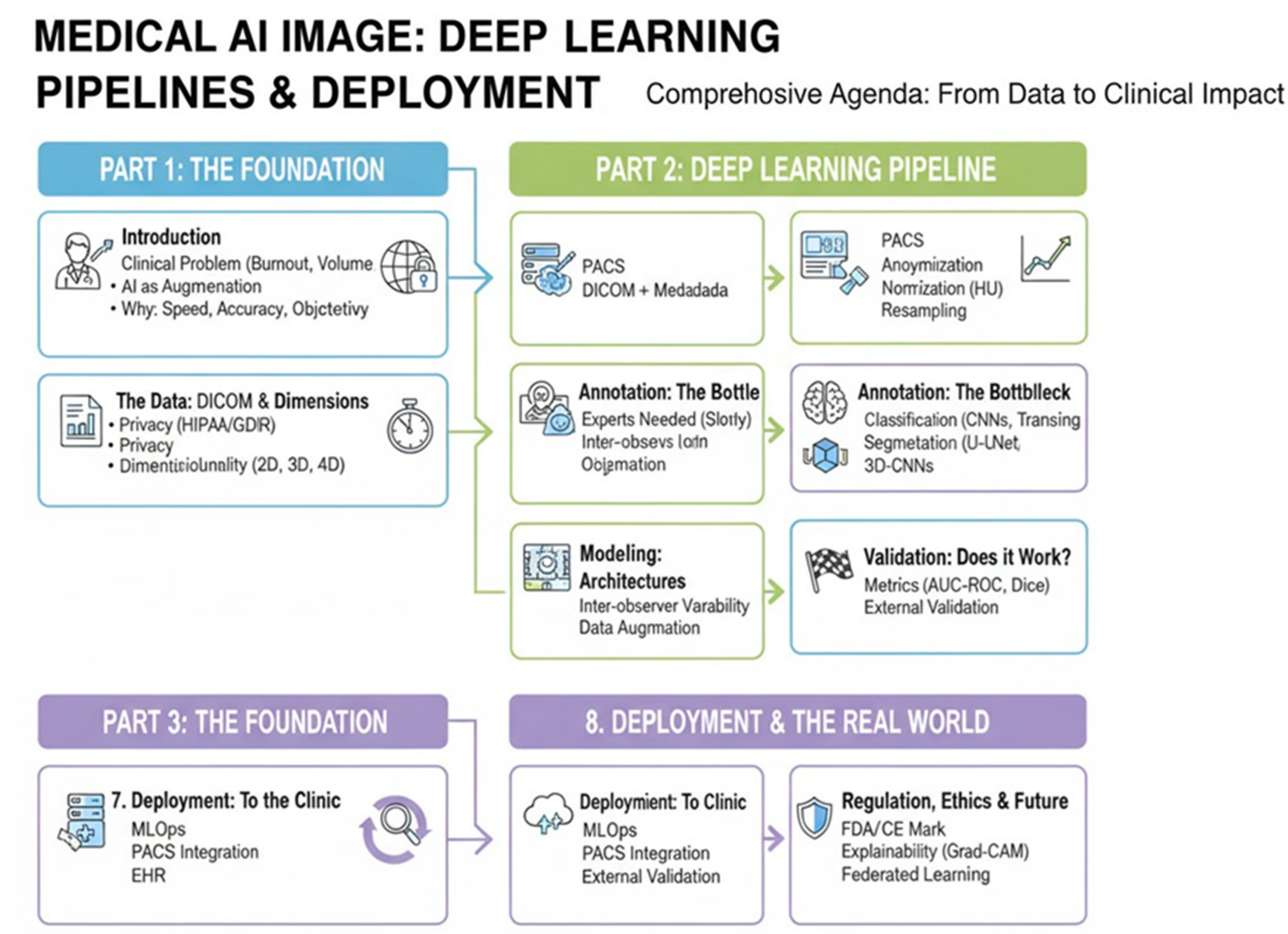
ราชวิทยาลัย
จุฬาภรณ์

Artificial Intelligence (AI) for Medical Innovation

จภวท 1262 ปัญญาประดิษฐ์สำหรับนวัตกรรมการแพทย์

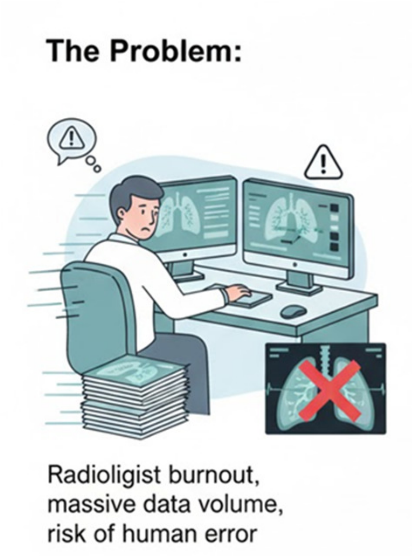
CHSC 1262 Artificial Intelligence (AI) for Medical Innovation

Agenda


MEDICAL AI IMAGE: DEEP LEARNING
PIPELINES & DEPLOYMENT
Comprehosive Agenda: From Data to Clinical Impact
PART 1: THE FOUNDATION
Introduction
Clinical Problem (Burnout, Volume
• AI as Augmenation
• Why: Speed, Accuracy, Objctetivy
The Data: DICOM & Dimensions
• Privacy (HIPAA/GDIR)
• Privacy
• Dimentiɪiolunality (2D, 3D, 4D)
PART 2: DEEP LEARNING PIPELINE
PACS
DICOM + Medadada
PACS
Anoymiization
Nomization (HU)
Resampling
Annotation: The Bottle
Experts Needed (Slotly)
Inter-obsevs lodn
Obigmation
Annotation: The Bottblleck
Classification (CNNs, Transing
Segmetation (U-UNet,
3D-CNNs
Modeling:
Architectures
Inter-observer Varability
Data Augmation
Validation: Does it Work?
Metrics (AUC-ROC, Dice)
External Validation
PART 3: THE FOUNDATION
7. Deployment: To the Clinic
MLOps
PACS Integration
EHR
8. DEPLOYMENT & THE REAL WORLD
Deployment: To Clinic
MLOps
PACS Integration
External Validation
Regulation, Ethics & Future
FDA/CE Mark
Explainability (Grad-CAM)
Federated Learning

The Clinical Problem & The AI Solution

The Problem: Why do we need this? Discuss radiologist burnout, the massive volume of scans (data explosion), the need for quantitative (not just qualitative) analysis, and the risk of human error (e.g., perceptual errors, missing subtle findings).

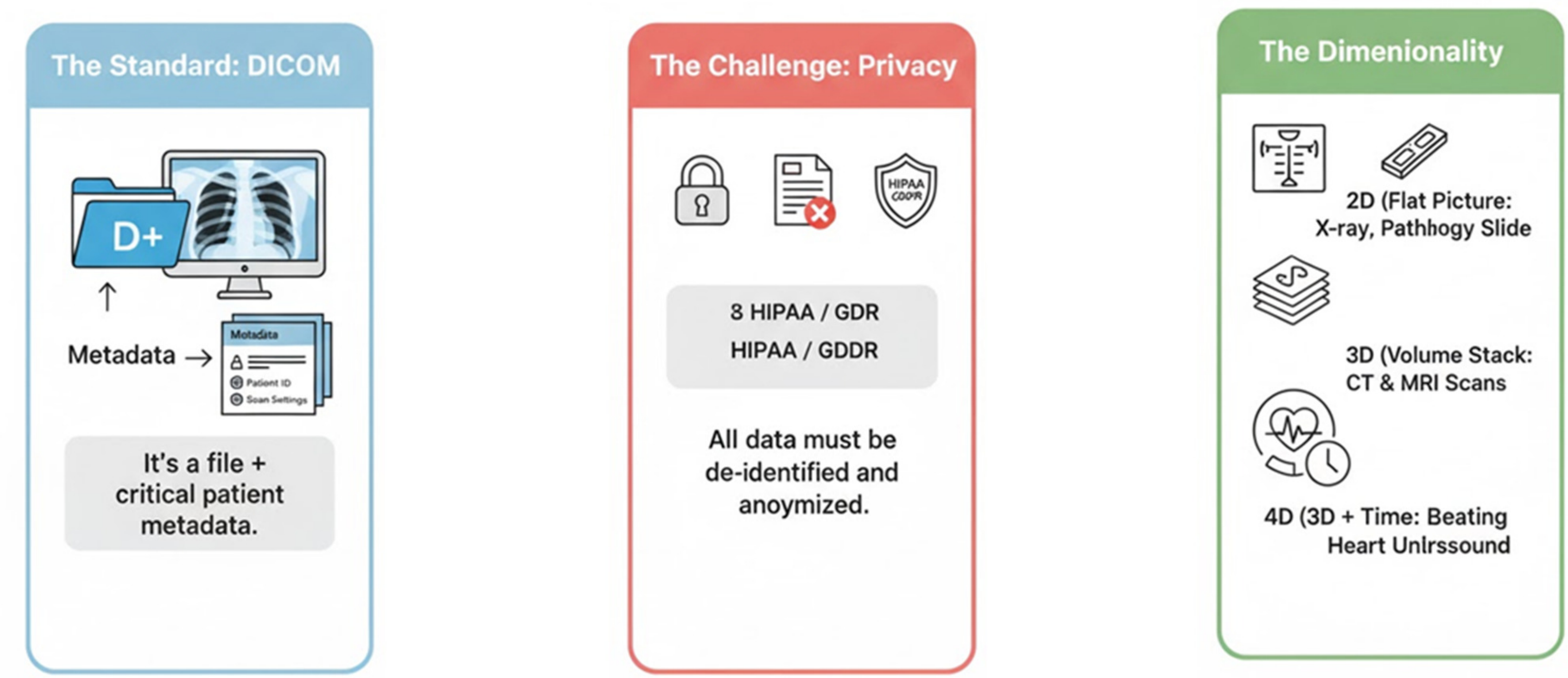


The Goal: Frame AI as an *augmenting* tool, not a replacement. Show the "doctor + AI" synergy.

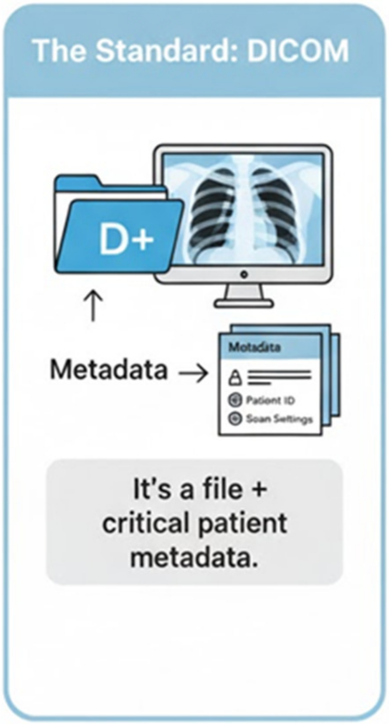
The Clinical Problem & The AI Solution

The "Why": Cover the key motivators: 1) **Speed/Efficiency**, 2) **Accuracy/Early Detection**, and 3) **Objectivity/Quantification**.

Medical Images are NOT JPEGs

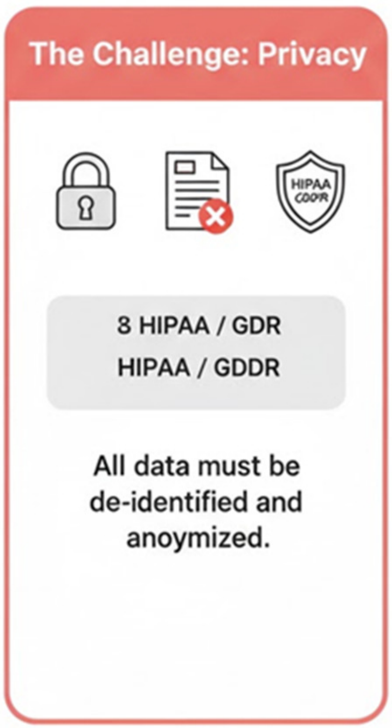


Medical Images are NOT JPEGs



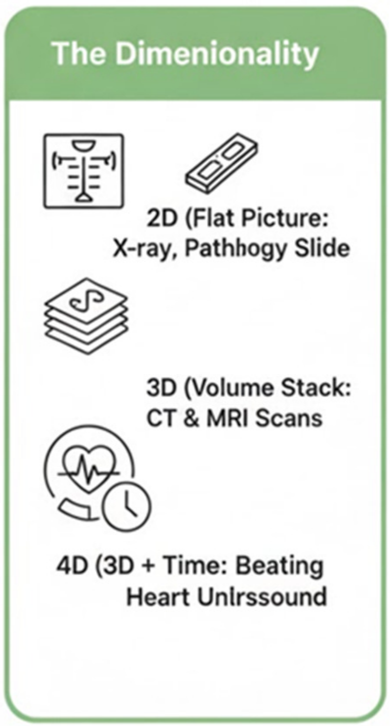
The Standard: DICOM. Explain what DICOM (Digital Imaging and Communications in Medicine) is. It's a file *and* a protocol. Emphasize that it's an image *plus* critical metadata (patient ID, age, scan parameters, etc.).

Medical Images are NOT JPEGs



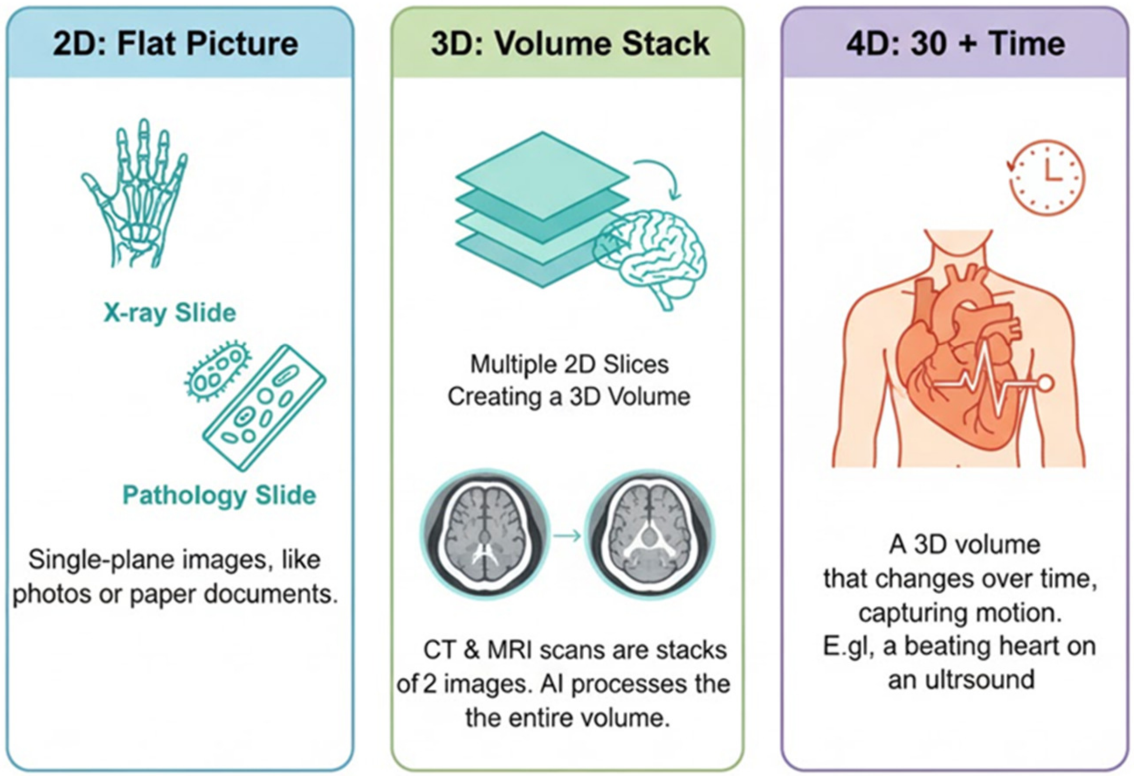
The Challenge: Privacy. This is the #1 rule. Introduce **HIPAA** (US) / **GDPR** (EU). All data *must* be de-identified and anonymized before any research begins.

Medical Images are NOT JPEGs

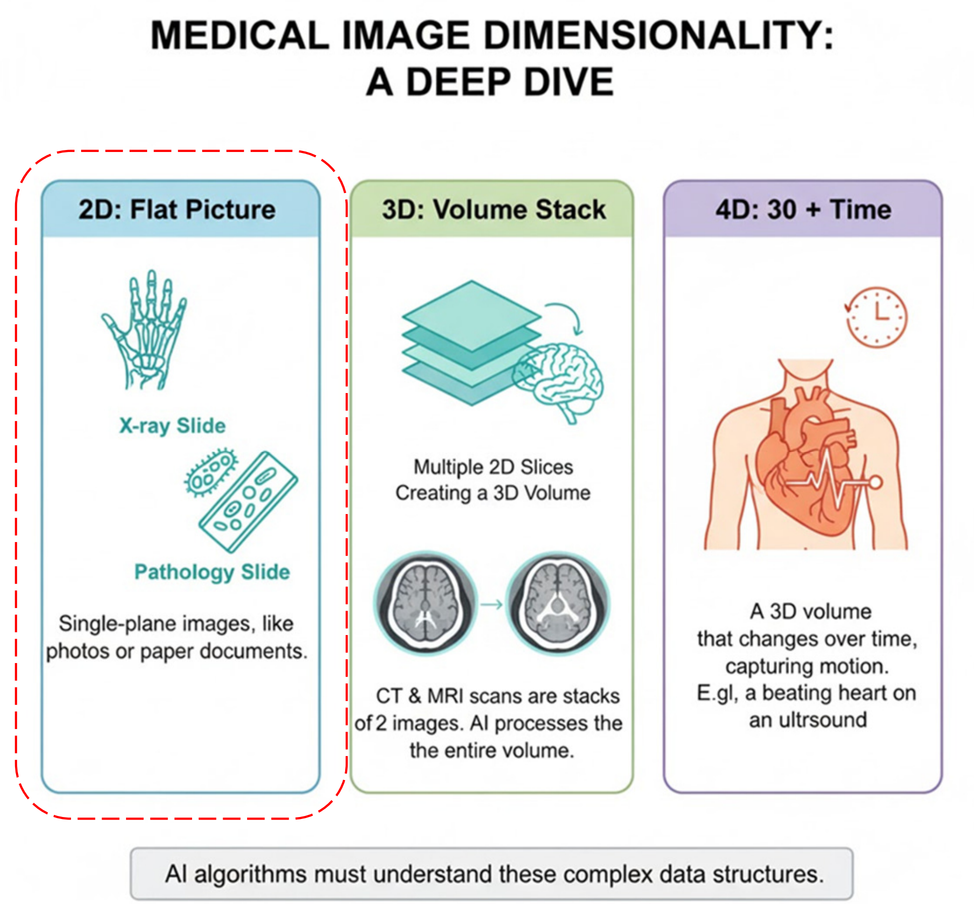


The Dimensionality: Explain the data types:
•**2D:** X-ray, Histopathology slides.
•**3D:** CT & MRI (a *stack* of 2D slices forming a 3D volume).
•**4D:** 3D + Time (e.g., a beating heart ultrasound).

MEDICAL IMAGE DIMENSIONALITY: A DEEP DIVE

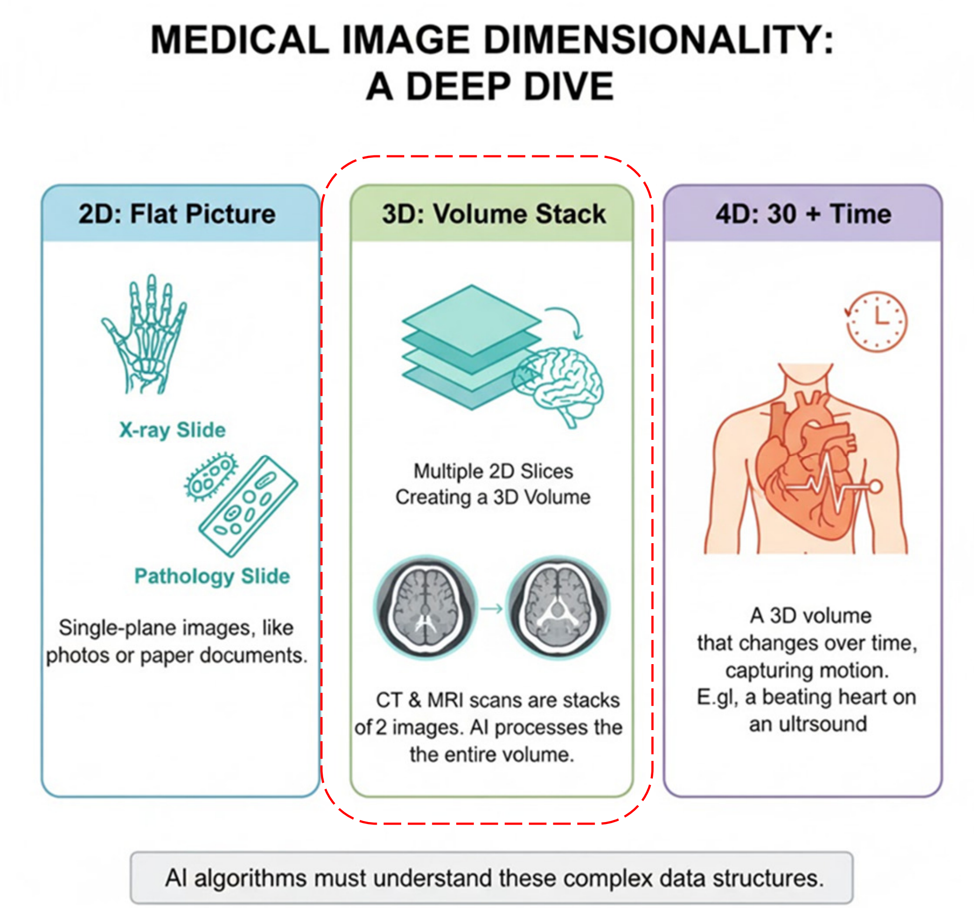


AI algorithms must understand these complex data structures.

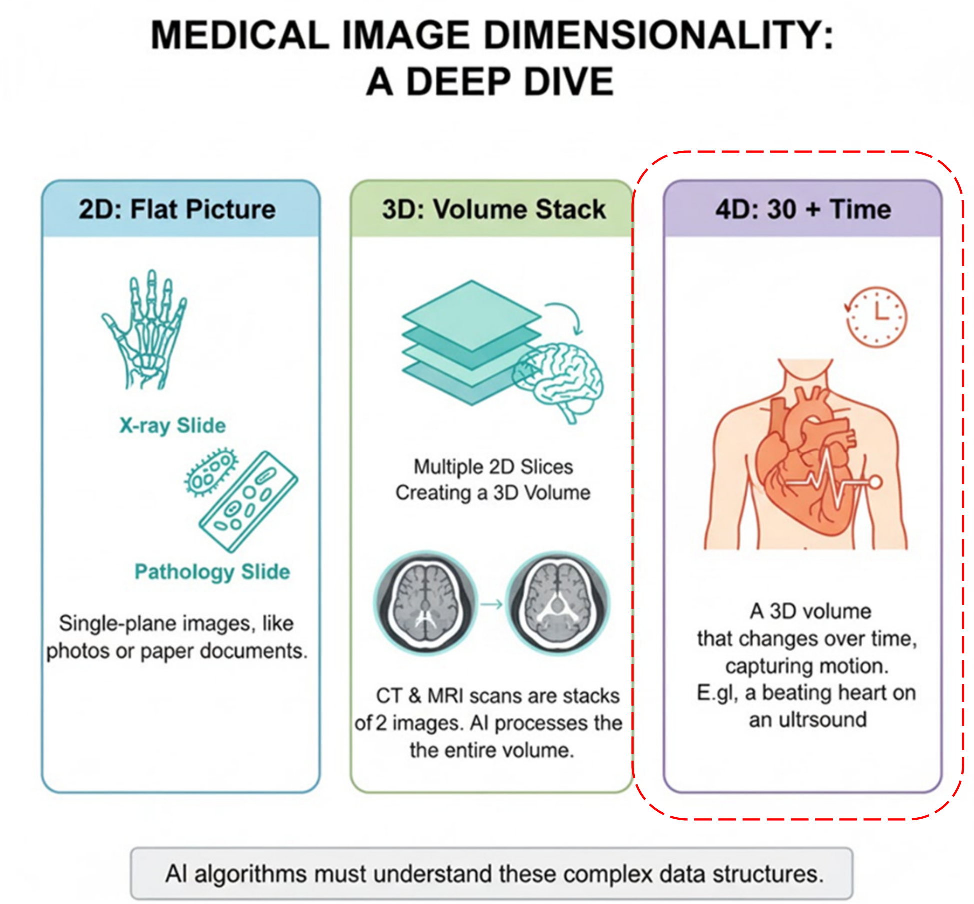



```python
# --- 1. 2D Data (Example: A single 2D X-Ray) ---
# This is a flat, grayscale image.
# It only has two dimensions: Height and Width.
# We'll simulate a small 256x256 pixel image.

xray_2d = np.random.rand(256, 256)

print(f"--- 1. 2D Grayscale X-Ray ---")
print(f"Dimensions (ndim): {xray_2d.ndim}")
print(f"Shape (Height, Width): {xray_2d.shape}")
print("-" * 30 + "\n")
```

```python
# --- (Special Case) 2D Color Image (Example: Pathology Slide) ---
# It's a "2D" image, but it has 3 color channels (Red, Green, Blue).
# This makes its data representation 3-dimensional.
# Shape: Height, Width, Channels

pathology_slide_2d_color = np.random.rand(256, 256, 3)

print(f"--- (Special Case) 2D Color Pathology Slide ---")
print(f"Dimensions (ndim): {pathology_slide_2d_color.ndim}")
print(f"Shape (Height, Width, Channels): {pathology_slide_2d_color.shape}")
print("-" * 30 + "\n")
```




```python
# --- 3. 3D Data (Example: A full CT Scan) ---
# This is a *volume* made of many 2D slices stacked together.
# It has three dimensions: Slices, Height, and Width.
# Let's simulate a scan with 90 slices, each 256x256 pixels.

ct_scan_3d = np.random.rand(90, 256, 256)

print(f"--- 3. 3D Volumetric CT Scan ---")
print(f"Dimensions (ndim): {ct_scan_3d.ndim}")
print(f"Shape (Slices, Height, Width): {ct_scan_3d.shape}")
print(f"You can access a single 2D slice with: ct_scan_3d[0].shape -> {ct_scan_3d[
print("-" * 30 + "\n")
```




```python
# --- 4. 4D Data (Example: A 3D Cardiac MRI over Time) ---
# This is a 3D volume that *changes over time*.
# It has four dimensions: Time, Slices, Height, and Width.
# Let's simulate our 90-slice scan over 20 time-steps (e.g., 20 frames of a heartb

cardiac_mri_4d = np.random.rand(20, 90, 256, 256)

print(f"--- 4. 4D Cardiac (Time-Series) MRI ---")
print(f"Dimensions (ndim): {cardiac_mri_4d.ndim}")
print(f"Shape (Time, Slices, Height, Width): {cardiac_mri_4d.shape}")
print(f"Access one moment in time (a 3D volume): cardiac_mri_4d[0].shape -> {cardi
print(f"Access one slice at one moment: cardiac_mri_4d[0, 0].shape -> {cardiac_mri_
print("-" * 30 + "\n")
```

The Data : DICOM & Dimensions

What a DICOM File Stores

A single DICOM (.dcm) file is like a highly-structured ZIP file.
It contains two completely different types of information bundled together

Metadata (The "Header" or "Tags"):

This is all the text-based information about the scan. It answers who, what, where, and how.Patient Info: Patient's Name, Patient ID, Age, Sex, Date of Birth. (This is the data that must be anonymized).Scan Info: Modality (e.g., 'CT', 'MRI', 'X-RAY'), Scan Settings (e.g., radiation dose, magnetic field strength), Body Part Examined (e.g., 'CHEST', 'BRAIN').Identifiers: Unique IDs (UIDs) for the patient, the study, and the specific image instance.

The Data : DICOM & Dimensions

Pixel Data (The "Image"):This is the actual image itself, stored as a raw grid of numbers.These numbers represent pixel intensity (e.g., Hounsfield Units in a CT scan).This data is often compressed (using formats like JPEG 2000 or RLE) to save space. It is not a simple JPG or PNG

The Data : DICOM & Dimensions

pydicom/**pydicom**

Read, modify and write DICOM files with python code

The pydicom Library (The "DICOM Parser")You use the function pydicom.dcmread("your_file.dcm").
This function reads the entire DICOM file from your disk.It creates a special

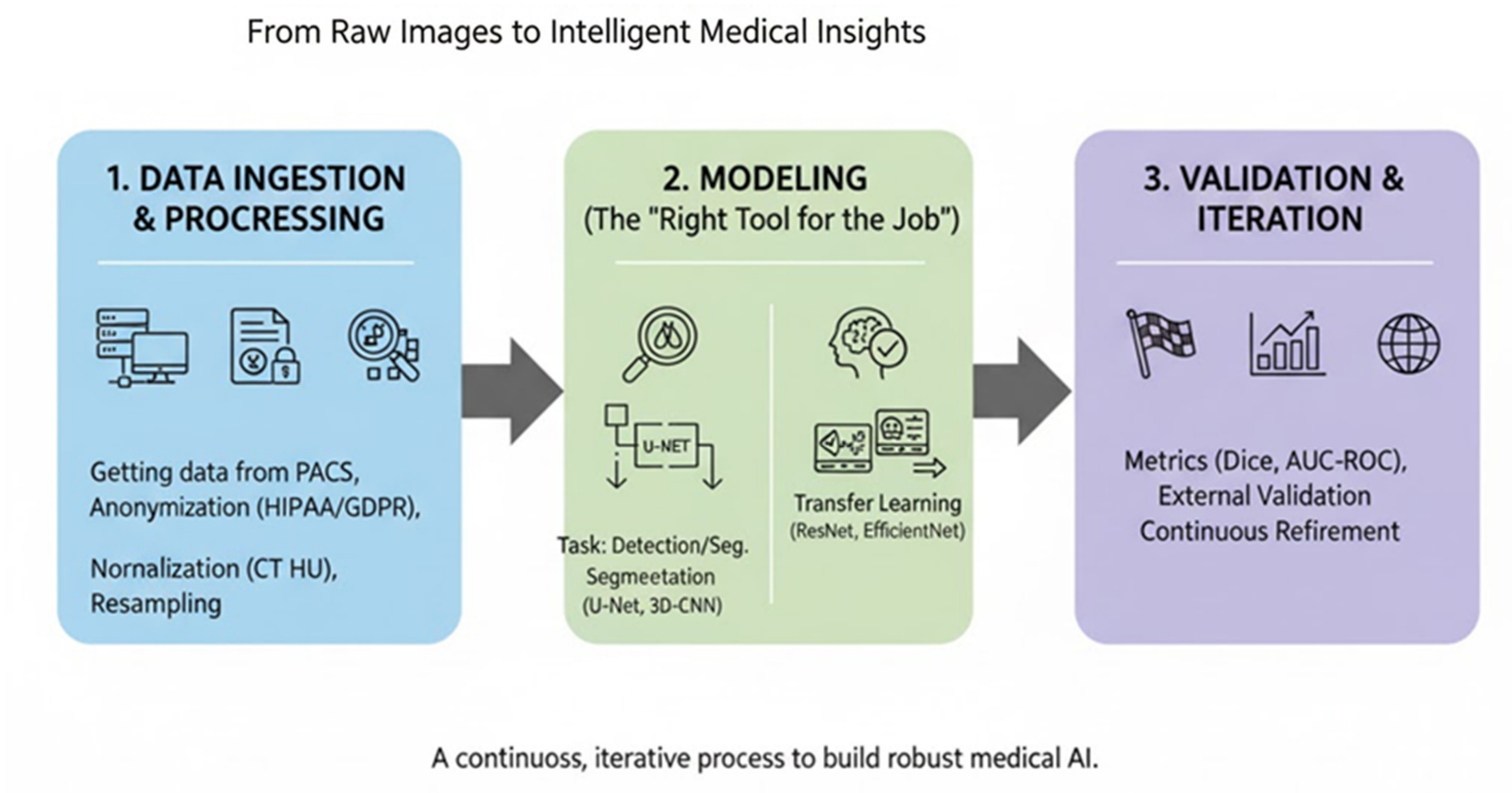
Python object called a FileDataset.What it stores: This FileDataset object holds all the METADATA. You can access it like a Python dictionary.

The numpy Library (The "Math Engine")The pydicom library by itself doesn't decompress the image.

It holds the raw, compressed pixel data.The magic happens when you type .pixel_array.When you access this attribute, pydicom performs its second job: it finds the compressed Pixel Data, decompresses it, and converts it into a NumPy array (ndarray).

What it stores: The numpy library holds the final, uncompressed PIXEL DATA in a grid of numbers.

The Deep Learning Pipeline



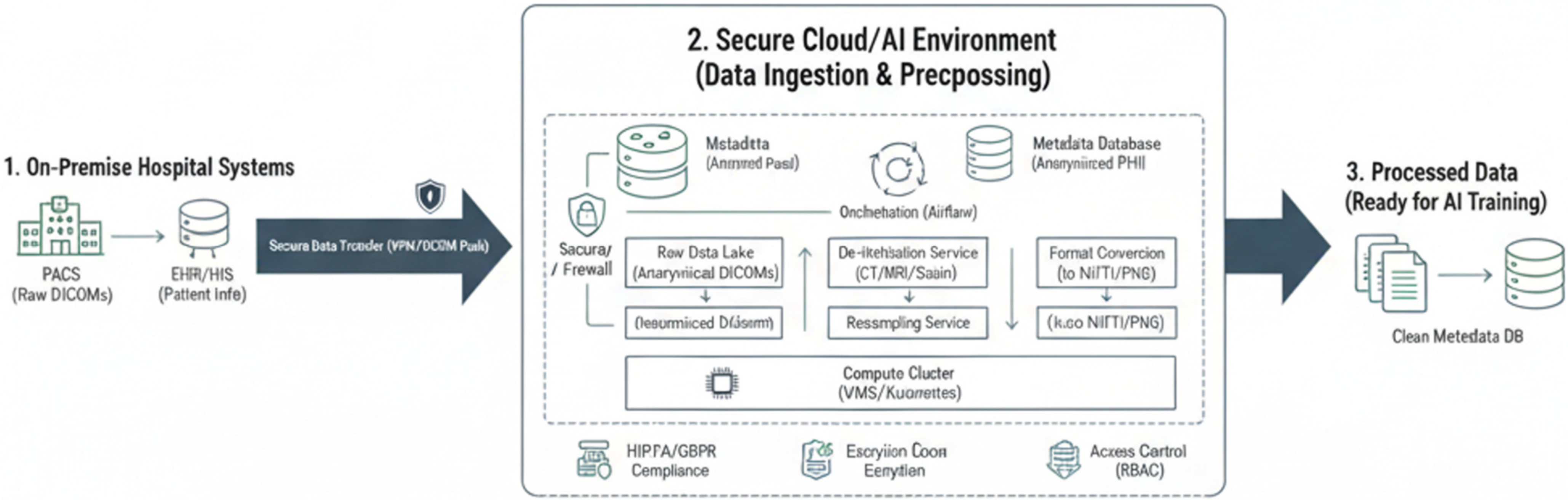
Data Ingestion & Preprocessing

The Raw Data: How do you get data from the hospital? Introduce the **PACS (Picture Archiving and Communication System)**. This is the hospital's image server.

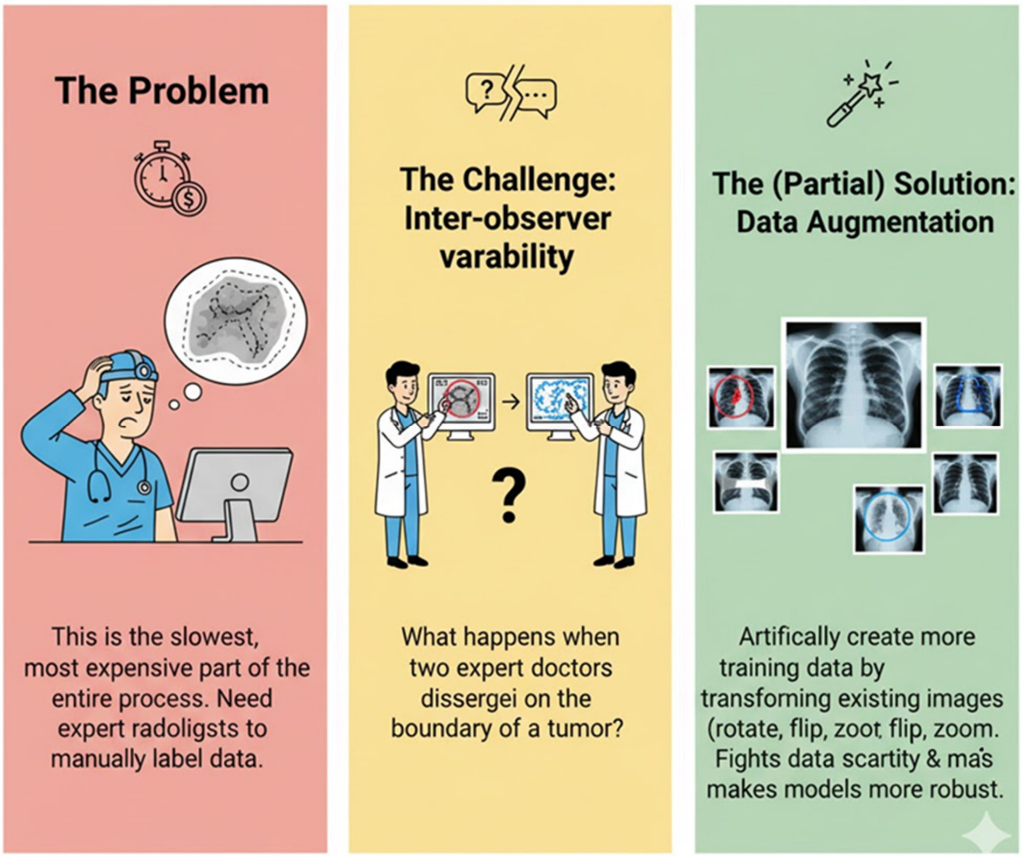
The "Cleaning" (Preprocessing): This is the most critical step.

•**Anonymization:** Stripping all patient metadata from DICOM headers.
•**Normalization:** Crucial for CT (converting to **Hounsfield Units (HU)**), MRI (bias field correction), and slides (stain normalization).
•**Resampling:** Ensuring all 3D volumes have the same voxel spacing.

Data Ingestion & Preprocessing



The Annotation Bottleneck

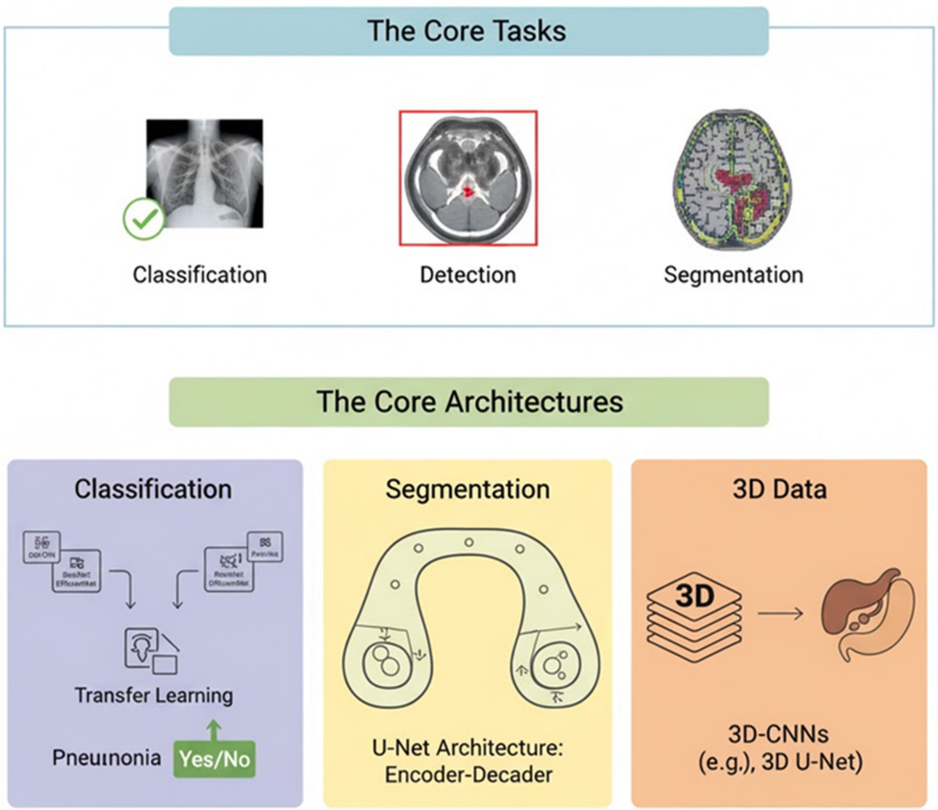


The Problem: This is the slowest, most expensive part of the entire process. You need expert radiologists or pathologists to manually label data.

The Challenge: Inter-observer variability. What happens when two expert doctors disagree on the boundary of a tumor?

The (Partial) Solution: Discuss strategies like **data augmentation** (rotating, flipping, zooming) to fight data scarcity, and "smart labeling" techniques (e.g., weak supervision).

Modeling (The "Right Tool for the Job")



Matching the AI model to the specific clinical question.

The Core Tasks: Briefly recap the 3 main tasks (Classification, Detection, Segmentation).

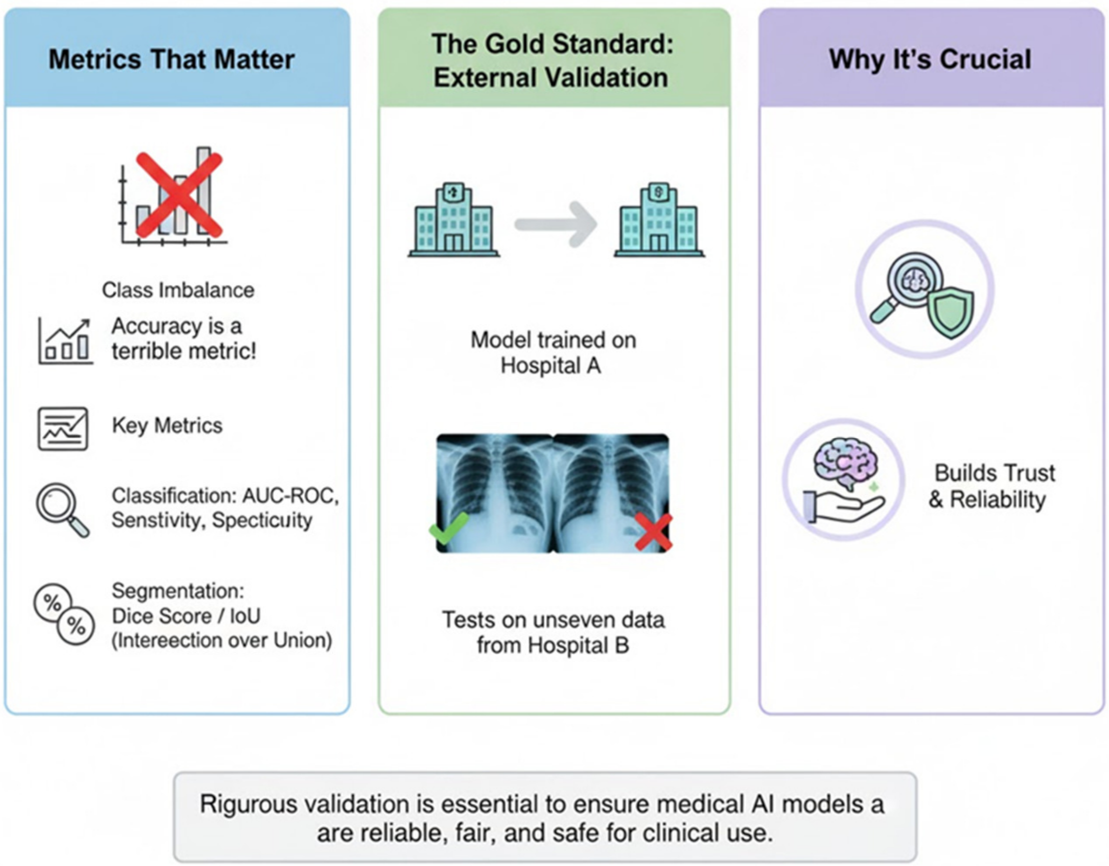
The Core Architectures:

•**Classification (e.g., "Pneumonia: Yes/No"):** Standard CNNs (like ResNet, EfficientNet) + **Transfer Learning**. Emphasize that *nobody* trains from scratch.

•**Segmentation (e.g., "Outline the tumor"):** This is the star of medical AI. You *must* introduce the **U-Net** architecture and explain *why* its encoder-decoder structure is perfect for this.

•**3D Data:** Explain that we can't just use 2D models. Introduce **3D-CNNs** (e.g., 3D U-Net) that process the entire 3D volume at once.

Validation (How We Know It *Actually* Works)



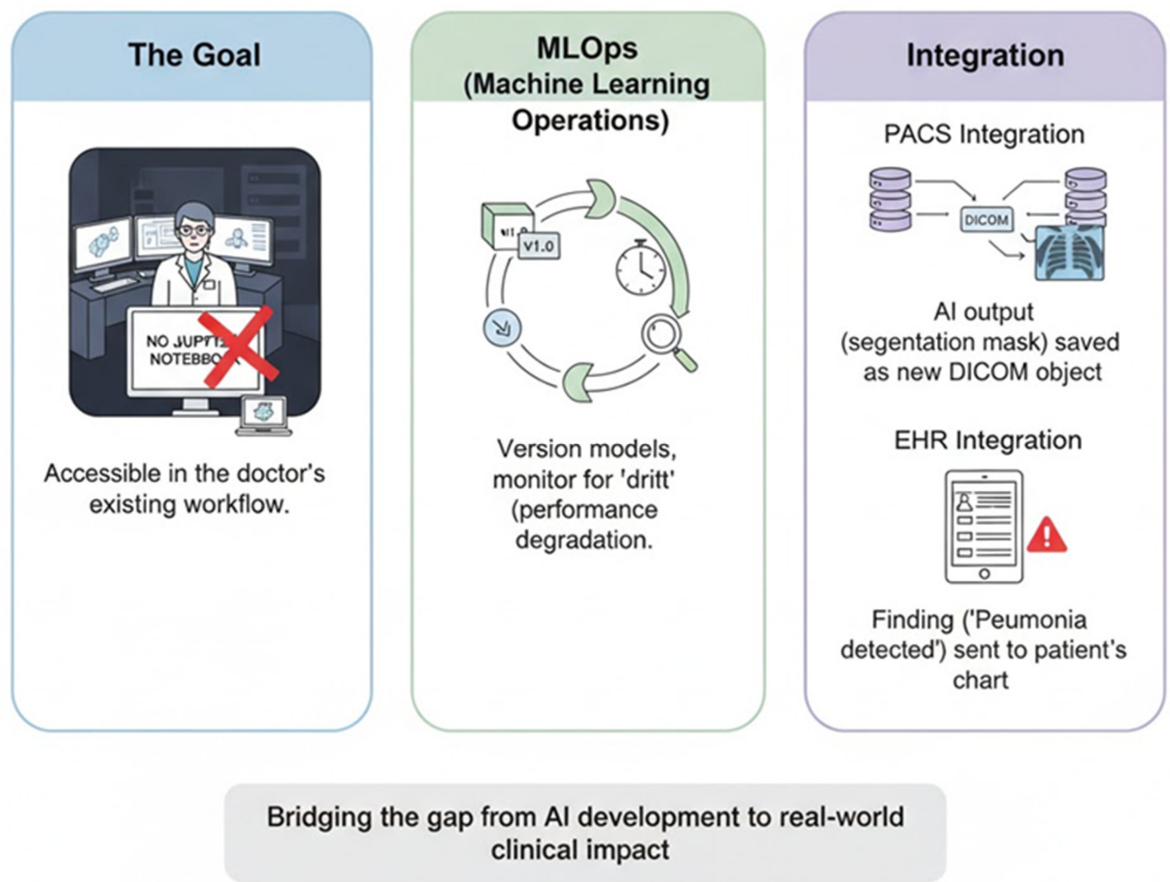
Metrics That Matter: Explain why "accuracy" is a *terrible* metric for imbalanced medical data (e.g., a model that always guesses "no tumor" is 99% accurate but 100% useless).

Key Metrics:
•**Classification: AUC-ROC** (Area Under the Curve), Sensitivity, Specificity.
•**Segmentation: Dice Score** / IoU (Intersection over Union).

The Gold Standard: External Validation. A model trained on data from Hospital A *must* be tested on data from Hospital B. This proves it has generalized and not just memorized the scanner noise from Hospital A.

Deployment & The Real World

Deployment: From Notebook to the Clinic



The Goal: The model must be accessible to the doctor *in their existing workflow*. A doctor will not open a Jupyter notebook.

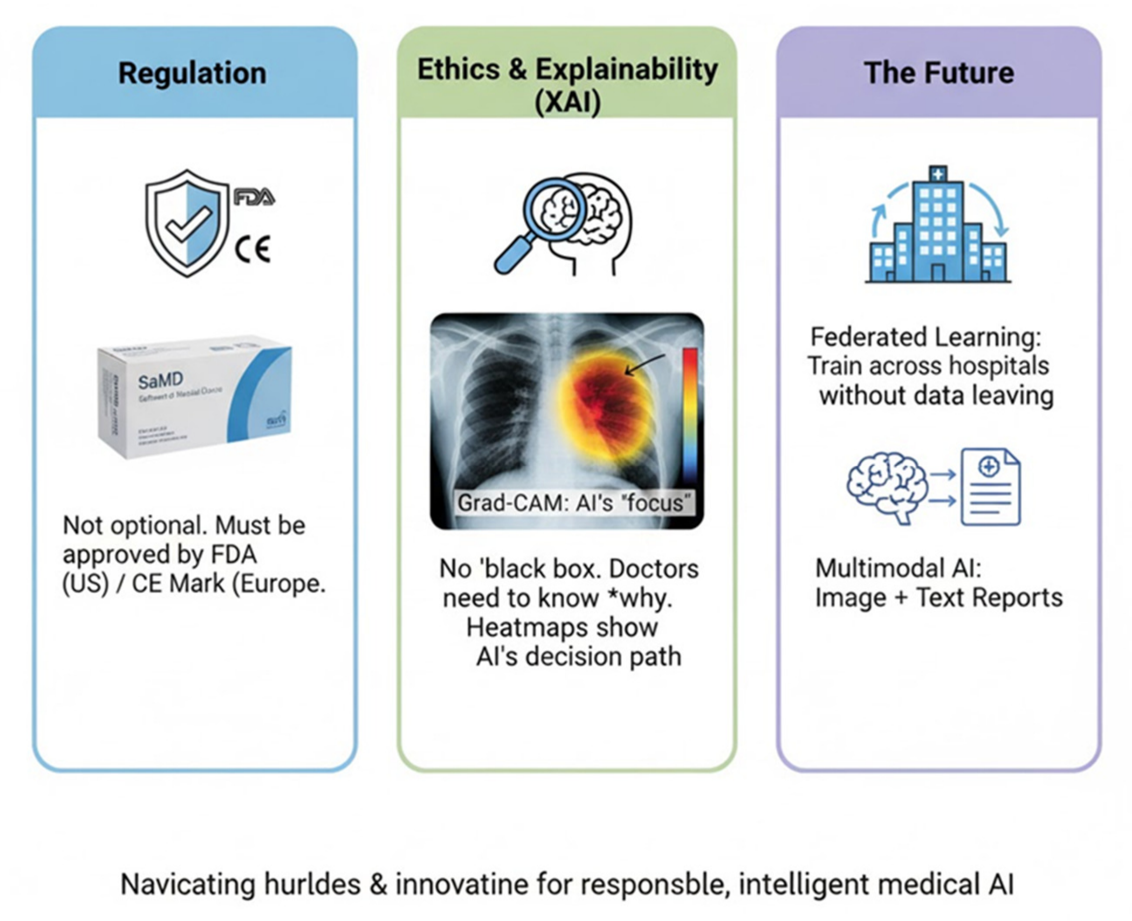
MLOps (Machine Learning Operations): Briefly explain this. How do you version models? How do you monitor for model "drift" (when its performance degrades over time)?

Integration: The model's output must be sent back to the hospital.

•**PACS Integration:** The AI's output (e.g., a segmentation mask) is saved as a *new* DICOM object and sent back to the PACS.

•**EHR (Electronic Health Record):** The finding ("Pneumonia detected") is sent to the patient's chart.

Regulation, Ethics & The Future



Regulation: This is not optional. A medical AI tool is a **Software as a Medical Device (SaMD)**. It *must* be approved by regulatory bodies like the **FDA** (US) or get a **CE Mark** (Europe). This is a long, expensive process.

Ethics & Explainability (XAI): We can't use a "black box." A doctor needs to know *why* the AI made a decision. Show examples of **heatmaps (Grad-CAM)** that highlight what the AI was "looking at."

The Future: Briefly touch on **Federated Learning** (training a model on data from multiple hospitals *without* the data ever leaving the hospital) and **Multimodal AI** (models that read the image *and* the radiologist's text report).

Questions?