

Got it — let's compress everything so it comfortably fits into about **20 minutes** (~30–35 seconds per slide on average).

Below is a **short, speaker-ready script**, one block per slide, already shortened and streamlined. You can still map **1:1 to your 37 PowerPoint slides** (titles match your PPT).

Slide 1 – Distribution Shift in Medical Imaging

“Good morning, my name is Moataz Mansour, and this is our CAS AML project on **distribution shift in medical imaging**, together with Sharad George.

We train on **NIH ChestX-ray14** and then ask: *What happens when we deploy the model on very different datasets like a Chinese pediatric cohort or CheXpert from Stanford?*

Our idea is to use a **convolutional autoencoder**, trained only on NIH, as a kind of **distribution sensor** and see whether its reconstruction error can predict where a DenseNet classifier will struggle.”

Slide 2 – Why Distribution Shift Matters

“When we train a model, we usually assume training and deployment data are **i.i.d.**, but in practice, new hospitals use different scanners, protocols and treat different patient groups.

In chest X-rays, this means changes in contrast, noise, anatomy and disease mix.

If we ignore this and only trust the test AUC on NIH, we can end up with a model that looks great on paper but fails silently in a new hospital.

So, we need a way to **detect shift without labels**, especially in medicine.”

Slide 3 – Research Question & Approach

“We ask three main questions:

1. Can we detect distribution shift **unsupervised**, just from images, using autoencoder reconstruction error?
2. Does this unsupervised signal correlate with **actual classifier performance** across datasets?
3. What seems to matter more: pathology, demographics, or institutional differences?

To answer this, the notebook is structured into Phases 1–5: from pixel statistics, to autoencoders, to DenseNet performance, and finally to correlation analysis.”

Slide 4 – Datasets Used

“We work with three datasets.

First, **NIH ChestX-ray14**, a large adult US dataset: this is our **training domain** for both autoencoder and classifier.

Second, a **Chinese pediatric dataset** – children instead of adults, and from a different hospital.

Third, **CheXpert** from Stanford, again mainly adults but with different acquisition and labeling pipelines.

These give us shifts in institution, demographics and label prevalence.”

Slide 5 – Label Distributions

“Label distributions already show differences.

After preprocessing, **NIH** is relatively balanced between ‘No Finding’ and ‘Abnormal’.

The **pediatric** dataset is much more skewed toward pneumonia, and **CheXpert** is extremely skewed toward ‘abnormal’.

This means that **plain accuracy** can be very misleading, especially on CheXpert, where predicting ‘abnormal’ almost always is already ‘good’ accuracy.

That’s why we later rely on **balanced accuracy**.”

Slide 6 – Phase 1 – Baseline Distribution Shift

“In **Phase 1**, we don’t train any model yet.

We compare simple **pixel-intensity histograms** between NIH, Pediatric and CheXpert and compute **Jensen–Shannon divergence** between them.

We do this both on the full datasets and on subsets containing only normal images.

The goal is to see whether we already spot shift before we involve any deep learning.”

Slide 7 – Phase 1 – Key Findings

“JS divergence clearly shows that the pixel distributions of the three datasets are **not the same**.

More importantly, even when we restrict ourselves to **normal images only**, the divergences stay large.

So the differences cannot be explained just by disease; they also come from **institutional and technical factors** like scanners and post-processing.

This motivates going deeper with an autoencoder that can learn a richer representation of NIH.”

Slide 8 – Phase 2 – Convolutional Autoencoder

“Phase 2 introduces a **convolutional autoencoder**.

The encoder takes a $1 \times 224 \times 224$ X-ray, passes it through conv layers and compresses it into a **256-dimensional latent vector**.

The decoder reconstructs back to the original size.

We train this model on **NIH only**, using mean squared error as reconstruction loss.

We train two variants: one on **all NIH images**, and one on **NIH normals only**.”

Slide 9 – Phase 2 – Convolutional Autoencoder

“Practically, the autoencoder has tens of millions of parameters and is trained with **Adam** and standard settings.

The training curves in the notebook show smooth convergence and no big gap between training and validation loss.

This tells us the autoencoder generalizes well within NIH and is not simply memorizing. From now on, we **freeze** it and treat it as a fixed model of the NIH distribution.”

Slide 10 – Phase 2 – Training Results

“When we evaluate NIH_Normal AE on NIH, we see the lowest reconstruction error for **NIH normals**.

If we feed it **NIH abnormalities**, the error increases by roughly **+6–7%**.

This is our internal ‘pathology effect’ baseline: within one institution, pathology raises reconstruction error only slightly.

We’ll use this number later to compare against cross-dataset shifts, which turn out to be much larger.”

Slide 11 – Phase 2 – Training Results

“Visually, reconstructions of NIH normals look very similar to the inputs: lungs, heart and ribs are well preserved, with some smoothing of fine detail.

For NIH abnormalities, the autoencoder often **softens** opacities or consolidations, reflecting its tendency to reconstruct ‘typical’ anatomy.

But overall, the reconstructions are still quite close, which explains the relatively small error increase.

This confirms the model has really captured the **NIH data manifold**.”

Slide 12 – Phase 2 – Training Results

“These results show that the autoencoder is a **stable, expressive model of NIH**. Within this domain, moving from normal to abnormal is a small step in terms of reconstruction error. This gives us a concrete reference for ‘how big’ a change pathology alone causes. In Phase 3, we’ll see that cross-dataset differences are much larger than this internal pathology effect.”

Slide 13 – Phase 2 – Training Results

“To summarize Phase 2:

We now have a frozen autoencoder that defines a **latent manifold** for NIH chest X-rays, plus a scalar reconstruction error that measures how far a new image is from that manifold.

The NIH_Normal AE, in particular, tells us what ‘normal NIH’ looks like numerically and visually.

With this tool ready, we’re now prepared to probe the autoencoder with **Pediatric and CheXpert images**.”

Slide 14 – Phase 2 – Key Findings

“Key findings:

1. The NIH autoencoder converges and reconstructs NIH images well.
2. The error difference between NIH normals and abnormal is **small** (~6–7%).
3. Reconstruction error therefore looks like a reasonable **unsupervised measure** of how ‘NIH-like’ an image is.

Now we ask: what happens when we feed it images from **completely different datasets**?”

Slide 15 – Phase 3 – Reconstruction Error Across Datasets

“In **Phase 3**, we keep the autoencoder weights fixed and change only the dataset.

We compute reconstruction error on NIH, Pediatric, and CheXpert for both NIH_Full AE and NIH_Normal AE.

For NIH_Normal AE we also look specifically at **normal images** from each dataset.

The hypothesis: the further a dataset is from NIH, the higher its reconstruction error.”

Slide 16 – Phase 3 – Quantitative Results

“The numbers are clear.

Compared to NIH, **Pediatric** images have roughly **+100%** higher mean reconstruction error, and **CheXpert** often more than **+200%** higher.

For normals-only with NIH_Normal AE, the same pattern appears: Pediatric normals and CheXpert normals are much harder to reconstruct than NIH normals.

Statistical tests show these differences are highly significant.

So cross-dataset shifts are **far larger** than the internal pathology effect.”

Slide 17 – Phase 3 – Quantitative Results

“Looking at full distributions, the error histograms for Pediatric and CheXpert are clearly shifted upward relative to NIH.

There’s only limited overlap, and effect sizes are large.

This tells us that, from the autoencoder’s point of view, Pediatric and CheXpert images are **much more unusual** than diseased NIH lungs.

So the dominant source of shift is not ‘this patient has pneumonia’, but ‘this image is from a different hospital or population’.”

Slide 18 – Phase 3 – Quantitative Results

“Putting it together, reconstruction error behaves like a **continuous distance** from the NIH training distribution:

NIH → lowest error, Pediatric → higher, CheXpert → highest.

This makes reconstruction error a promising **early-warning signal**: if a new dataset shows error levels comparable to Pediatric or CheXpert, we should expect trouble for the classifier.

The next step is to visualize the latent space and then bring in the classifier.”

Slide 19 – Phase 3 – Interpretation

“The interpretation of Phase 3 is: the autoencoder has learned what NIH looks like, and it considers Pediatric and CheXpert to be **far off-manifold**.

Even healthy lungs from those datasets are seen as atypical compared to NIH.

This means institutional, technical and demographic factors are driving most of the shift, more than pathology itself.

From a deployment perspective, we cannot assume that a model learning ‘pneumonia vs normal’ at NIH will behave the same in a different hospital.”

Slide 20 – Phase 3c – Latent Space Visualization (t-SNE)

“In Phase 3c, we look inside the model’s **latent space** using t-SNE. Each image is encoded into a 256-dimensional vector, and t-SNE maps these vectors into 2D. When we color points by dataset, NIH, Pediatric and CheXpert form **distinct clusters**. So, even though the autoencoder was trained only on NIH, its representation naturally separates the other datasets as separate groups.”

Slide 21 – t-SNE and Classifier Performance

“How does this relate to performance?
The **NIH cluster** is where the classifier has seen data during training; there its decision boundary is reliable.
The Pediatric and CheXpert clusters live in regions where the classifier has never seen examples.
So when we apply the NIH-trained classifier there, it is extrapolating, which explains why **balanced accuracy drops** on these datasets.”

Slide 22 – t-SNE and Classifier Performance

“If we overlay reconstruction error onto the t-SNE plot, points near the NIH cluster have **low error**, while Pediatric and especially CheXpert points have **high error**. This reinforces the idea that reconstruction error is effectively a **radial distance** from the NIH manifold in latent space. The further away a cluster is, the more out-of-distribution it is, and the more we should expect the classifier to struggle.”

Slide 23 – t-SNE and Classifier Performance

“If we also mark **misclassified points** on this map, they tend to appear more often outside the dense NIH core and in the Pediatric/CheXpert regions. So latent distance, reconstruction error and misclassification all line up: the further a dataset is from NIH in latent space, the more errors we see. This gives a clear geometric intuition for why distribution shift harms performance.”

Slide 24 – t-SNE and Classifier Performance

“In practice, we wouldn’t run t-SNE in production, but this analysis tells us how to **use the autoencoder**.
If a new dataset falls in latent regions similar to Pediatric or CheXpert and shows high

reconstruction error, we should be very cautious about trusting a NIH-trained model there.

This is useful both for deployment decisions and for evaluating **third-party models** on local data.”

Slide 25 – t-SNE and Classifier Performance

“For AML students, the takeaway is:

Latent-space visualization is a powerful **debugging and explanation tool** in multi-domain settings.

It helps us see that NIH, Pediatric and CheXpert are fundamentally different distributions in the eyes of the model.

Combined with reconstruction error and performance metrics, it gives a coherent picture of where and why the model fails.”

Slide 26 – Phase 4 – DenseNet-121 Classifier

“In Phase 4 we introduce the **DenseNet-121 classifier**.

We use an ImageNet-pretrained DenseNet, adapt it to 1-channel 224×224 X-rays, and **fine-tune** it on NIH only.

We freeze early layers, fine-tune later blocks and a new classification head with a single sigmoid output.

Data augmentation and class weights handle the mild imbalance in NIH.

Important: the classifier is **completely separate** from the autoencoder; it uses raw images, not AE features.”

Slide 27 – Phase 4 – Classifier Performance

“On **NIH**, the classifier achieves high AUC and high **balanced accuracy** – this is the best-case, in-distribution performance.

On **Pediatric**, both sensitivity and specificity degrade, so balanced accuracy drops.

On **CheXpert**, AUC and raw accuracy might still look good because almost everything is abnormal, but **balanced accuracy** and specificity on the tiny normal class are much worse.

The datasets with the **highest reconstruction errors** are exactly the ones where the classifier performs worst.”

Slide 28 – Evaluation Metrics – AUC and Accuracy

“A quick reminder about metrics.

AUC measures ranking quality over all thresholds, which is useful but doesn't tell us how many actual errors we make at a chosen threshold.

Plain accuracy can be very misleading under class imbalance – especially in CheXpert, where ‘always abnormal’ gives high accuracy.

That's why we place more weight on **balanced accuracy** and per-class performance when we compare across domains.”

Slide 29 – Slide 29 (ROC / curves)

“Here we see ROC curves or similar visualizations.

On NIH, the ROC curve is clearly above the diagonal; on Pediatric and CheXpert it looks somewhat worse, but still not terrible if you only look at AUC.

However, when we inspect the confusion matrices and per-class metrics, we see that we're paying a price in either sensitivity or specificity in the external datasets.

So **ROC alone** doesn't fully reveal the impact of distribution shift.”

Slide 30 – Balanced Accuracy – Fair Evaluation

“**Balanced accuracy** solves part of this problem.

It is the average of **sensitivity** (true positive rate) and **specificity** (true negative rate).

This gives equal weight to both classes, independent of class prevalence.

A classifier that predicts ‘abnormal’ on all CheXpert images has high sensitivity but near-zero specificity, so its balanced accuracy is about 0.5.

In our results, balanced accuracy clearly shows a hierarchy: best on NIH, worse on Pediatric, worst on CheXpert.”

Slide 31 – Slide 31 (Confusion / threshold)

“This slide helps read the results in terms of **confusion matrices**.

On NIH, at a reasonable threshold, we have relatively few false positives and false negatives.

On Pediatric, errors increase, reflecting a mismatch between the learned decision boundary and children's data.

On CheXpert, preserving sensitivity leads to many false positives on the rare normal cases.

These patterns match the story from **reconstruction error and latent space**.”

Slide 32 – Phase 5 – Correlation Analysis

“In **Phase 5**, we explicitly link **unsupervised shift** to **supervised performance**. We compare average reconstruction error for each dataset with its balanced accuracy. We see a clear negative trend: NIH has lowest error and highest balanced accuracy; Pediatric is in the middle; CheXpert has highest error and lowest balanced accuracy. This suggests reconstruction error is indeed informative about expected performance.”

Slide 33 – Phase 5 – Correlation Analysis

“We can go finer by grouping images into **reconstruction-error quantiles** and computing performance within each bin. Low-error bins tend to have better classifier performance; high-error bins worse performance. Even though we only have a few datasets, the trend is consistent: **higher reconstruction error → lower balanced accuracy**. So the autoencoder is not just picking up arbitrary differences; its error is genuinely related to how reliable the classifier is.”

Slide 34 – Limitations, Future Work & Takeaways

“Our study has limitations. We focus only on **chest X-rays** and only on these particular datasets, one autoencoder architecture and one classifier family. We also work retrospectively; we don’t deploy this in a real hospital. So we can’t claim universal generalization yet. However, the patterns we see are strong enough to motivate more systematic, multi-center studies.”

Slide 35 – Limitations, Future Work & Takeaways

“For future work, we could test **other unsupervised models** – VAEs, diffusion models, self-supervised encoders – and compare their shift sensitivity. We could also build a more formal model that predicts expected balanced accuracy from reconstruction error and simple dataset statistics. And we could analyze performance **per pathology**, to see which diseases are more robust to domain change. Finally, integrating this into a real **MLOps pipeline** with automatic alerts would make it directly useful in practice.”

Slide 36 – Limitations, Future Work & Takeaways

“The high-level takeaways are:

1. Distribution shift is inevitable in real deployments, especially in medicine.
2. **Unsupervised tools** like autoencoders can provide valuable early warnings about domain mismatch.
3. In our experiments, reconstruction error correlates with performance degradation across datasets.
4. Choosing the right metrics, like **balanced accuracy**, is crucial to seeing the true impact of shift.
So combining unsupervised shift detection with careful evaluation can make medical AI safer.”

Slide 37 – Real-World Applications

“Finally, some real-world applications.

Hospitals could deploy an autoencoder alongside any classifier as a ‘**watchdog**’, monitoring reconstruction error over time.

If the error distribution drifts, they get an alert to reconsider thresholds, retrain, or pause the model.

Similarly, when evaluating external models, they can use reconstruction error on local data as part of their **accept/reject** decision.

In short, autoencoder-based shift monitoring can become a practical building block for safer and more robust deployment of medical imaging models.

Thank you for your attention.”

If you want, next step I can **regenerate the PDF** with this shortened script (one page per slide) so you just download and print/bring it.