Outlier Detection in Large Radiological Datasets using UMAP

Mohammad Tariqul Islam MTISLAM@PRINCETON.EDU
Jason W. Fleischer JASONF@PRINCETON.EDU
Department of Electrical and Computer Engineering,
Princeton University, NJ, USA

Abstract

The success of machine learning algorithms heavily relies on the quality of samples and the accuracy of their corresponding labels. However, building and maintaining large, high-quality datasets is an enormous task. This is especially true for biomedical data and for meta-sets that are compiled from smaller ones, as variations in image quality, labeling, reports, and archiving can lead to errors, inconsistencies, and repeated samples. Here, we show that neighbor embedding algorithms can find these anomalies, essentially by forming independent clusters that are distinct from the main ("good") data but similar to other points with the same error type. As a representative example, we apply UMAP to discover outliers in the publicly available ChestX-ray14, CheXpert, and MURA datasets. While the results are archival and retrospective, and focus on radiological images, the graph-based methods work for any data type and will prove equally beneficial for curation at the time of dataset creation.

Keywords: neighbor embedding, UMAP, dataset curation, data visualization, representation learning, unsupervised learning

1. Introduction

A prominent reason behind the current success of machine learning-based disease detection is the availability of large medical datasets. However, for the machine learning models to be reliable, quality datasets representative of the target population need to be ensured (Yu et al., 2018). The labels in these datasets are often generated from human annotations using automated extraction or entity detection tools. However, these annotations (and their archiving) can have errors due to faulty perceptions and interpretations. Even if the error rate of the annotator is less than 4%, this can lead to millions of annotation errors per year (Bruno et al., 2015). Despite having a structured way of evaluating medical images, human errors are still inevitable (Waite et al., 2017). Thus, there needs to be a better way to identify such errors before they are included in a dataset.

For images, the search can be performed visually. However, examining individual images is a daunting task that requires many human hours. A popular automated alternative is neighbor embedding (Hinton and Roweis, 2002), which can produce a two-dimensional (2-D) cluster plot that can be analyzed visually quickly. (This class of methods is also known as nonlinear dimensionality reduction as the 2-D plot is obtained by preserving the pairwise similarity of the original high-dimensional structure.) Widely used neighbor embedding algorithms are t-distributed stochastic neighbor embedding (t-SNE) (Maaten and Hinton, 2008) and uniform manifold approximation and projection (UMAP) (McInnes et al., 2018). UMAP was introduced relatively recently and has become very popular, as this method has rich algebraic and topological structure and is computationally fast.

In this paper, we propose a UMAP-based visual analytic method for extracting outlier images from large x-ray datasets. We validate the framework by analyzing three publicly available and widely used medical image datasets. We show that the method can successfully cluster image features and produce interpretable visualization. We also discover labeling errors and erroneous images that have slipped through the verification process done prior to dissemination.

2. Related Works

In the literature the term outlier is often used interchangably with abnormality and anomaly (Fritsch et al., 2012). Here, we define outliers as images that do not have any signal necessary for final decision-making or do not belong in the dataset due to specification. Generally, outlier detection methods assume

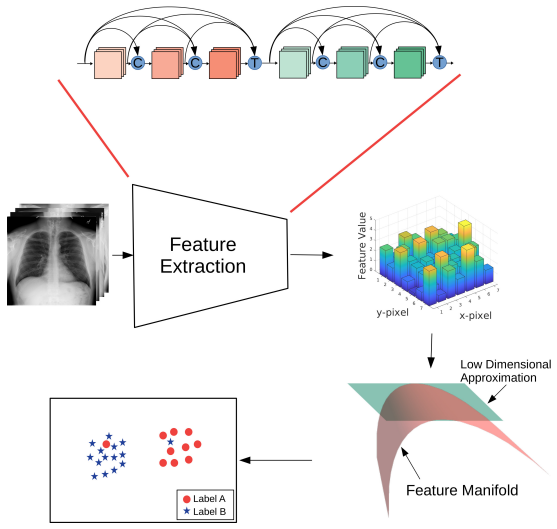


Figure 1: **Schematic of the outlier search algorithm.** Image features extracted from a DenseNet-121 neural network are projected onto a 2-D plane using UMAP.

that there is an underlying distribution of data which is often modeled to be a normal distribution (Hodge and Austin, 2004). A data point is an outlier if it is far away from the mean of the fitted distribution. Fritsch et al. (2012) used the minimum covariance determinant estimator and its extensions to find outliers (due to motion or registration issues) in neuroimaging data by analyzing principal components. Gang et al. (2018) used a t-SNE plot to find outliers from binary lung masks in terms of size variation and segmentation error. Fleischer and Islam (2020) employed UMAP on chest x-rays for phenotyping COVID-19 response.

3. Method

Following preprocessing (discussed in Appendix A), the major parts of the framework are feature extraction and dimensionality reduction (Fig. 1). To extract features from these images, we employed DenseNet-121 (Huang et al., 2017) trained on ImageNet (Russakovsky et al., 2015), a widely used deep learning architecture designed to efficiently propagate features from earlier layers of a network to deeper layers. Importantly, neural features are usually robust to many variabilities in images, and thus can accommodate standard images and outliers on equal terms.

Medical images usually vary in resolution, have different contrast, brightness, and alignment, and often suffer from registration issues. While pre-trained DenseNet using radiological data is available, we chose to use ImageNet models so that the network is not biased by radiologiy-specific data. In our framework, the features have been extracted from the final layer (before the softmax layer) of the network, where the features are generally most discriminating. Since we are not using a radiologically pre-trained model, these features generally will not be able to identify individual diseases. Rather, we employ other related labels (e.g., x-ray views and body parts) to examine the datasets. After extracting the features, we employ a neighbor embedding algorithm, UMAP (McInnes et al., 2018), to obtain a 2-D approximation of the high-dimensional features. Thus, images that are similar will be placed close to each other after the embedding.

4. Results

In this section, we describe the datasets and discuss representative results.

4.1. Data

We evaluate our approach on three publicly available datasets: ChestX-ray14 (Wang et al., 2017), CheXpert (Irvin et al., 2019), and Musculoskeletal Radiographs (MURA) (Rajpurkar et al., 2018). ChestX-ray14 contains 112,120 frontal chest x-ray images from 30,805 unique patients. Images are from posterior-anterior (PA) and anterior-posterios (AP) views. CheXpert dataset contains 224,316 chest x-rays (PA, AP, and Lateral) from 65,240 patients. We used 223,414 JPEG formatted x-rays from the training set of the dataset. MURA dataset contains 40,561 musculoskeletal x-rays from 14,863 studies. Similar to CheXpert, we used 36,808 x-rays from the training set for analysis. Additional details are provided in Appendix A.

4.2. Experiments

4.2.1. Lateral X-rays in ChestX-ray14

Our 2-D embedding is shown in Fig. 2(a). There are two large clusters of PA and AP views, corresponding to the primary tolopogy of the DenseNet features.

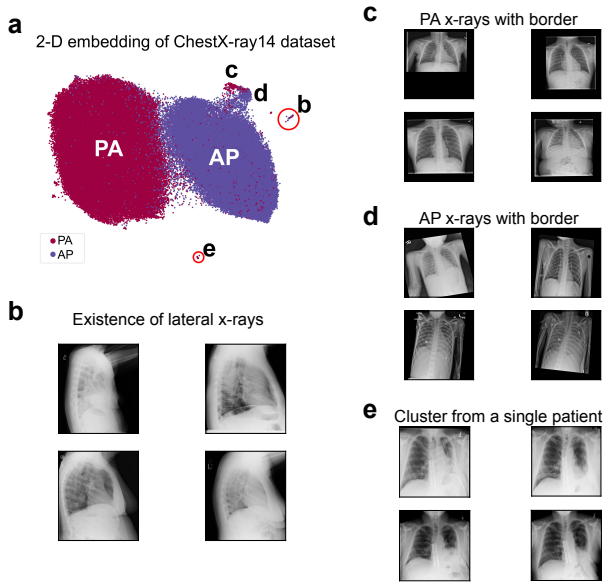


Figure 2: **Neighbor embedding of the ChestX-ray14 dataset. a)** 2-D embedding. Labeled clusters from (a) are: **b)** Lateral x-rays which were not supposed to be in the dataset, **c)** PA x-rays with borders, **d)** AP x-rays with borders, and **e)** cluster from a single patient.

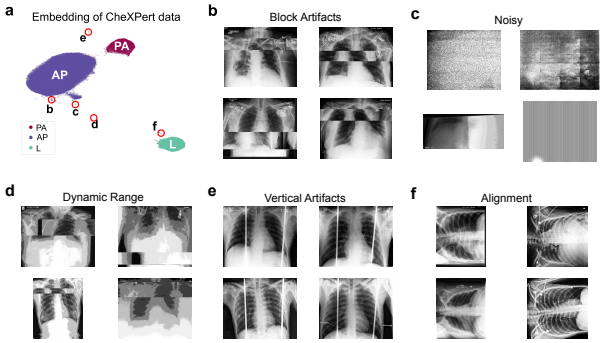


Figure 3: **Neighbor embedding of x-rays from CheXpert dataset. a)** 2-D Embedding. Example images with **b)** block artifacts, **c)** noise, **d)** improper dynamic range, **e)** vertical artifacts, and **f)** alignment issues.

The satellite clusters around the larger ones occur because the nearest neighbor graph creates a loop (or isolated sub-graph) of common features that are distinct from the rest of the data (more details in Appendix A). In most cases, each of the satellite clusters of x-rays is from a single patient with a unique signature. However, if any specific image features (such as similar artifacts in multiple images) are present in x-rays of different patients, these can create satellite clusters as well.

Representative examples of anomalous clusters are shown in Figs. 2 (b)-(e). The most surprising finding is the existence of some lateral x-rays in the dataset (Fig. 2 (b)), as this dataset is supposed to be composed of frontal chest x-rays only. We found 92 lateral x-rays using our method. The images are listed in the supplement.

Another interesting structure in Fig. 2 (a) is the protruding region from the AP cluster marked c and d, consisting of x-rays with dark borders of PA and AP views, respectively. Finally, cluster (e) shown in Fig. 2 (e) groups 46 x-rays from patient ID 9845 and a single x-ray from patient ID 12562.

4.2.2. Corrupted Images in CheXpert

The 2-D embedding of CheXpert dataset is shown in Fig. 3 (a). As before, the large PA and AP clusters form the bulk of the mapping. The lateral x-rays also form a separate large cluster. A few of the large satellite clusters (b-e) have been marked by red circles in Fig. 3 (a). Four images from each of the clusters are plotted in Fig. 3 (b)-(e). Fig. 3 (b) depicts images with block artifacts, e.g., from poor JPEG compression or accidental splicing. We found 107 such images in this cluster. Fig. 3 (c) depicts images that are just noise. This cluster contains 19 such images. Fig. 3 (d) shows four images with dynamic range issues in addition to block artifacts. The cluster contains 53 such images. Fig. 3 (e) shows four representative x-rays with vertical artifacts. A total of 88 such images are found in this cluster. Finally, Fig. 3 (f) shows are rotated images. This cluster is placed near the large cluster of lateral x-rays. Thus, DenseNet considers rotated x-rays to be more similar to lateral images than upright frontal X-rays. Additional discussion is in the Appendix B.

4.2.3. Extracting Chest X-rays from MURA

Since the MURA dataset consists of x-rays from different parts of the arm and the shoulder, there is a natural ambiguity in labels, e.g., both wrist and hand

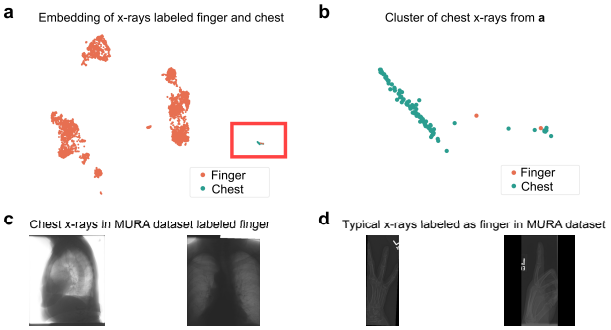


Figure 4: **Neighbor embedding of 'finger' x-rays from MURA dataset and 100 chest x-rays from CheXpert dataset.** **a)** Scatterplot of the embedding. The cluster of chest x-rays is marked using a red rectangle. **b)** Scatterplot in the red rectangle. **c)** two x-rays labeled 'finger' are actually chest x-rays. **d)** Typical finger x-rays from the MURA dataset.

x-rays may contain the hand of a person, and shoulder x-rays may contain part of the chest. In such cases, finding mislabeled x-rays by embedding all the images may be sub-optimal. To find outliers more directly, we searched for misclassified images by explicitly using labels of the dataset. The method has two parts:

1. Introduce target images with a specific label (preferably from a different dataset than the MURA one); and

2. Perform neighbor embedding on the joint dataset.

For example, to look for possible chest x-rays falsely classified as finger x-rays in the MURA dataset, we added 100 chest x-rays from the CheXpert dataset to the 5,106 finger x-rays of MURA. We then applied the UMAP to the composite set (Fig. 4 (a)). As shown in Figs. 4 (b)-(d), the seeded chest x-rays acted as an attractor for mislabeled images in MURA, with x-rays labeled 'finger' now appearing the (new) chest cluster. Interestingly, both of these x-rays were from patient 04547 (another 3 from this patient were labeled correctly). Another example, that of leg x-rays mislabeled as "shoulder", is shown in Appendix C.

5. Discussion and Future

Neighbor embedding algorithms can be an effective tool for summarizing datasets and identifying outlier images. The principle of the method is that the outliers are different from the main data but they can have similarities among themselves. Thus, the outliers form distinct clusters in the embeddings. Our experiments, using a DenseNet-121 feature extractor and UMAP neighbor embedding method on the ChestX-ray14, CheXpert, and MURA datasets distinguished different radiological views of chest x-rays, classified different, and identified wrongly labeled or corrupted images. We further found specific types of outliers by seeding the dataset with target images and performing neighbor embedding.

One of the major limitations of our approach is that the embedding quality depends on the quality of the image features obtained from the neural network. To avoid contamination of features from x-ray images in the pre-trained models, we used a network trained on ImageNet. Future work will consider different datasets for the pre-training, including those designed specifically for biomedical imaging.

It may also help to consider other algorithms, such as self-supervised and foundation models. Likewise, other neighbor embedding techniques, such as TriMAP (Amid and Warmuth, 2019), PaCMAP (Wang et al., 2021), and combined feature learning and embedding (Böhm et al., 2023) may be beneficial. For larger datasets, more accurate resutls and faster embedding may be achieved by dividing them into smaller subsets and applying better alignment techniques (Islam and Fleischer, 2022). Correlating statistic outlier detection methods (Han et al., 2022) with the UMAP embeddings for improved explainability may also be explored.

While this study performed retrospective analysis of large x-ray datasets, outlier curation can be achieved during the initial assembly of the dataset as well. For suspected outliers, the method of seeding data with known labels can be applied. To streamline the process, appropriate reference datasets may be created beforehand. Undoubtedly, cleaner input data will result in cleaner output data.

Finally, since the methods are graph-based and agnostic to the underlying data type, all of the methods here can be applied to arbitrary datasets, including and especially those that are mixed modality.

References

Ehsan Amid and Manfred K Warmuth. TriMap: Large-scale dimensionality reduction using triplets. *arXiv preprint arXiv:1910.00204*, 2019.

Jan Niklas Böhm, Philipp Berens, and Dmitry Kobak. Attraction-repulsion spectrum in neighbor embeddings. *The Journal of Machine Learning Research*, 23(1):4118–4149, 2022.

Niklas Böhm, Philipp Berens, and Dmitry Kobak. Unsupervised visualization of image datasets using contrastive learning. In *The Eleventh International Conference on Learning Representations*, 2023.

Michael A Bruno, Eric A Walker, and Hani H Abujudeh. Understanding and confronting our mistakes: the epidemiology of error in radiology and strategies for error reduction. *Radiographics*, 35(6): 1668–1676, 2015.

Jason W Fleischer and Mohammad Tariqul Islam. Identifying and phenotyping COVID-19 patients using machine learning on chest x-rays. *European Respiratory J.*, 56 (suppl 64), 2020.

Virgile Fritsch, Gaël Varoquaux, Benjamin Thyreau, Jean-Baptiste Poline, and Bertrand Thirion. Detecting outliers in high-dimensional neuroimaging datasets with robust covariance estimators. *Medical image analysis*, 16(7):1359–1370, 2012.

Peng Gang, Wang Zhen, Wei Zeng, Yuri Gordienko, Yuriy Kochura, Oleg Alienin, Oleksandr Rokovyi, and Sergii Stirenko. Dimensionality reduction in deep learning for chest x-ray analysis of lung cancer. In *2018 tenth international conference on advanced computational intelligence (ICACI)*, pages 878–883. IEEE, 2018.

Songqiao Han, Xiyang Hu, Hailiang Huang, Minqi Jiang, and Yue Zhao. ADBench: Anomaly detection benchmark. *Advances in Neural Information Processing Systems*, 35:32142–32159, 2022.

Geoffrey Hinton and Sam T Roweis. Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems*, volume 15, pages 833–840, 2002.

Victoria Hodge and Jim Austin. A survey of outlier detection methodologies. *Artificial intelligence review*, 22(2):85–126, 2004.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019.

Mohammad Tariqul Islam and Jason W Fleischer. Manifold-aligned neighbor embedding. In *ICLR 2022 Workshop on Geometrical and Topological Representation Learning*, 2022.

George C Linderman and Stefan Steinerberger. Clustering with t-SNE, provably. *SIAM Journal on Mathematics of Data Science*, 1(2):313–332, 2019.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Pranav Rajpurkar, Jeremy Irvin, Aarti Bagul, Daisy Ding, Tony Duan, Hershel Mehta, Brandon Yang, Kaylie Zhu, Dillon Laird, Robyn L Ball, et al. MURA Dataset: Towards radiologist-level abnormality detection in musculoskeletal radiographs. In *Medical Imaging with Deep Learning*, 2018.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng

Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014.

Jian Tang, Jingzhou Liu, Ming Zhang, and Qiaozhu Mei. Visualizing large-scale and high-dimensional data. In *Proceedings of the 25th international conference on world wide web*, pages 287–297, 2016.

Stephen Waite, Jinel Scott, Brian Gale, Travis Fuchs, Srinivas Kolla, and Deborah Reede. Interpretive error in radiology. *American Journal of Roentgenology*, 208(4):739–749, 2017.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

Yingfan Wang, Haiyang Huang, Cynthia Rudin, and Yaron Shaposhnik. Understanding how dimension reduction tools work: an empirical approach to deciphering t-SNE, UMAP, TriMAP, and PaCMAP for data visualization. *The Journal of Machine Learning Research*, 22(1):9129–9201, 2021.

Richard E. Woods and Rafael C. Gonzalez. Digital image processing, 2008.

Kun-Hsing Yu, Andrew L Beam, and Isaac S Kohane. Artificial intelligence in healthcare. *Nature biomedical engineering*, 2(10):719–731, 2018.

Appendix A. Implementation Details

In this section, we provide details of the method, implementation, and datasets.

A.1. Image Pre-processing

We apply the following transformations: histogram equalization (Woods and Gonzalez, 2008), resizing the images, center cropping, and normalization. Images are resized such that the lowest dimension contains 256 pixels and then center-cropped to a 224×224 dimensional image for feature extraction. Then the images are normalized according to the specification of Imagenet: mean (0.485, 0.456, 0.406) and standard deviation (0.229, 0.224, 0.225) of red, green, and blue channels, respectively.

A.2. Feature Extraction

We use a DenseNet-121 architecture (Huang et al., 2017) pre-trained on the ImageNet dataset (Russakovsky et al., 2015) from the PyTorch deep learning library (Paszke et al., 2017). (DenseNet is a deep neural network with many inter-layer connections designed to reduce the numerical instabilities that originate due to the depth of the network. The usage of neural network outputs as features is an effective baseline in machine learning algorithms (Sharif Razavian et al., 2014) and is extensively used in medical image analysis.) Here, we remove the classification (softmax) layer from the neural network and use the output of the last layer as the feature set.

Let, $f(\cdot;\theta)$ be the feature extractor parameterized by $\{\theta\}$. Then, the feature $\mathbf{x}$ obtained from a pre-processed image $\mathbf{I}$ is given by

$$\mathbf{x} = f(\mathbf{I};\theta) \tag{1}$$

A.3. Neighbor Embedding

The first step is to characterize the high-dimensional structure of the feature set $\mathbf{X} = \{\mathbf{x}_i \in \mathbb{R}^n | i = 1,\ldots,N\}$ using a pairwise metric. More specifically, we create a graph with the adjacency matrix:

$$p_{ij} = f_H(d_x(\mathbf{x}_i,\mathbf{x}_j)|\mathbf{X}), \tag{2}$$

where $d_x(\cdot,\cdot)$ is the pairwise distance metric and $f_H(\cdot)$ is a function defining the weight of the edge.

Given the set $\mathbf{Y} = \{\mathbf{y}_i \in \mathbb{R}^d | i = 1,\ldots,N\}$ of the corresponding low-dimensional approximation of the features $\mathbf{X}$ (typically $d << n$), the graph in the low dimension is given by the adjacency matrix:

$$q_{ij} = f_L(d_y(\mathbf{y}_i,\mathbf{y}_j)|\mathbf{Y}), \tag{3}$$

where $d_y(\cdot,\cdot)$ is a pairwise distance metric used for low-dimensional embedding and $f_L(\cdot)$ is a function that provides the weight of the graph edges.

Finally, the low-dimensional embedding is optimized from an initialization of the set Y by minimizing a loss function,

$$\mathcal{L} = \sum_{i,j} l(p_{ij}, q_{ij}) \tag{4}$$

A.4. Uniform Manifold Approximation and Projection (UMAP)

UMAP constructs a high-dimensional graph of the original dataset by the following system of equations:

$$p_{i,j} = p_{i|j} + p_{j|i} - p_{i|j}p_{j|i}, \tag{5}$$

$$p_{i|j} = \begin{cases} \exp\left(-\frac{d(\mathbf{x}_i,\mathbf{x}_j)-\rho_i}{\sigma_i}\right) & \text{if } x_j \in \text{KNN}(\mathbf{x}_i, k) \\ 0 & \text{otherwise} \end{cases}, \tag{6}$$

$$\rho_i = \min_{\mathbf{x}_j \in \text{KNN}(\mathbf{x}_i,k)} d(\mathbf{x}_i, \mathbf{x}_j), \tag{7}$$

where $\text{KNN}(\mathbf{x}_i, k)$ is the set of k-nearest neighbors of the point $\mathbf{x}_i$ and σ_i is a scaling parameter such that $\sum_j p_{i|j} = \log_2(k)$.

The low-dimensional graph is given by a differentiable function

$$q_{ij} = \frac{1}{1 + a(||\mathbf{y}_i - \mathbf{y}_j||_2^2)^b}, \tag{8}$$

where the parameters a and b determine the density of the mapping. a and b are chosen by fitting q_{ij} to

$$\Psi(d_{ij}) = \begin{cases} 1 & \text{if } d_{ij} < m_d \\ \exp(-(d_{ij} - m_d)) & \text{otherwise} \end{cases} \tag{9}$$

where $d_{ij} = ||\mathbf{y}_i - \mathbf{y}_j||_2$, and m_d regulates the minimum distance between the two nearest low-dimensional points. This later parameter ensures that if the minimum distance between points is small, then neighboring points come close to each other forming compact clusters; otherwise, the points are spread out.

UMAP aims to minimize the following cross-entropy loss function:

$$\mathcal{L} = \sum_{i,j} p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right) + (1 - p_{ij}) \log\left(\frac{1 - p_{ij}}{1 - q_{ij}}\right). \tag{10}$$

The first term provides an attractive force and the second term provides a repulsive force. Instead of optimizing every point in each iteration, UMAP takes the negative sampling (Mikolov et al., 2013; Tang et al., 2016) approach. For each edge with $p_{ij} > 0$, named a positive edge, several edges are sampled randomly, named negative edges. The attractive force is applied on the positive edge, whereas the repulsive force is applied on the negative edges (McInnes et al., 2018).

A.5. Data

Here, we describe each of the datasets we used in the main text.

A.5.1. ChestX-ray14

The dataset was initially compiled as a smaller dataset named ChestX-ray8 (Wang et al., 2017) with 8 abnormality labels. Later the dataset was expanded with additional images and labels resulting in 112,120 frontal chest x-ray images from 30,805 unique patients and 14 abnormality labels. The data has two perspective labels (PA and AP) and 14 abnormality labels.

A.5.2. CheXpert

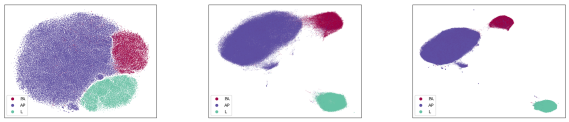
This dataset was compiled from chest radiographic studies collected from Stanford Hospital, performed between October 2002 and July 2017 (Irvin et al., 2019). The dataset contains 224,316 chest x-rays from 65,240 patients. We have used only the training set in our experiment which consists of 223,414 x-ray images.

A.5.3. Musculoskeletal Radiographs (MURA)

MURA is a large dataset of musculoskeletal radiographs (Rajpurkar et al., 2018). The dataset contains 40,561 musculoskeletal x-rays from 14,863 studies. The primary label is 'normal' or 'abnormal', with the latter indicating fractures, plates, and screws from operating procedures, degenerative changes, etc. Secondary labels are finger, wrist, hand, forearm, elbow, humerus, and shoulder. Similar to CheXpert, we used the training set only, which consists of 36,808 samples.

A.6. Parameter Settings

In general, we kept the number of nearest neighbors k to be low, as increasing k increases the computational budget.



(a) (b) (c)

Figure 5: **Embedding of CheXpert data using t-SNE and UMAP.** (a) t-SNE with no exaggeration. (b) t-SNE with exaggeration factor 4. (c) UMAP.

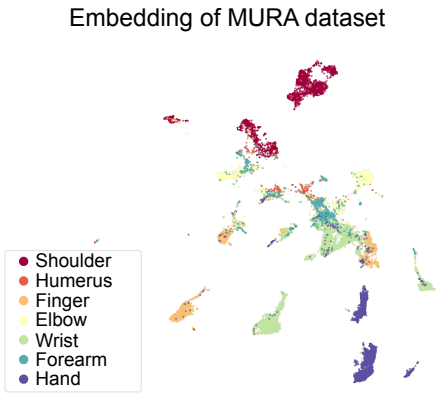


Figure 6: **Embedding of MURA dataset.**

Figure 7: **Chest x-rays chosen from CheXpert dataset to produce the embedding in Fig 4.**

ChestX-ray14 - Fig. 2: For ChestX-ray14 we used $k = 50$. The minimum distance parameter m_d was set to 0.1. We experimented with smaller m_d values to increase the separation of the AP and PA x-rays, but it had little effect. The embedding was optimized for 200 epochs.

CheXpert - Fig. 3: The embedding was obtained by using $k = 10$. We used a smaller minimum distance $m_d = 0.001$, as we found that this value provided a better separability of the large clusters. We ran the optimization for 300 epochs.

MURA - Figa. 4, 6, and 8: Similar to CheXpert, we used $k = 10$, $m_d = 0.001$, and ran the optimization for 300 epochs.

Appendix B. Comparison with t-SNE Algorithm

In the main text, we focused on the UMAP algorithm. Here, we briefly explore t-SNE. The default t-SNE is tuned to preserve the neighborhood as best as possible. This often causes the individual clusters to be spread out and less compact. The typical t-SNE output is shown in Figure 5 (a). Despite t-SNE being able to cluster the PA, AP, and lateral x-rays, the separation among them is minimal and there is little room for the satellite clusters. However, t-SNE can be tuned to produce a more UMAP-like output. Following the findings of Linderman and Steinerberger (2019) and Böhm et al. (2022), we used an exaggeration factor of 4, which means we applied 4 times more repulsive force than the attractive force. The standard early exaggeration factor of 12 was also applied at the start of the optimization. The resulting plot is shown in Fig. 5 (b). Comparison of the t-SNE plot to the UMAP plot reveals that most satellite clusters are absorbed within the larger PA and AP clusters. Overall, UMAP is superior.

Appendix C. Additional Discussion on MURA

Fig. 6 shows the embedding of 36,808 musculoskeletal radiographs from the MURA dataset. There is

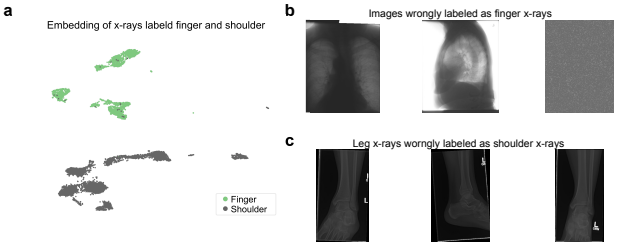


Figure 8: **Neighbor embedding of 'finger' and 'shoulder' x-rays from MURA dataset.** (a) Scatterplot of the embedding. (b) Chest x-ray and non-x-ray images were discovered which are labeled as 'finger' x-rays. (c) Leg x-rays labeled as 'shoulder' x-rays.

decent separation among the x-rays in terms of the labels, but there is also a considerable overlap. For example, finger, wrist, and hand x-rays overlap, as finger x-rays include parts of the wrist and hand, and vice versa. Similarly, wrist and forearm clusters are often merged, since x-ray of the forearm tend to capture a portion of the wrist, and vice versa. The same happens for humerus and shoulder. The images within each cluster thus share similar acquisition views, aspect ratios, and specific features (e.g., circular window function, stitching of multiple x-rays in one image). Unlike ChexPert case, analysis of this mapping did not reveal any satellite clusters with corrupted images.

Based on this, we decided to focus on individual labels and extract images with specific labels. The chest x-rays from the CheXpert dataset used to produce Fig. 4 are shown in Fig. 7.

C.1. Finding More Mislabeled Images

Here we used 'finger' and 'shoulder' x-rays to search for misclassified images (Fig. 8). If shoulder labels include images that look like 'finger', they will be attracted to the finger clusters, and vice versa.

As expected, the broad features of the finger and shoulder are easily separable with a few misclassified points. Analyzing 'finger' x-rays misclassified in 'shoulder' clusters, we can find the two chest x-rays labeled as finger (which we found in section 4.2.3 as well) and two images that are just noise/non-x-ray images (Fig. 8 (b)). The latter belongs to patient ID 04687, which includes two more outliers. One of the three is an x-ray of a collection of keys. This anomaly was not put into a different cluster by the algorithm but was discovered because of manually checking patient ID 04687. The misclassified 'shoulder' labels in the 'finger' cluster reveal three leg x-rays (Fig. 8 (d)). These should not be in the MURA dataset.