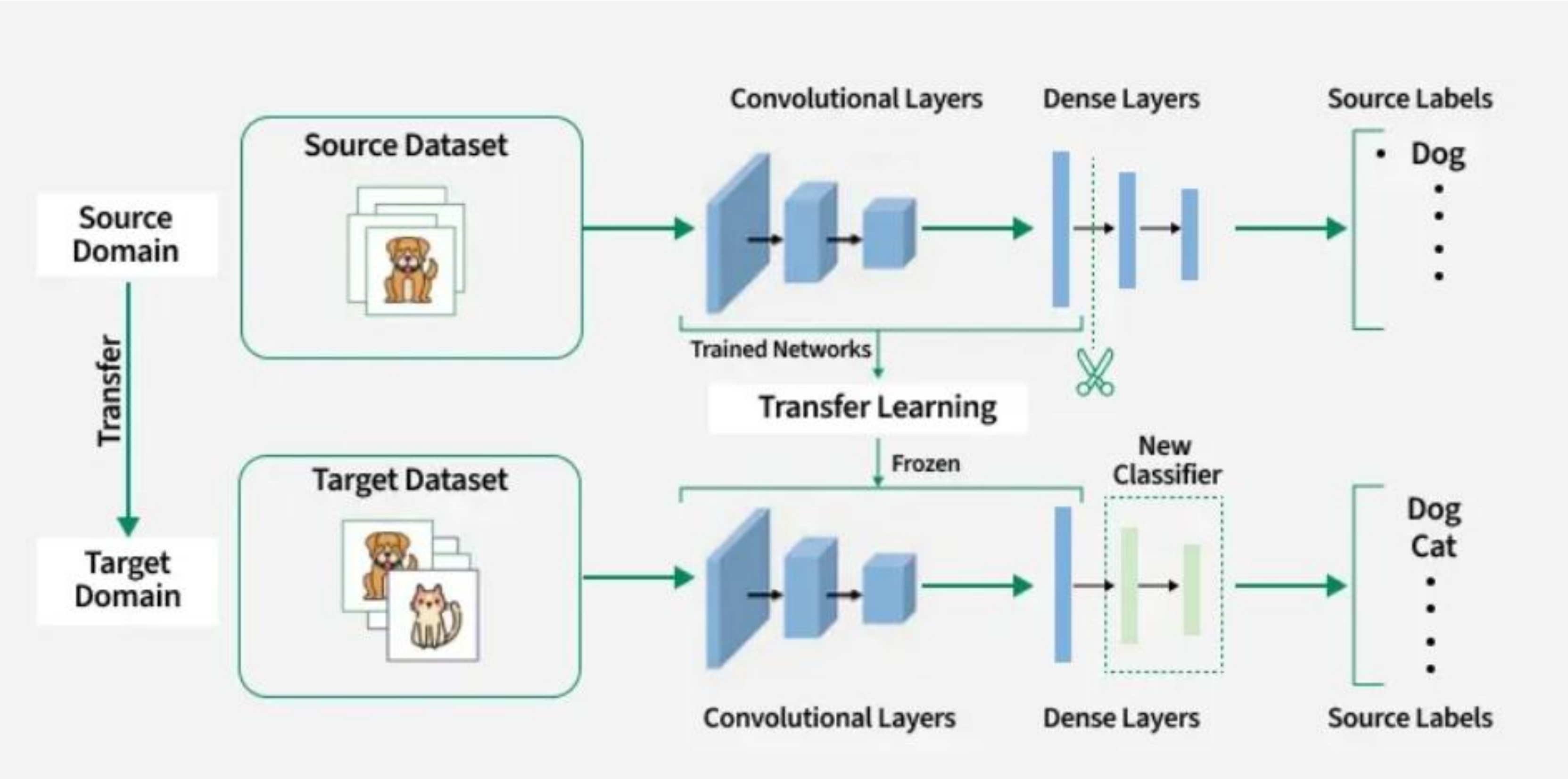
Transfer Learning and Model Explainability

Transfer Learning

- Transfer learning is a machine learning technique where a model trained on one task is repurposed as the foundation for a second task.
- This approach is beneficial when the second task is related to the first or when data for the second task is limited.


Source Domain
Transfer
Target Domain
Source Dataset
Target Dataset
Convolutional Layers
Dense Layers
Source Labels
Dog
Trained Networks
Transfer Learning
Frozen
New Classifier
Dog
Cat
Convolutional Layers
Dense Layers
Source Labels

- ❑ Using learned features from the initial task, the model can adapt more efficiently to the new task, accelerating learning and improving performance.

- ❑ Transfer learning also reduces the risk of overfitting, as the model already incorporates generalizable features useful for the second task.

Working of Transfer Learning

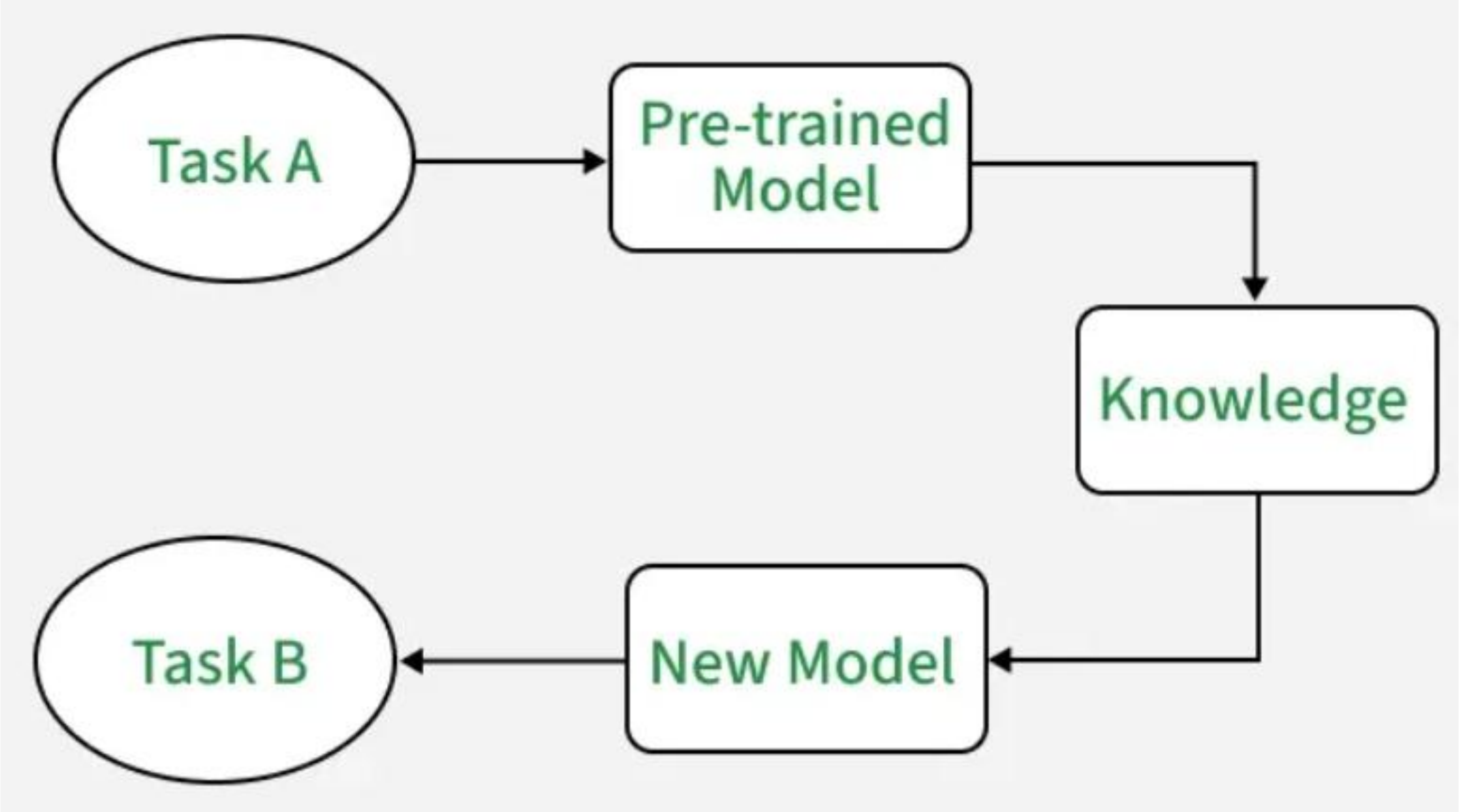
Transfer learning involves a structured process to use existing knowledge from a pre-trained model for new tasks:

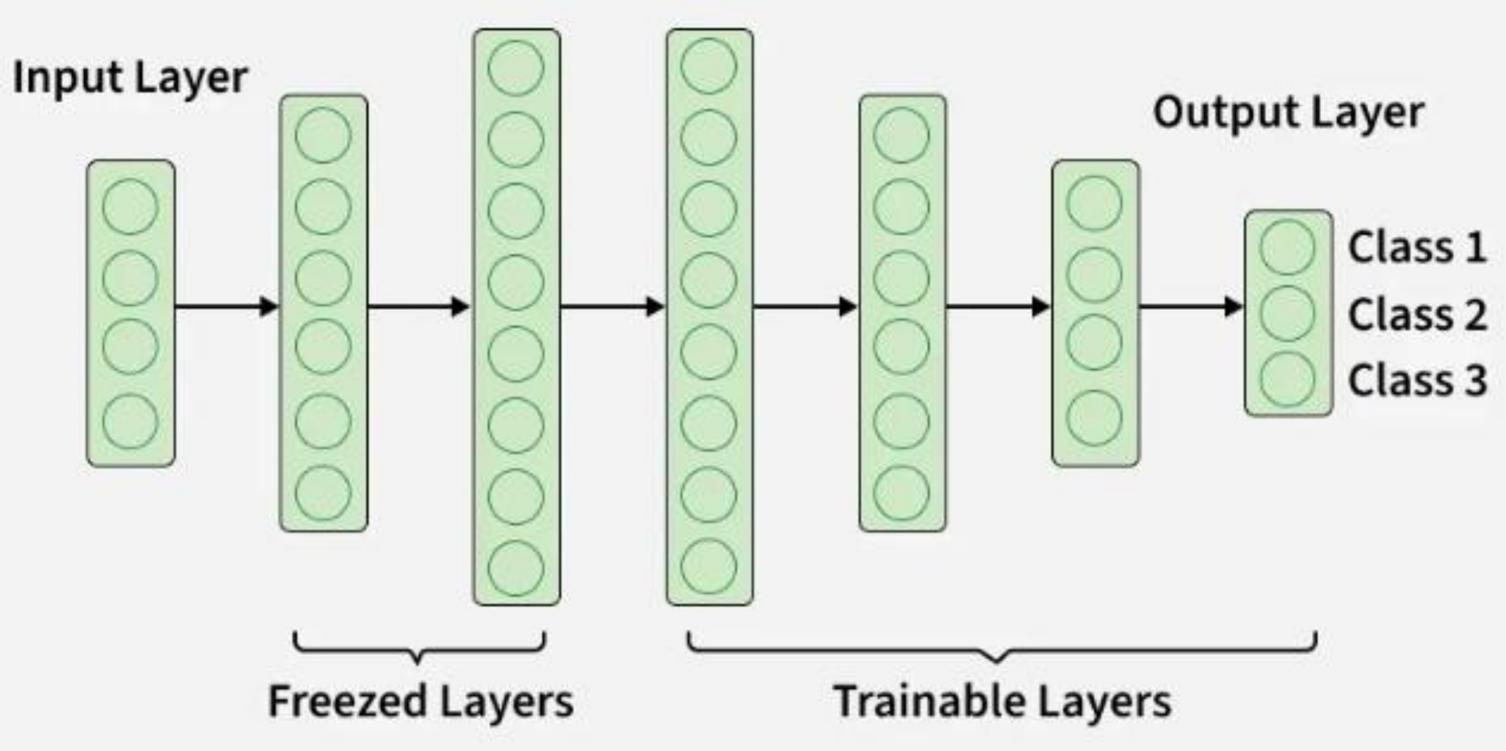
- ❑ **Pre-trained Model:** Start with a model already trained on a large dataset for a specific task. This pre-trained model has learned general features and patterns that are relevant across related tasks.

- ❑ **Base Model:** This pre-trained model, known as the base model, includes layers that have processed data to learn hierarchical representations, capturing low-level to complex features.

- **Transfer Layers:** Identify layers within the base model that hold generic information applicable to both the original and new tasks. These layers often near the top of the network capture broad, reusable features.

- **Fine-tuning:** Fine-tune these selected layers with data from the new task. This process helps retain the pre-trained knowledge while adjusting parameters to meet the specific requirements of the new task, improving accuracy and adaptability.


Task A
Pre-trained Model
Knowledge
New Model
Task B


Input Layer
Output Layer
Class 1
Class 2
Class 3
Freezed Layers
Trainable Layers

How to Decide Which Layers to Freeze or Train?

The extent to which you freeze or fine-tune layers depends on the similarity and size of your target dataset:

- ❑ **Small, Similar Dataset**: For smaller datasets that resemble the original dataset, you freeze most layers and only fine-tune the last one or two layers to prevent overfitting.

- ❑ **Large, Similar Dataset**: With large, similar datasets you can unfreeze more layers allowing the model to adapt while retaining learned features from the base model.

❑ **Small, Different Dataset**: For smaller, dissimilar datasets, fine-tuning layers closer to the input layer helps the model learn task-specific features from scratch.

❑ **Large, Different Dataset**: In this case, fine-tuning the entire model helps the model adapt to the new task while using the broad knowledge from the pre-trained model.

Domain Adaptation

- Domain Adaptation is a specialized area within transfer learning that addresses the challenge of training a machine learning model on one data distribution (the source domain) and applying it to a related but different data distribution (the target domain).

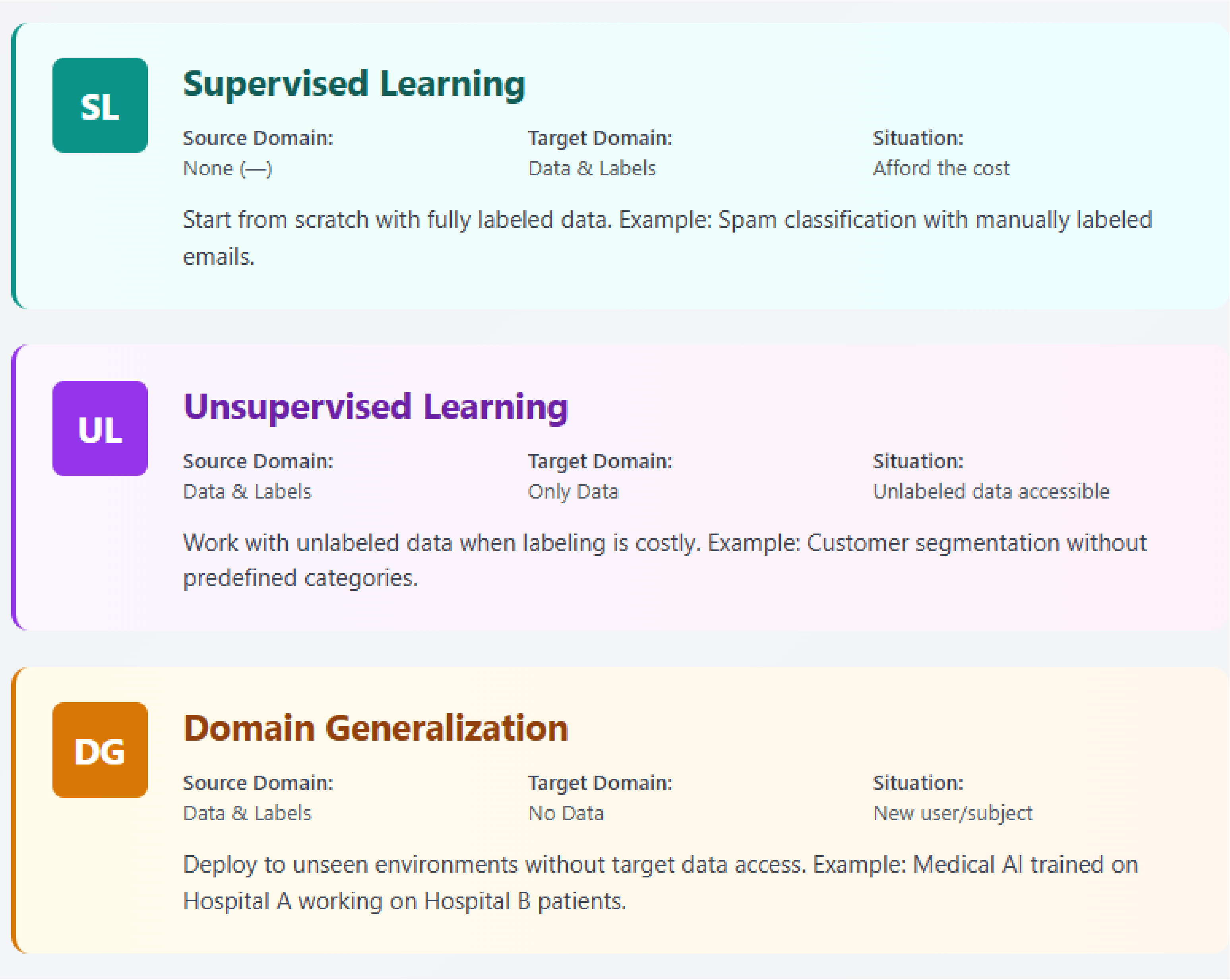
Standard machine learning assumes that training and test data are drawn from the same distribution. However, in real-world scenarios:

- ❑ **Training data** (source domain) may come from one environment
- ❑ **Test data** (target domain) may come from a different but related environment

Example: A spam filter trained on one user's emails (source domain) must adapt to another user's significantly different email patterns (target domain).

Why Domain Adaptation ?

- ❑ **Data Collection is Expensive**: Labeling data for every new domain is costly and time-consuming
- ❑ **Distribution Shift**: Real-world data often differs from training conditions
- ❑ **Computational Efficiency**: Retraining models from scratch is computationally expensive
- ❑ **Scalability**: Models need to generalize across diverse environments



SL

Supervised Learning

Source Domain: None (—)

Target Domain: Data & Labels

Situation: Afford the cost

Start from scratch with fully labeled data. Example: Spam classification with manually labeled emails.

UL

Unsupervised Learning

Source Domain: Data & Labels

Target Domain: Only Data

Situation: Unlabeled data accessible

Work with unlabeled data when labeling is costly. Example: Customer segmentation without predefined categories.

DG

Domain Generalization

Source Domain: Data & Labels

Target Domain: No Data

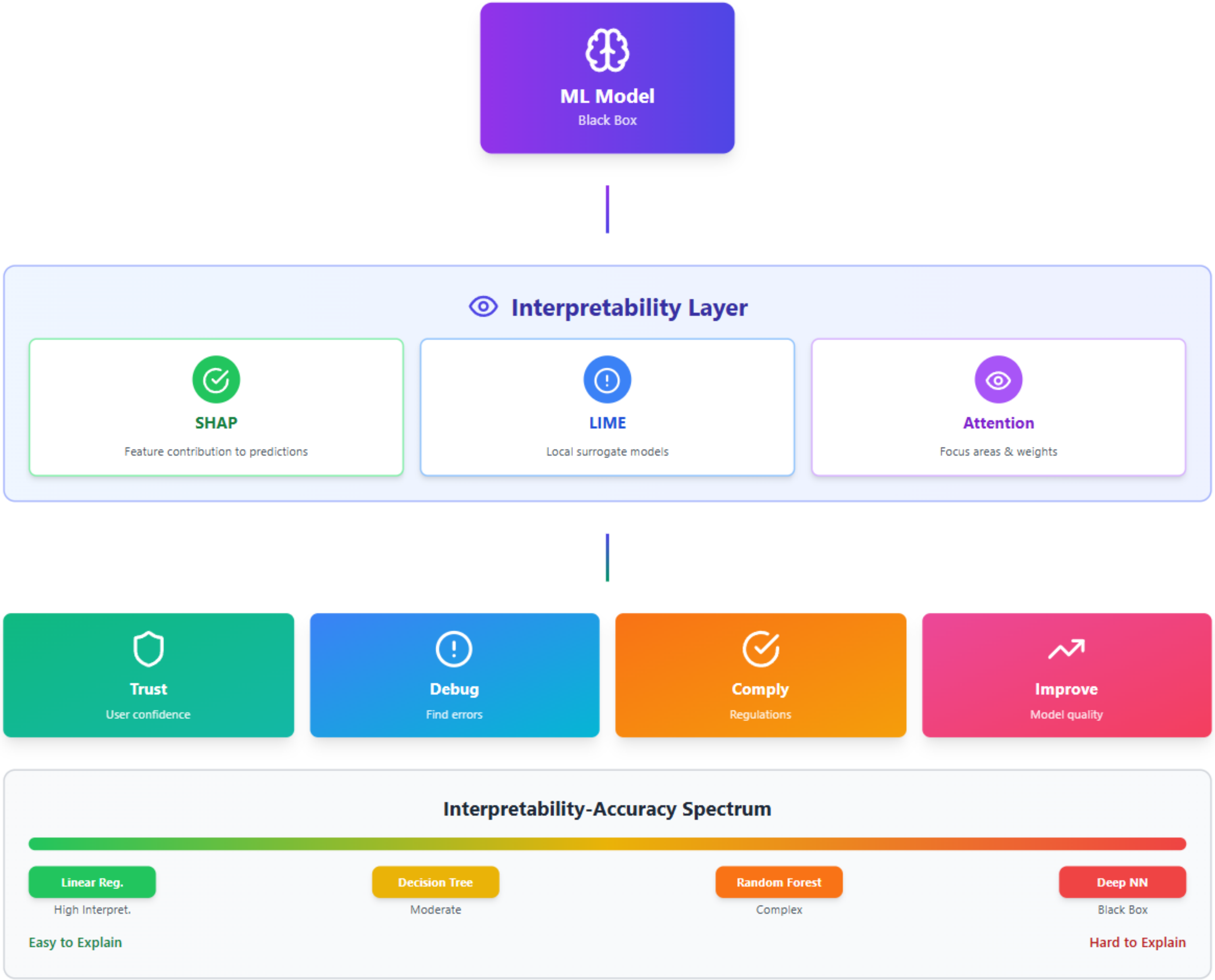
Situation: New user/subject

Deploy to unseen environments without target data access. Example: Medical AI trained on Hospital A working on Hospital B patients.

Model Interpretability

- ❑ Model interpretability refers to the ability to understand and explain how a machine learning or deep learning model makes its predictions or decisions.
- ❑ In traditional machine learning models, such as decision trees or linear regression, understanding the model's behavior is relatively straightforward due to their transparency.
- ❑ However, deep learning models, especially neural networks, operate as complex, multi-layered black boxes, making interpretability a challenging task.

Model Interpretability Framework



ML Model

Black Box

Interpretability Layer

SHAP

Feature contribution to predictions

LIME

Local surrogate models

Attention

Focus areas & weights

Trust

User confidence

Debug

Find errors

Comply

Regulations

Improve

Model quality

Interpretability-Accuracy Spectrum

Linear Reg.

High Interpret.

Decision Tree

Moderate

Random Forest

Complex

Deep NN

Black Box

Easy to Explain

Hard to Explain

The need for model interpretability arises for several reasons:

- **Trust and Transparency**: When deploying deep learning models in critical applications such as healthcare, finance, or law, stakeholders need to trust the model’s decisions. Interpretability provides insights into why the model made a specific decision, increasing transparency.

- **Bias Detection and Mitigation**: Interpretability allows researchers and developers to detect and correct biases in the model that could lead to unfair or incorrect predictions. Without it, models may perpetuate harmful biases that exist in training data.

❑ **Model Debugging**: Interpretability helps in debugging models by providing insights into which features or aspects are influencing the predictions. This is particularly useful in improving model performance and correcting errors.

❑ **Compliance with Regulations**: In some industries, there are legal requirements to explain decisions made by AI models. For example, the General Data Protection Regulation (GDPR) mandates that individuals have a right to explanation when decisions are made by automated systems.

Model Explainability

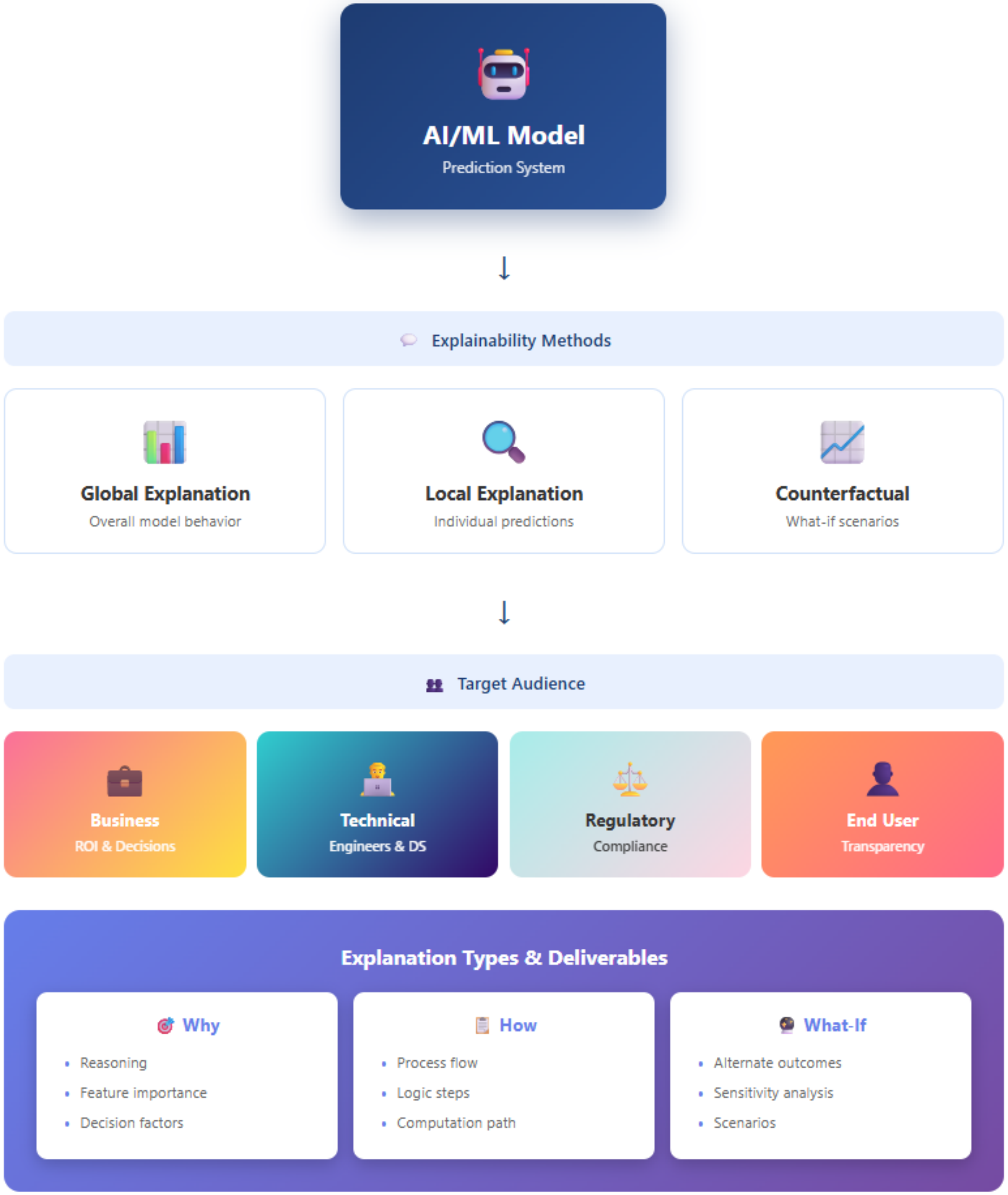
- Model explainability is the ability to understand how a machine learning model arrives at its decisions, transforming a "black box" into a transparent system.
- This process uses various techniques to explain the model's inner workings and the reasoning behind its outputs, making it crucial for building trust, ensuring fairness, and meeting regulatory compliance in sensitive areas like finance and healthcare.

- **Transparency:** Explainability makes the model's decision-making process understandable to humans, from the input data to the final output.

- **Explainable AI (XAI):** This is the formal term for the set of tools and techniques used to achieve model explainability across artificial intelligence.

- **Beyond the output:** It answers questions about *why* a model made a specific prediction, classification, or recommendation, not just *what* the prediction was.

Model Explainability Framework



AI/ML Model

Prediction System

↓

Explainability Methods

Global Explanation
Overall model behavior

Local Explanation
Individual predictions

Counterfactual
What-if scenarios

↓

Target Audience

Business
ROI & Decisions

Technical
Engineers & DS

Regulatory
Compliance

End User
Transparency

Explanation Types & Deliverables

Why

- Reasoning
- Feature importance
- Decision factors

How

- Process flow
- Logic steps
- Computation path

What-If

- Alternate outcomes
- Sensitivity analysis
- Scenarios

- ❑ **Interpretability** = Understanding how the model works internally
- ❑ **Explainability** = Explaining model decisions to stakeholders

Key Differences:

Explainability Methods: Global, Local, and Counterfactual explanations

Target Audience: Business, Technical, Regulatory, and End Users

Explanation Types: Why (reasoning), How (process), and What-If (scenarios)