boundary of each domain’s region in the latent-space; and (iv) use the location of encoded data relative to these boundaries to cluster our dataset. Using this workflow, our sample data can quickly be assigned to clusters and these labels can be used to produce domain maps of the sample. We now explore each of these steps in more detail.

Workflow—variational autoencoder

The AutoEncoder was first developed in the late 1980s and was later extended to the VAE by replacing each single latent-space coordinate with a pair of values which define the statistical distribution from which that coordinate could be sampled. The function of the VAE within our workflow is to act as a flexible approach for identifying the most important sources of variation within the diffraction dataset and embedding these patterns conveniently into a low-dimensional representation.

The VAE was built using Tensorflow and the model architecture is summarised in Fig. 2. The trained models are also available in a repository associated with this paper.

Upon training, the VAE has no inclination as to the physical phenomena underlying the observed data nor an appreciation of what aspects of the dataset one might be interested in. As the VAE trains it seeks to minimise the loss function, which is calculated as the sum of mean squared error (difference between input and reconstructed patterns) and Kullback–Leibler divergence (deviation of the distributions in the latent-space from the normal distribution). The loss is minimised by modifying the parameters which control the encoding into the latent-space and the decoding of these values to a reconstructed pattern. Performance is biased towards optimising the separation of the largest sources of variance within the training dataset first and then refining the model for smaller and smaller sources of variance. Both of these biases define how the model will separate out the input dataset and essentially define the model’s understanding of similarity.

In order to use the VAE most effectively, it is, therefore, necessary to manipulate these biases to align the VAE’s interpretation of similarity with the definition of similarity imposed by the focus of the particular experiment. This is most effectively achieved by modifying the training set to contain the representative proportions of the relevant sources of variance and truncating training once all the desired variance has been identified. This truncation is achieved by saving iterations along the training process and evaluating the performance at different stages. This evaluation can be done by visually comparing the patterns inputted and outputted by the model and also by comparing decoded points from the different regions in the latent-space to input the patterns that have been encoded into those regions. If the bias is controlled effectively, the VAE can be used in vastly disparate use cases to give a primary identification of which patterns fall within which domain.

Workflow—latent clustering

In order to cluster the patterns into single domains, the 2D encodings are used in tandem with the decoder and the structural similarity index (SSI). The aim is to find and define the regions within the 2D latent-space, that correspond to each domain. To achieve this, the workflow seeks to identify the locations of all boundaries between these domain regions, and also identify a typical pattern for each domain (yellow and orange sections of Fig. 1, respectively). Then, by passing points along the boundaries through the decoder to generate diffraction patterns, the workflow uses the SSI image similarity metric between these decoded patterns and the domain typical patterns, to identify which boundary points correspond to each domain.

Workflow—boundary point estimation

The first requirement for the clustering approach is to identify candidate points that could define the boundaries of each domain. It is assumed, due to the bias of the VAE, that the encoded data will be grouped such that similar patterns are nearby and so each domain will occupy a continuous, definable region, within the latent-space. The challenge is that the same features of the VAE, which allow for each domain to occupy a continuous, smoothly varying region within the latent-space, also lead to the boundaries between these regions becoming more smoothly varying. This makes it difficult to identify the locations of the boundaries between domains, simply by evaluating changes in the decoded patterns as one moves through latent-space. This is because there tend not to be discrete boundaries between the patterns of different domains. While these smoother inter-domain transition...
The aim is to use the sampled candidate boundary points to define a bounded area for each domain region. To achieve this, the approach used is to identify a set of representative patterns (referred to as centroids), one for each domain, that can be compared to the decoded boundary patterns and used to assess which region the boundary point most closely resembles. The comparison metric that is used in this work is the SSI, and while other metrics are not implemented in the code, it would be straightforward to add and use any such metrics. The requirement of the comparison metric is that the decoded pattern of a general point sampled from within a given domain’s boundary will give a stronger response when compared to that domain’s centroid pattern than any of the other centroids. The SSI was chosen as this performance was observed upon inspection by a domain expert. Due to the continuous nature of the latent-space, almost any point within the region should fulfil the role of a representative pattern. Thus the centroids can be estimated by a few approaches: using the derivatives of KDE to find stationary points in the density function which correspond to centres of boundary regions; being manually selected; or by using non-negative matrix factorisation (NMF). NMF is the most automated approach and is used as a standard in this workflow. An overview of the mathematical formalisation of NMF is given in Section ‘Methods’.

In order to find a set of patterns to be representative of the domains within the latent-space by NMF, the workflow aims to use a dataset of diffraction patterns that contains all the pattern variation which has been encoded within the latent-space. Then the NMF is applied to extract the different sources of variations in the patterns that are found within this dataset. Assuming that the patterns within each domain exhibit some characteristic of their pattern that distinguishes them from the other domains, then these characteristics are sources of variation within the dataset. As such, these characteristics will be identified by the NMF. As each characteristic will be shared by the patterns within this domain but not in the other domains it should fulfil the role of the centroid, as defined earlier.

The dataset used could just be the original data, but this will have performance issues due to the uneven sampling of domains. The original data will be larger than is required, due to oversampling points in the more populated domains, and so will take longer to perform the NMF. Further, as the less populated domains will be significantly undersampled, the source of pattern variation corresponding to these might prove harder to extract from the inherent noise within the more populated domains.

To generate a better-performing dataset for centroid estimation, the workflow evenly samples points across the populated regions of the latent-space, using the already calculated density approximation to identify the populated regions. These points can be fed through the decoder to generate a collection of diffraction images representative of the pattern variance within the original dataset. This rebalanced dataset reduces the issues identified for the raw dataset. Firstly, it is a much smaller dataset while still containing all the variance the VAE has identified within training; this allows for faster calculation of the NMF. It is also a significantly de-noised dataset, as it only contains the variances that have been identified as relevant during the VAE training. Removing the sources of variation not corresponding to changes in domain, will improve the applicability of these NMF components for use as centroids. Finally, the distribution of patterns within the dataset is...
less dependent on the occurrence frequency within the sample and will be proportional to the size of the region in the latent-space. This increases the visibility of minor phases and, in turn, results in better segmentations of less populated domains.

An NMF is performed on the rebalanced dataset. The workflow then compares these components to the patterns from the rebalanced dataset and replaces the NMF component with the reconstructed pattern that most closely matches it, as measured by SSI. This is because the NMF component will contain only the distinguishing characteristic of that pattern, but ideally, the centroid would be the entire diffraction pattern that is representative of that domain, not just part of it. This set of patterns will then correspond to the centroids of our domains. The total number of these centroids is, therefore, dependent on the number of components requested by the NMF. The fewer the components requested of the NMF, the larger the sources of variance in the dataset has to be in order to be classified as a component. To reduce the dependence of the clustering on this parameter, the workflow allows for comparisons and evaluation of domain maps using a different level number of centroids. Additionally, once the centroid patterns have been identified by NMF, they can be compared by SSI and patterns centroids which are sufficiently similar (by a user-defined threshold) can be merged (by averaging the point locations in the latent-space). By doing this, the number of domains is slightly more flexible and is driven by a maximum level of similarity between domains, rather than being directly determined by a chosen parameter.

Once the centroids and potential boundary points have been identified, the SSI can be used to assign each boundary point to the best matching centroid. Then, for each centroid, the classified boundary points can be used to define an outer limit of that domain region using a convex hull. The convex hull is used as it allows for faster evaluation of points falling within the hull. A convex hull is applicable as, despite not producing a perfect bounding hull, the inaccuracy should only produce false positive inclusions, which should then lead to multiple group inclusion conflicts which can be straightforwardly resolved. With the latent-space now defined by these hulls, encoded patterns can fall into one of three categories: not within any hull; within exactly one hull; or within more than one hull. The vast majority of the points tend to fall within exactly one hull. The points shared between hulls are classified by direct SSI of the pattern with the conflicting hull centroids and the points not within any hulls are classified by direct comparison with all hull centroids.

Once the points have all been classified into domains, an additional means of centroid approximation becomes trivial, taking the centre of mass (CoM) of the encoded points within each domain. This can offer a useful approach for refining a centroid location and might improve the outcome of a successive iteration of the workflow. This can be beneficial if the region is approximately centrosymmetric and the CoM approximation moves the centroid more centrally within the region. As standard, the workflow outputs the CoM of the identified region as well as the centroid used.

Case studies

In this section, three case studies are presented to show the diverse use cases for the workflow. These case studies are a dataset populated with simulated diffraction patterns for known, distinct orientations of a single crystal structure; an experimental SEND dataset from an Al alloy thin foil with precipitates; and an experimental dataset of a particle containing a mixture of phases and orientations of layered metal oxides (LMO).

Case study—simulated dataset

The simulated dataset is populated using 12 distinct zone axis orientations of an LMO structure (as detailed in Simulated dataset—data collection) and is shown in Fig. 4c. The regions 1–5 and 7–11 are each populated with the simulated pattern of a single zone axis orientation. The bar at the bottom containing labels 12, 13 and 14 is populated with the simulated pattern of a single zone axis, but the diffraction signal decreases in intensity from right to left, this is designed to emulate real-world conditions where signal intensity might be reduced due to factors such as thickness changes or beam damage. Region 7 contains patterns for an amorphous region. The final region, 0, is populated with an unscattered electron probe to emulate the vacuum around particles of interest. The goal of the domain mapping for this dataset was to identify the areas of different zone axis orientations.

The dataset was generated with a 296 × 295 real space and 128 × 128 pixel array in each diffraction pattern. This pattern size is ideal for training the VAE and no pre-processing was required. The model was trained, from scratch, on all the shuffled input patterns with a random sample of 1000 patterns used to monitor performance. The training was truncated at the point where the model began identifying distinctions that were deemed no longer relevant to the aim of the segmentation. Once the best iteration of the model is selected, the dataset is processed by the workflow and the results are shown in Fig. 4e, f.

Comparing the output to the ground truth, the consistency of the clustering is above 98% for all regions, excluding the intensity-varying bar, which identified 86% of the total patterns. The drop in performance for the bar is a result of patterns with very low diffraction signals that closely resemble the vacuum. This resemblance grows increasingly along the bar such that at the far left, the diffraction signal is far more akin to a vacuum than the actual structure present, so this could be considered a valid interpretation of the domains.

Case study—precipitates in Al alloy thin foil

Precipitation-hardening is an important mechanism for increasing the strength of commercial aluminium alloys. Gaining statistical insight into the population and crystallographic nature of the precipitates in these alloys is experimentally challenging, and thus SEND is a suitable technique to tackle this problem. The experimental dataset presented here is an example of such SEND data. This dataset is comprised of 511 × 511 probe positions in real space with 256 × 256 pixel array in diffraction patterns. The goal of the segmentation is to identify the diffraction character of the precipitates, which form a minor component of a dataset dominated by the diffraction of the Al matrix.

To optimise the training for this functionality, the data is pre-processed to crop to the central 128 × 128 of the diffraction pattern, where the majority of the desired information can be observed. The training data is then re-sampled using the Shannon entropy of the processed patterns as an indicator of precipitate character. The Shannon entropy is a measure of information content in an image and is given by $-\sum p_i \log(p_i)$, where $p_i$ is the frequency/probability of pixels of value k. Shannon entropy is used as the precipitate patterns are expected to have a higher entropy due to the additional diffraction information and is based on a similar application of Shannon entropy by Slouf et al. The VAE was trained and the resulting latent-space clustered into two regions, which correspond to the precipitates and the background Al matrix.

Other methods for masking the precipitate region will yield similar results more straightforwardly, but this workflow was used to illustrate how VAE bias can be directed using iteration and how this might be applicable in less trivial cases where classical masking techniques are not applicable. The mask was used to generate a new 4D dataset with the variance of the background Al substrate replaced with a single mean pattern. This new dataset was then used to retrain the VAE and the model was used to...
cluster the navigation region using 16 components in the NMF as indicated in Fig. 5.

From the clustering in Fig. 5d, it can be observed that the boundaries of the classes do not precisely follow the gaps between the more dense point encodings (the yellow-green class can be seen to span across various point densities). This is a similar case to that of the thickness effects for the simulated data, as there is a deviation from where one might anticipate boundaries, but again, this is the result of regions with insignificantly low diffraction intensity merging. Inspection of the resulting patterns

---

**Fig. 4  Example domain mapping of a simulated dataset.** a The maximum diffraction pattern of the entire dataset with a virtual annular detector. The scale bar represents a length of 0.3 Å⁻¹. b The virtual annular dark field (VADF) image for the simulated layered metal oxide (LMO) sample dataset. c The region map used to populate the simulated dataset, each region contains patterns of the same crystal structure (or amorphous structure) and orientation but with added noise (and variation in thickness for the bar at the bottom). Pixel locations have been indicated with numbers to aid the understanding of the patterns which populate the dataset. d The diffraction pattern corresponding to the pixel locations highlighted in (c). e The encoded positions of the patterns clustered by colour. The open black circles show the initial centroids identified by NMF and refined by SSI. The black crosses show the new centroids by averaging the locations of points within clusters. f The clustering on the real space pixel locations, a pixel location has been highlighted in each of the identified domains.
in Fig. 5e, f, corroborates this observation. Separating out the low diffraction information regions from the strongly reflecting centres is an entirely valid domain distinction in this use case.

From Fig. 5, the utility of this workflow can be seen. The workflow has taken a sample of 261,121 diffraction patterns and condensed all the information relevant to the experiment into a domain map and 14 patterns. Additionally, in almost all the examples, the signal-boosted pattern for each domain contains better resolved and more usable diffraction information than in the individual patterns. In order to highlight the variation between

Fig. 5  Example domain mapping of a dataset of precipitates on Al alloy. a The maximum diffraction pattern of the entire dataset with a virtual annular detector. The scale bar represents a length of 0.6 Å⁻¹. b The virtual annular dark field (VADF) image for the precipitates in Al alloy thin foil dataset. The scale bar represents a length of 50 nm. c The clustering on the real space pixel locations, a pixel location has been highlighted in each of the identified domains. d The encoded positions of the patterns clustered by colour. The open black circles show the initial centroids identified by NMF and refined by SSI. The black crosses show the new centroids by averaging the locations of points within clusters. e The single diffraction pattern corresponds to the pixel highlighted in (c) and the pattern border colour corresponds to the colour of the class in the segmentation. f The mean diffraction pattern corresponds to the class of the pixel highlighted in (c) and the pattern border colour corresponds to the colour of the class in the segmentation.
the diffraction character for the different clusters, a comparison matrix showing the different patterns for each pair of clusters is available in the Supporting Material.

Case study—phase distribution in the layered metal oxides powder sample

The final case study is a layered metal oxide sample used as cathode material in sodium ion batteries. For these materials, the structural composition can vary across the sample due to local changes in stoichiometry. All the structure types are based on alternating layers of edge-sharing transition metal-oxygen (TM-O₆) octahedra and layers of intercalated sodium ions. The dataset is comprised of 256 x 256 probe positions in real space with a 512 x 512 pixel array in the diffraction plane. The goal of the segmentation is to identify different phases present within the sample, as well as variations in orientation or composition within these phases. For pre-processing, the diffraction information is centred and cropped to 256 x 256 and then downsampled by a factor of 2 to get 128 x 128 input patterns. As before, the irrelevant vacuum and carbon support are masked out to remove unnecessary variance from the VAE training. This is achieved by a more straightforward approach than in the previous example: by masking the central beam and selecting patterns with low sum total intensities. The remaining patterns are again sampled using the Shannon Entropy. The new training data is used to train the VAE and the workflow is implemented to produce the domain map in Fig. 6.

It can be seen from the latent space and the subsequent domain map that this is a more structurally rich sample than the previously studied examples but the workflow provides a robust segmentation, identifying minor phases in classes of fewer than 70 pixels alongside major classes with as many as 10,000 pixels. The segmentation breaks the sample down into 33 largely distinct regions. While some of these regions correspond to multiple classes containing similar diffraction characters, this is a conscious decision for the dataset to search for minor factors that might arise due to changes in composition, such as super-lattice reflections, and if desired we could tune this by altering the SSI threshold.

It is key to the use of the workflow that there is a human-in-the-loop, to ensure that obtained domains are meaningful statistically (as discussed below in Section 'Evaluating confidence in domain identification') and then to extract the physical basis for each of the identified domains. Further work is ongoing to help aid the human-in-the-loop in interpreting the patterns, incorporating the orientation-matching algorithms used in ACOM¹⁵ to identify changes in crystal structure and orientation. A detailed discussion of its application to this dataset is, however, beyond the scope of this study. As with the precipitate dataset, a comparison matrix showing the different patterns for each pair of clusters is also available in the Supporting Material and this form of evaluation can act as a primary indication of whether the domains contain relevant variation in diffraction character.

A cursory review of the domains observed in Fig. 6 shows that the domains result from large differences in crystal orientation (an example seen in the different Bragg peaks present in patterns 3 and 7), minor misorientations (seen by changes in peak intensity in patterns 0 and 11) and changes in superstructure presence (seen by changes in observation of extra peaks in 6 and 8).

Comparing patterns in Figs. 5, 6 the difference in challenge between the Al alloy with precipitates sample and the layered metal oxide sample can be seen. The individual patterns of the LMO dataset show better signal-to-noise than the precipitate dataset. As a result, there is less of a stark contrast in the domain patterns compared with the individuals, with regard to signal boosting. However, the quality of the domain mapping can still be qualified by the domain patterns presenting the same sets of reflections that are present in the individual. By this assessment, the clustering is successful in creating clusters where the deviation between the mean patterns and the individual patterns are small. There is also evidence of the level of variation within a phase being extracted, with patterns of the same phase with small intensity shifts (such as the row of reflections in 4 and 5) being separated out.

Evaluating confidence in domain identification

As discussed in previous sections the exact domain map outputted is primarily dependent on the VAE model and the cluster centroids. Altering the training dataset and the length of training will alter the prominence of the different sources of pattern variation in the latent-space and this broadly defines the variation within the diffraction data that the VAE is able to differentiate. How finely/coarsely these variations are separated into different domains will then be determined by the number of cluster centroids and their locations within the latent-space. As such, altering any of the above factors can lead to a change in the final domain map. In order to increase the confidence that the best domain map is being used to evaluate the data (one with the most experimentally applicable separation of domains), a statistical approach was applied.

The motivation of a statistical approach is that while the ideal evaluation would be to go through each individual pattern, within a domain, and check that it has been classified correctly, this is untenable for most domains due to the sheer number of patterns. Instead, using a statistical approach, the pixels that are most likely to change class can be identified and evaluation can be focussed on the classification of these patterns. Pixels likely to change class are determined by comparing maps generated with different parameterisations, and finding which pixels are consistently grouped together.

The consistency is formalised by comparing all pairs of maps (where one is assumed to be the ground truth map and the other a comparison) and evaluating each pixel and its label: what proportion of the total pixels sharing this label in the ground truth map also share their labels in the comparison map. These proportions for each pair of ground truth and comparison map are then averaged to give the overall consistency map. For a given pair of ground truth and comparison map, the proportion of the total pixels sharing one arbitrary label in the ground truth map and another arbitrary label in the comparison map can be obtained from the proportional label overlap matrices (PLOM). The procedure for calculating a statistical consistency map is summarised below and these steps are illustrated in Fig. 7:

1. Calculate all proportional label overlap matrices (PLOM) for every ground-truth/ comparison map pair
2. For each pixel in real space:
   (a) Find the pixel label in the ground-truth map and the label in the comparison map
   (b) For each ground-truth/ comparison map pair, lookup the proportional overlaps for these two labels from the relevant PLOM
3. Average the proportional overlaps of the pixels from one ground-truth/ comparison map pair over all ground-truth/ comparison map pairs

Calculating the proportional label overlap matrix for ground-truth map A and comparison map B is visualised in Fig. 8 and is done using Equations (1) and (2): where $M_{ij}^{A}$ is the element $ij$ of the PLOM for GTDM Map A and comparison Map B; $A_{i}$ is the binary mask of label $i$ in Map $A$; and $A_{j}$ is the class label of the pixel at $A_{j}$. The motivation of a statistical approach is that while the ideal evaluation would be to go through each individual pattern, within a domain, and check that it has been classified correctly, this is untenable for most domains due to the sheer number of patterns. Instead, using a statistical approach, the pixels that are most likely to change class can be identified and evaluation can be focussed on the classification of these patterns. Pixels likely to change class are determined by comparing maps generated with different parameterisations, and finding which pixels are consistently grouped together.

The consistency is formalised by comparing all pairs of maps (where one is assumed to be the ground truth map and the other a comparison) and evaluating each pixel and its label: what proportion of the total pixels sharing this label in the ground truth map also share their labels in the comparison map. These proportions for each pair of ground truth and comparison map are then averaged to give the overall consistency map. For a given pair of ground truth and comparison map, the proportion of the total pixels sharing one arbitrary label in the ground truth map and another arbitrary label in the comparison map can be obtained from the proportional label overlap matrices (PLOM). The procedure for calculating a statistical consistency map is summarised below and these steps are illustrated in Fig. 7:

1. Calculate all proportional label overlap matrices (PLOM) for every ground-truth/ comparison map pair
2. For each pixel in real space:
   (a) Find the pixel label in the ground-truth map and the label in the comparison map
   (b) For each ground-truth/ comparison map pair, lookup the proportional overlaps for these two labels from the relevant PLOM
3. Average the proportional overlaps of the pixels from one ground-truth/ comparison map pair over all ground-truth/ comparison map pairs

Calculating the proportional label overlap matrix for ground-truth map A and comparison map B is visualised in Fig. 8 and is done using Equations (1) and (2): where $M_{ij}^{A}$ is the element $ij$ of the PLOM for GTDM Map A and comparison Map B; $A_{i}$ is the binary mask of label $i$ in Map $A$; and $A_{j}$ is the class label of the pixel at $A_{j}$.
Fig. 6  Example domain mapping of a layered metal oxide dataset.  

(a) The maximum diffraction pattern of the entire dataset with a virtual annular detector. The scale bar represents a length of $1 \, \text{Å}^{-1}$.  

(b) The virtual annular dark field (VADF) image for the layered metal oxide dataset. The scale bar represents a length of 298 nm.  

c The clustering on the real space pixel locations, a pixel location has been highlighted in each of the identified domains.  

d The encoded positions of the patterns clustered by colour. The open black circles show the initial centroids identified by NMF and refined by SSI. The black crosses show the new centroids by averaging the locations of points within clusters.  

(e) The single diffraction pattern corresponds to the pixel highlighted in (c) and the pattern border colour corresponds to the colour of the class in the segmentation.  

(f) The mean diffraction pattern corresponds to the class of the pixel highlighted in (c) and the pattern border colour corresponds to the colour of the class in the segmentation.
Fig. 7  An example of the procedure for calculating the statistical consistency map using 3 maps: Map A; Map B; and Map C (denoted by (i)). All the permutations of the ground truth map (the top row of each triplet of Map X, Map Y, label overlaps) and comparison map (the middle row of each triplet of Map X, Map Y, label overlaps) are given by (ii) along with the respective PLOM. For each pixel, the proportional label overlap is evaluated between the label of that pixel in the ground truth map and the label of that pixel in the comparison map (this can be done by simply referencing the appropriate element of the PLOM). The label overlaps for each of the ground truth map-comparison map pairs are then averaged in (iii) to give the final statistical consistency map.

Fig. 8  An example of the procedure for generating the proportional label overlap matrix for two 4x1 pixel maps, Map A and Map B (denoted by the red (i), to avoid confusion). Each map contains two arbitrarily assigned labels, a and b. The matrix marked (ii) has each element populated by the specific calculation required to evaluate that element. The values resultant from these calculations are given in (iii). The intermediate steps for evaluating (iii) are given in (iv), with each element showing: the binary masks for the respective labels of A and B; the total number of the specified A label; and the summation that yields the dot product of the binary masks.
coordinates \((x, y)\) in Map \(A\).

\[
M_{ij}^{A} = \frac{A_{i}^T B_{j}}{\sum_{i} A_{i}}
\]

(1)

\[
A_{i} = \begin{cases} 
1, & \text{where } A_{xy} = i, \\
0, & \text{where } A_{xy} \neq i.
\end{cases}
\]

(2)

This consistency map provides a visual summary of how invariant the different regions of the sample are to a change in the parameters of the workflow, an example of the precipitates dataset discussed above is shown in Fig. 9.

Pixels with values close to 1.0 consistently appear in clusters with the same sets of pixels, whereas pixels with low values tend to change which sets of pixels they are grouped with. Lower values in the consistency map correspond to regions which are sensitive to hyper-parameter choice. This allows the consistency map to be used to help assess if a selected domain map provides an appropriate segmentation for the desired experiment. For a selected domain map, each individual domain can be applied as a mask to the consistency map. Assessment of the values of the consistency map within one domain can be used to help identify which regions (if any) within this domain are likely to change with a change of clustering parameters. The average patterns associated with each of these identified regions can then be calculated and a visual assessment can be made if the differences between these patterns should, in fact, constitute separation into discrete domains. This can be repeated for all the domains in a given map in order to increase the confidence in this map as a suitable representation of the data and that the best choice has been made for the number of centroids used in the segmentation.

Figure 9 shows an example of applying this protocol to two classes within a domain map. The top left image shows a domain map which was produced using ten centroids identified using NMF. The top right shows the statistical consistency map generated by comparing the map using ten centroids, to a series of maps using 12, 14, 16, 18, 20, 22 and 24 centroids, respectively. The bottom left and bottom right images, show the regions of the Statistical Consistency Map, which correspond to domains 2 and 4 in the 10-centroid map. These are the domain-masked consistency maps. Within each of these two domain-masked consistency maps, it can be seen that there are different consistency regions: regions with higher consistency values (darker green); and regions with lower consistency values (lighter green). Overlaid onto these domain-masked consistency maps are the mean diffraction patterns averaged over each of the consistency regions. For domain 2, five consistency regions are identified, and while the bottom-left-most of these patterns is more noisy than the others, all patterns show good agreement in reflection location and this domain could be considered satisfactory. Conversely, while domain 4 (bottom right) also shows five regions, the mean patterns associated with these regions are less consistent in peak position and intensity, suggesting that the map in Fig. 9a, is not performing a satisfactory segmentation, by classifying all these pixels within the same domain. If, upon inspection, all the classes within a map produce pattern for the different consistency regions akin to domain 2 (bottom left), then this should improve the confidence in the domain map as suitable. If, upon inspection, some of the classes within a map produce pattern for the different consistency regions more like domain 4 (bottom right), then this should reduce the confidence in the domain map and might suggest investigating the use of different parameters for generating the mapping.
Comparison with other techniques

In this section, the outcome of the workflow presented in this paper is compared to some other more established approaches for processing 4D-STEM datasets. The techniques in question are a direct NMF of the sample data (without the use of a VAE or the hull clustering technique) and a feature extraction-based approach utilising Py4DSTEM for much of the data processing. For the sake of simplicity, the techniques are compared using the same single dataset, the precipitate on thin Al foil, but the technique is applied to both the raw dataset and also the precipitate-masked dataset to see how this pre-processing step affects performance.

Direct NMF—method

NMF is used on the dataset to extract 16 principal components (the same number used in the workflow example above) and the patterns are compared against these components using a root mean squared (RMS) comparison and an SSI comparison.

Direct NMF—results

In Fig. 10, the advantage of using the VAE to learn the sources of variance can be seen. As the VAE can be directed to learn to identify the precipitate peaks first, and can be truncated before learning the more subtle variations in the substrate matrix (as described in Results and discussion), the rebalanced dataset generated will only show variance corresponding to the reflections of the precipitate. Thus, when performing NMF, the set of components produced will also be dominated by these precipitate peaks (Fig. 10, right). This is not the case when performing NMF directly on the raw dataset, even if only using patterns from within the precipitate regions to perform the NMF, the resultant components are dominated by variation in the intensity of the substrate reflections (Fig. 10, left).

In order to use the components to then generate clusters, a typical approach would be to compare the sample patterns to the principal components using RMS. Using SSI can provide a superior comparison metric but takes longer, and is still ultimately limited by the weak component detection and, in this case, produces an even poorer domain map than RMS (see Fig. 11). Neither of the maps in Fig. 11 are able to differentiate between the different precipitates in a satisfactory manner. This can be seen explicitly by looking at the patterns in Fig. 5e, f. For example, the single patterns of pixels 3 and 12 are clearly from different orientations and the mean patterns in (f) corroborate this. Both methods in Fig. 11 would have these pixels in the same domain.
The feature extraction work aims to represent the dataset (511 × 511 × 256 × 256) and the default processing pipeline, the iteration time for parameter tuning is prohibitive. An approximate comparison of the compute times, on the same hardware, for the Bragg peak feature extraction workflow and our workflow are given in Table 1. The run times were calculated running the code on a cluster on the Diamond Light Source jupyterhub equipped with 16 CPU cores, 128 GB RAM, Nvidia Tesla V100 GPU and Cuda 10.1.

The iteration time is the time required to complete the workflow if a change is made to the given process. As completing the workflow is often required to evaluate the quality of the parameter change, this metric can be used to indicate how quickly parameter iterations can occur at each stage of the workflow. From the iteration times in Table 1, it can be seen that parameter tuning, even for the last step of the feature extraction workflow, will take in the order of hours, whereas our workflow can be iterated from complete model retraining in under an hour with the clustering behaviour tuned in closer to 5 min per iteration.

The memory usage for the two workflows was also calculated by running the same operations on the Diamond Light Source Hamilton Cluster equipped with Nvidia P100 GPU and 128 GB RAM. The Bragg peak feature extraction used a maximum of 36 GB of virtual memory (vmem), whereas the workflow in this paper used a maximum of 116 GB of vmem.

**Workflow automation, limitations and resource requirements**

The discussions throughout the previous sections have intended to highlight the flexibility of the workflow in different use cases, but also emphasise that a versatile approach, requires directing towards producing domains optimised for the investigations of each experiment. This then promotes questions of how much intervention and resources are required for analysing each new dataset. The drive towards more autonomous workflows aims to promote higher throughput analysis and improve the statistical significance of the observations that can be draw out of these domain maps.

The first area for automation is at the VAE training. While curating the training data is required to optimally bias the encoder towards learning relevant variation, this processing pipeline should not require alteration from sample to sample within the same experiment. This means that once a pre-processing...
routine has been identified which satisfactorily biases the encoder, this routine can then be applied automatically to other datasets with the same desired domain separation. For example, for the layered metal oxide dataset, the pre-processing requires the removal of the carbon support and vacuum patterns, and then balancing the dataset to increase the occurrence of high entropy patterns. With this identified, to produce the domain map of other LMO samples, the same pre-processing workflow could be applied automatically to each dataset to produce the training set.

With the training of the VAE requiring the bulk of the total processing time, automating this process provides a significant improvement in the rate at which domain maps can be produced (see Iteration time in Table 1). The testing of different parameterisations in the clustering step can also be automated, to produce a set of domain maps which can be used for producing a statistical consistency map. The final step of selecting whether an outputted domain map is suitable, however, currently requires human assessment. The resource requirements of this process are high, due to the training of the VAE requiring GPU acceleration to execute in a workable time, and this is certainly a limitation of the workflow. To mitigate this training time, if there is access to sufficient memory, it is possible to aggregate all the pre-processed training datasets together and train one model on all of the data; this model should have a larger latent dimension to handle the added variance and avoid overly compressing the data. As a result of the architecture of the model, transfer learning can then be used to help improve the training time. As the model is made up of 2D convolution (Conv2D) layers, followed by densely connected (Dense) layers, it is expected that the Conv2D layers learn the low-level concepts and then these are passed to the dense layers, which learn to apply these concepts to the specific sample case. Therefore we can apply transfer learning to take the weights of the outer Conv2D layers of this bulk-trained model and apply them to the standard VAE architecture, which is then trained on the specific training dataset in reduced time.

Alongside limitations imposed by resource requirements, we would also like to highlight other potential limitations of the workflow and areas where future work will be directed. The first limitation to discuss is the quality of the diffraction data: while the workflow will function with any 4D-STEM dataset, there is a limit to what can meaningfully be extracted with noisy/low signal data. Additionally to this, the sampling rate of the scan data across the sample will also play a role in the quality of the segmentation.

When the probe size is large compared to the size of the domains of interest, such that a significant proportion of the recorded patterns show overlapping diffraction information (this might be anticipated in a poly-crystalline sample where grains with similar orientations or twins are present across a very
small area) then the workflow may struggle to satisfactorily identify the unique domains. We do not, however, observe that the presence of probe scans at locations with overlapping grains is an inherent limitation. Provided there are scan locations where just the individual grain are present, the workflow should be able to identify the separate domains. For such situations with overlapping grains we might anticipate a response similar to the thickness variation in the simulated dataset, where the exact edge becomes less well-defined or, depending on the size of the overlap in the sample and the parameterisation of the segmentation, the overlapping patterns might be identified as a separate unique domain. Neither of these scenarios should cause an issue in further structural analysis, as the subsequent zone-axis indexing step being developed utilising ACOM \(^{15}\) is capable of dealing with overlapping patterns. In more difficult cases, where most patterns in the sample show some form of grain overlap/twinning, we would also still expect the workflow to help cluster regions where the nature of this overlap/twinning is similar, if such regions exist. In the most extreme case, however, where there are highly varied systems with features of interest that are small relative to the probe size, this workflow would probably not be very applicable.

Additionally, the workflow is limited by the requirement of some guidance from a domain expert. The workflow relies on having a human-in-the-loop to help direct which similarities within the diffraction data were relevant to the particular experiment, and that a pre-processing pipeline can be setup and applied in order to enforce this bias on the VAE.

To summarise, the limitations of this technique are a result of it being reliant on the information present within the distribution of patterns available and it requires the input of a domain expert to aid in biasing the workflow. If the features of interest cannot be extracted from within the distribution of patterns then the workflow will not provide a satisfactory segmentation. Difficulty extracting these features will most likely be a result of features being unobservable in the dataset due to poor signal-to-noise; features being poorly resolved spatially; a prohibitively large number of features being present; or that the pre-processing workflow has not effectively biased the VAE to extract them.

**Discussion**

We have presented an automated workflow for analysing SEND datasets. The approach that we present, while automated, is flexible enough to allow the adaptation of inductive biases in the model, to account for different experimental targets. We have demonstrated the power of our methodology first on a simulated dataset where ground truth is available for evaluation, and then by extracting useful domain maps from a heterogeneous set of experiments where the data and the desired outcomes are significantly different. Our workflow was able to extract these maps with minimal human intervention (aside from deciding the aim of the experiment) and in a short time. We compared our workflow to other common approaches for SEND data analysis and showed that our approach is more versatile and more efficient, ultimately yielding better, more meaningful domain maps. Finally, we also demonstrated a confidence map associated with our SEND analysis, that would allow the experimenter to identify and investigate regions where domain classifications are ambiguous.

With the growth in the speed of data collection, automation of SEND data analysis is imperative. However, automation cannot come at the cost of expert biases of experimenters being able to drive the analysis. The workflow that we present balances automation against the ability to flexibly alter physical biases to drive the analysis. We believe that this synergistic interaction of humans and automated analysis will be critical for unlocking the power of SEND experiments to link information on the atomic and bulk scales in materials.

**METHODS**

The experimental datasets presented were collected at the Electron Physical Sciences Imaging Centre (ePSIC) at Diamond Light Source.

**Simulated dataset—data Collection**

The simulated dataset used a crystallographic information file (CIF) from the inorganic crystal structure database (ICSD) \(^{43}\) for \(\text{NaCoO}_2\) with space group \(R\overline{3}m\) \(^{14}\). This CIF is used in conjunction with PyPrismatic’s multisilice algorithm \(^{45–47}\) to simulate patterns for the different zone axes using an accelerating voltage of 200 kV, a sample thickness of 12 unit cells (~10 nm) and a convergence semi-angle of 2 mrad. These patterns were then used to populate a real space based on the ground truth map in Fig. 4f. The bar was populated by using a weighted combination of the simulated pattern and the vacuum pattern, weighted by the horizontal coordinate of the pixel (the lower the coordinate, the higher the proportion of unscattered pattern). Region 6 was populated using a sampling of amorphous diffraction data from a real dataset. All patterns than had a Poissonian noise filter applied to simulate real detector noise.

**Simulated dataset—VAE training**

VAE training was performed on the Diamond Light Source jupyterhub \(^{42}\) equipped with 16 CPU cores, 128 GB RAM, Nvidia Tesla V100 GPU and Cuda 10.1. The training was performed on the full dataset of 87,320, 128 × 128 diffraction patterns. As the training was conducted on the full dataset, performance was evaluated on the full dataset as the model was not required to perform on patterns outside of this domain. The training was performed using the Adam optimiser \(^{48}\), using a learning rate of 0.0001. The training was executed for 1000 epochs, with the model weights saved at each epoch that showed a reduction in validation loss.

**Precipitates on Al alloy foil—synthesis**

The synthetic preparation of the precipitates on Al foil is commercially sensitive to our collaborators and is not relevant to the scope of discussion within this work.

**Precipitates on Al alloy foil—data collection**

This dataset was collected on the JEOL Grand-ARM 300F microscope at ePSIC, using pencil-beam alignment, operating at 200 kV accelerating voltage. The 4D-STEM data were collected on a single chip MerlinEM detector from Quantum Detectors (256 × 256 pixel array). The pencil beam was set up by turning the probe spherical aberration corrector optics off and reducing the probe convergence semi-angle by gradually changing the lens excitations in the pre-specimen optics to lower the cross-over of the beam. The final convergence semi-angle of the probe using the 10 μm condenser lens aperture was around 1.5 mrad.

**Precipitates on Al alloy foil—VAE training**

VAE training was performed on the Diamond Light Source jupyterhub \(^{42}\) equipped with 16 CPU cores, 128 GB RAM, Nvidia Tesla V100 GPU and Cuda 10.1. The training was performed on a dataset of 8650, 128 × 128 diffraction patterns. The raw 256 × 256 patterns were cropped to select the central 128 × 128 pixels. The training dataset was sampled from the precipitate regions identified by the masking method described in ‘Case Studies’. The training was performed using the Adam optimiser \(^{48}\), using a
learning rate of 0.0001. The training was executed for 1000 epochs, with the model weights saved at each epoch that showed a reduction in validation loss.

Layered metal oxide—synthesis
The synthesis of the layered metal oxide was a standard ceramic synthesis of sodium ion battery cathode material. The material had a target chemical composition of Na2/3Mn2/3Ni1/3O2. The procedure for synthesis was as follows:

1. In a glove box under an inert argon atmosphere, 1.7065 g of Na2CO3, 1.9112 g of NiCO3 and 2.7995 g of MnO2 were weighed out.
2. These powders were ground together in a pestle and mortar until visibly homogeneous.
3. The powder mixture was transferred into a tungsten carbide ball mill vessel along with 75 g of 3 mm tungsten carbide ball bearings and 10 g of acetone.
4. The vessel was sealed and removed from the glove box.
5. The mixture was milled for 60 min at 150 rpm with a direction inversion every 15 min.
6. The acetone was then evaporated off using a vacuum oven set to 333 K.
7. The dried mixture was pressed into a pellet using an 18-tonne load applied for 30 s.
8. The pellet was then fired in a tube furnace, under air, at 1173 K for 8 h, using a heating and cooling ramp rate of 5 K/min.
9. The cooled pellet was then transferred back to a glove box where it was reground by hand using mortar and pestle.
10. At a later date, a small quantity of powder was transferred into a weighing boat, further broken down using a spatula, and dispersed on a Lacey Carbon TEM Cu finder grid.

Layered metal oxide—data collection
This dataset was collected on the JEOL Grand-ARM 300F at ePSIC, in similar optical arrangement as above, operated at 300 kV accelerating voltage. A quad MerlinEM detector from quantum ePSIC, in similar optical arrangement as above, operated at 300 kV accelerating voltage. A quad MerlinEM detector from quantum

Layered metal oxide—VAE training
VAE training was performed on the Diamond Light Source jupyterhub equipped with 16 CPU cores, 128 GB RAM, Nvidia Tesla V100 GPU and Cuda 10.1. The training was performed on the dataset of 50,000, 128 × 128 diffraction patterns. The raw 476 × 476 patterns were cropped to select the central 256 × 256 pixels and these patterns were downsampled by a factor of 2 to obtain 128 × 128 patterns. The training dataset was sampled using the Shannon entropy of the patterns within the full dataset to define a probability distribution. Training performance was evaluated on a validation set containing 5000 randomly selected patterns. The training was performed using the Adam optimiser, using a learning rate of 0.0001. The training was executed for 1000 epochs, with the model weights saved at each epoch that showed a reduction in validation loss.

VAE model selection
The final weight selection is done by visually inspecting, at various stages of the training, the decoded points from different regions in the latent-space. To aid this inspection, a regular grid of points, spanning the region of the latent-space populated by encoded data, can be sampled and passed through the decoder to form a new 4D-STEM ‘latent-space dataset’. This new dataset then allows quick investigation of the patterns associated with the different regions of the latent-space. To select an appropriate model, the final set of weights from the training was selected, the data encoded and the distribution of the points in the latent-space viewed. Separated regions were observed in the encoded points and the patterns associated with these regions were inspected using the latent-space dataset. If the differences in the patterns associated with these regions were deemed to be experimentally relevant (showing meaningful variation in intensity and/or Bragg peak location) then the model was used. If the differences in the regions were not relevant (such as showing variation in background noise or insignificant variation in single peak intensities) then the model was deemed to be overtrained and an earlier model was investigated. The selection of a model is a balance of time and performance. Some level of overfitting is manageable due to the workflow’s flexibility in domain selection, and for a preliminary high-throughput investigation, saving time on optimising the model performance is likely preferable. For an in-depth study of a sample, such as in this paper, however, selecting a model that is well-tuned to the experimental goals will give a better domain mapping performance.

Root mean squared error
The root mean squared error (RMSE) is given by Equation (3)

\[ \text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\hat{x}_i - x_i)^2} \]  

where \( \hat{x}_i \) is the expected value of \( x \) for \( i \), \( x_i \) is the observed value of \( x \) for \( i \) and \( n \) is the number of observations.

Kullback–Leibler divergence
Kullback–Leibler divergence \( D_{KL}(P||Q) \) is a statistical measure of the divergence between two distributions \( P \) and \( Q \) given by Equation (4)

\[ D_{KL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \]  

Kernel density estimation
Kernel density estimation (KDE) is a non-parametric approach to estimating the population density from a sample. This is done using scikit-learn, which utilises Tree searches for efficient nearest neighbour queries. For the set of \( d \)-dimensional samples \( x_i \), \( i = 1, 2, \ldots, n \) in \( \mathbb{R}^d \), sampled from a parent population with density function \( p(x) \), the kernel density estimate at point \( y \), \( p_k(y) \) can be given by Equation (5)

\[ p_k(y) = \frac{1}{N} \sum_{i=1}^{N} K(y - x_i; h) \]  

where \( K(x; h) \propto e^{-\frac{x^2}{2h^2}} \) for a Gaussian Kernel.

Structural similarity index
The structural similarity index (SSI) is an image comparison metric implemented in skimage.

Non-negative matrix factorisation
Non-negative matrix factorisation is a means of factorising a matrix into a linear combination of a set of basis vectors such that

\[ V \approx WH \]  

where \( V \) is the matrix, \( W \) is the transformed data and \( H \) is the basis vector. The constraint of NMF is that \( V, W \) and \( H \) are all entirely \( >0 \). Equation (6) does not have a closed-form solution, so can be solved by iteration using different algorithms. The algorithm used in this workflow is built into PyXem.
DATA AVAILABILITY

The datasets and model weights for both the Layered Metal Oxide (P2 Sample) and the simulated dataset are available at https://doi.org/10.5281/zenodo.6839129. The dataset and model weights for the precipitates in AI alloy thin foil may be made available by our industrial collaborator upon reasonable request. Worked jupyter notebooks are available on Github.19

Received: 25 July 2022; Accepted: 22 December 2022; Published online: 21 January 2023

REFERENCES

1. Ophus, C., Ern, P., Saraham, M., Czarnik, C. & Ciston, J. Recording and using 4D-STEM datasets in materials science. Microsc. Microanal. 20, 62–63 (2014).
2. Buniciu, A. A., Gabriela Udritiu, E. & Aboul-Enein, H. Y. X-ray diffraction: instrumentation and applications. Crit. Rev. Anal. Chem. 45, 289–299 (2015).
3. Cheetham, A. K. & Taylor, J. C. Profile analysis of powder neutron diffraction data: its scope, limitations, and applications in solid state chemistry. J. Solid State Chem. 21, 253–275 (1977).
4. James, E. M. & Browning, N. D. Practical aspects of atomic resolution imaging and analysis in STEM. Ultramicroscopy 78, 125–139 (1999).
5. MacLaren, I., MacGregor, T. A., Allen, C. S. & Kirkland, A. I. Detectors for electron microscopy. J. Microsc. 236, 243–259 (2009).
6. Ophus, C. Four-dimensional scanning transmission electron microscopy (4D-STEM): from scanning nanodiffraction to pyctograpy and beyond. Microsc. Microanal. 25, 563–582 (2019).
7. Lualainen, J. E. M. et al. Mapping non-crystalline nanostructure in beam sensitive systems with low-dose scanning electron pair distribution function analysis. Microsc. Microanal. 25, 1636–1637 (2019).
8. Johnstone, D. N. et al. Direct imaging of correlated defect nanodomains in a metal-organic framework. J. Am. Chem. Soc. 142, 13081–13089 (2020).
9. Brunetti, G. et al. Confirmation of the domino-cascade model by LiFePO4/FePO4 precession electron diffraction. Chem. Mater. 23, 4515–4524 (2011).
10. Kobler, A., Kashkar, A., Hahn, H. & Kübel, C. Combination of in situ straining and ACOM TEM: A novel method for analysis of plastic deformation of nanocrystalline metals. Ultramicroscopy 128, 68–81 (2013).
11. Gallagher-Jones, M. et al. Nanoscale mosaicity revealed in peptide microcrystals by scanning electron nanodiffraction. Commun. Biol. 2, 1–8 (2019).
12. Nalin Mehta, A. et al. Unravelling stacking order in epitaxial bilayer MX2 using 4D-STEM with unsupervised learning. Ultramicroscopy 231, 113517 (2022).
13. Ortiz, G. C., Zhu, M., Dou, L. & Hwang, J. 4D-STEM quantification of nanoscale ordered domains in organic semiconductor polymers. Microsc. Microanal. 26, 1740–1742 (2020).
14. Kallin, S. V. et al. Deep Bayesian local crystallography. npj Comput. Mater. 7, 181 (2021).
15. Ophus, C. et al. Automated crystal orientation mapping in py4DSTEM using sparse correlation matching. Microsc. Microanal. 28, 390–403 (2022).
16. Cauteaerts, N. et al. Free, flexible and fast: orientation mapping using the multi-core and GPU-accelerated template matching capabilities in the Python-based open source 4D-STEM analysis toolbox Pyxem. Ultramicroscopy 237, 113517 (2022).
17. Meyer, T. et al. Phase transitions in a perovskite thin film studied by environment in situ heating nano-beam electron diffraction. Small Methods 5, 9 (2021).
18. Liu, H. et al. Manganese-based layered oxide cathodes for sodium ion batteries. Nano Select 1, 200–225 (2020).
19. Valery, A., Rauch, E. F., Clemen, L. & Lorut, F. Retrieving overlapping crystals information from TEM nano-beam electron diffraction patterns. J. Microsc. 268, 208–218 (2017).
20. Vincent, R. & Midgley, P. A. Dual conical beam-rocking system for measurement of integrated electron diffraction intensities. Ultramicroscopy 53, 271–282 (1994).
21. Bruefach, A., Ophus, C. & Scott, M. C. Analysis of interpretable data representations for 4d-stem using unsupervised learning. Microsc. Microanal. 28, 1998–2008 (2022).
22. Shi, C. et al. Uncovering material deformations via machine learning combined with four-dimensional scanning transmission electron microscopy. npj Comput. Mater. 8, 114 (2021).
23. Oviedo, F., Ferrer, J. L., Buonassisi, T. & Butler, K. T. Interpretable and explainable machine learning for materials science and chemistry. Acc. Mater. Res. 3, 597–607 (2022).
24. Liu, Y. et al. Experimental discovery of structure–property relationships in ferroelectric materials via active learning. Nat. Mach. Intell. 4, 341–350 (2022).

ACKNOWLEDGEMENTS

We thank Diamond Light Source for access and support in the use of the electron Physical Science Imaging Centre (Instrument E02 and proposal numbers EM19064 and MG28749) that contributed to the results presented here. A.B. gratefully acknowledges support from the Oxford-Diamond-STFC studentship scheme. T.J.W. and W.I.F.D. also acknowledge the Faraday Institution (Grant No. FIRG018) for their funding contributions.

AUTHOR CONTRIBUTIONS

Based on the CRediT taxonomy, the authors have made the following contributions: A.B.—Conceptualisation, data curation, formal analysis, investigation, methodology, software, visualisation, writing—original draft, writing—review and editing, W.I.F.D.—Conceptualisation, funding acquisition, project administration, resources, supervision, writing—review and editing, T.J.W.—Conceptualisation, funding acquisition, project administration, resources, supervision, writing—review and editing,

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences npj Computational Materials (2023) 14
M.D.—Conceptualisation, funding acquisition, formal analysis, methodology, project administration, resources, supervision, writing—review and editing. K.T.B.—Conceptualisation, funding acquisition, formal analysis, methodology, project administration, supervision, writing—review and editing.

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/s41524-022-00960-y.

Correspondence and requests for materials should be addressed to M. Danaie or K. T. Butler.

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.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.

© The Author(s) 2023