"The Evolution of Natural Language Processing: Leveraging Generative AI and Transformers in Healthcare"

Anomaly Detection

- Anomalies in medical device data or patient monitoring systems can indicate potential safety risks.
- Detecting these anomalies in real-time can help healthcare providers take immediate action, ensuring patient safety and preventing adverse events.
- Anomaly detection plays a significant role in the identification of adverse events. It can help to uncover unusual patterns or deviations from normal care procedures that may indicate potential adverse events.
- Using anomaly detection for adverse event identification has several advantages. It can help to uncover hidden patterns and trends that might be missed by traditional methods. It can provide early warning of potential adverse events, allowing for timely intervention. It can also help to quantify the risk of adverse events, which can aid in risk management

The challenges we set out to solve

Unstructured language in digital channels
- Perform on a broad problem domain
- Handle specific channel style text
- Rare frequency of Anomaly Detection

1 in a 1000?

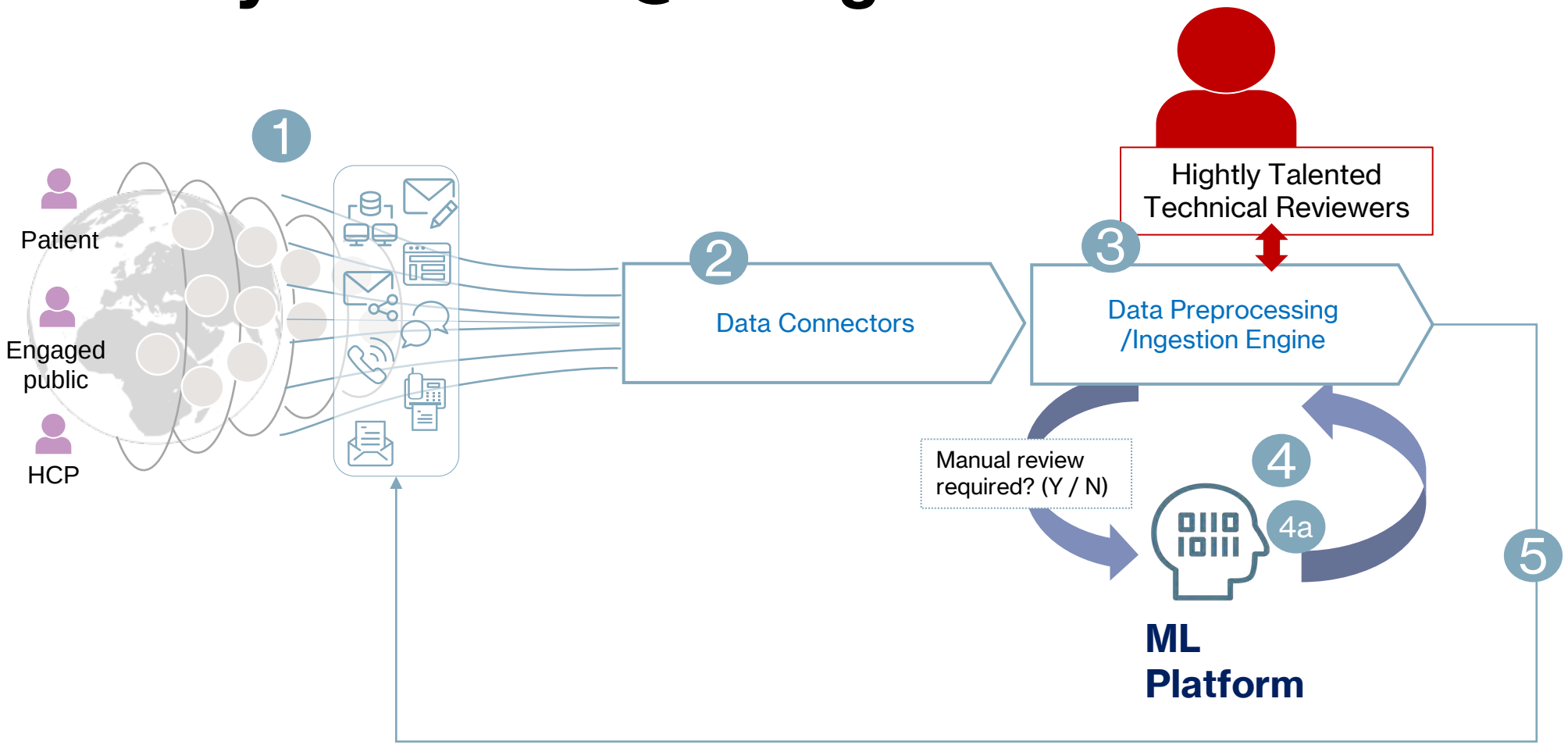
Medical domain knowledge
- Mimic practitioners‘ understanding
- Cause & effect reasoning

„Why did I get a headache after taking Aimovig?“

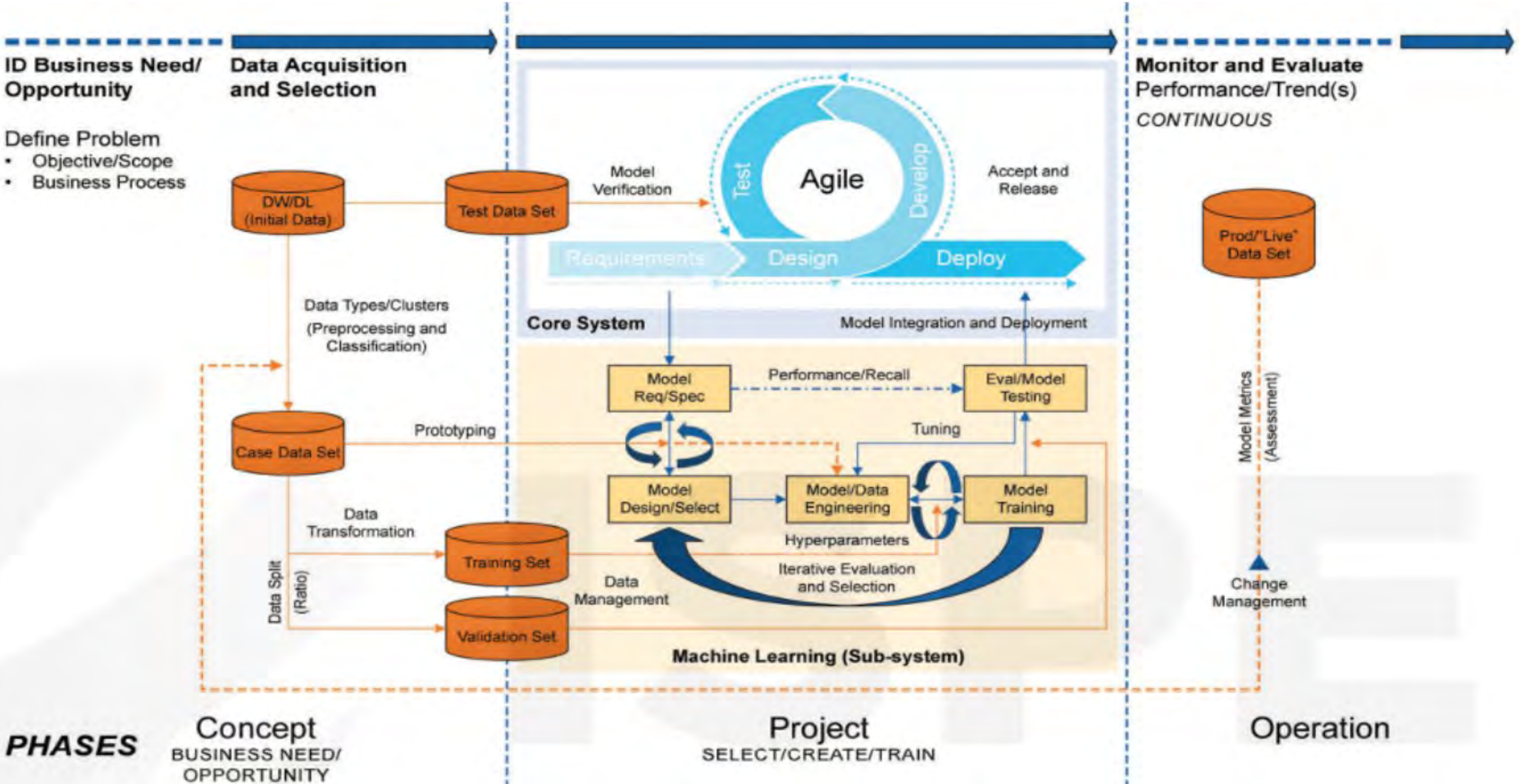
Business expectations
- Low fault tolerance (False Negative rate)
- Updating definitions
- GxP compliance

GxP-compliant AI:
Among the first in industry; but only in combination with **human** review factor

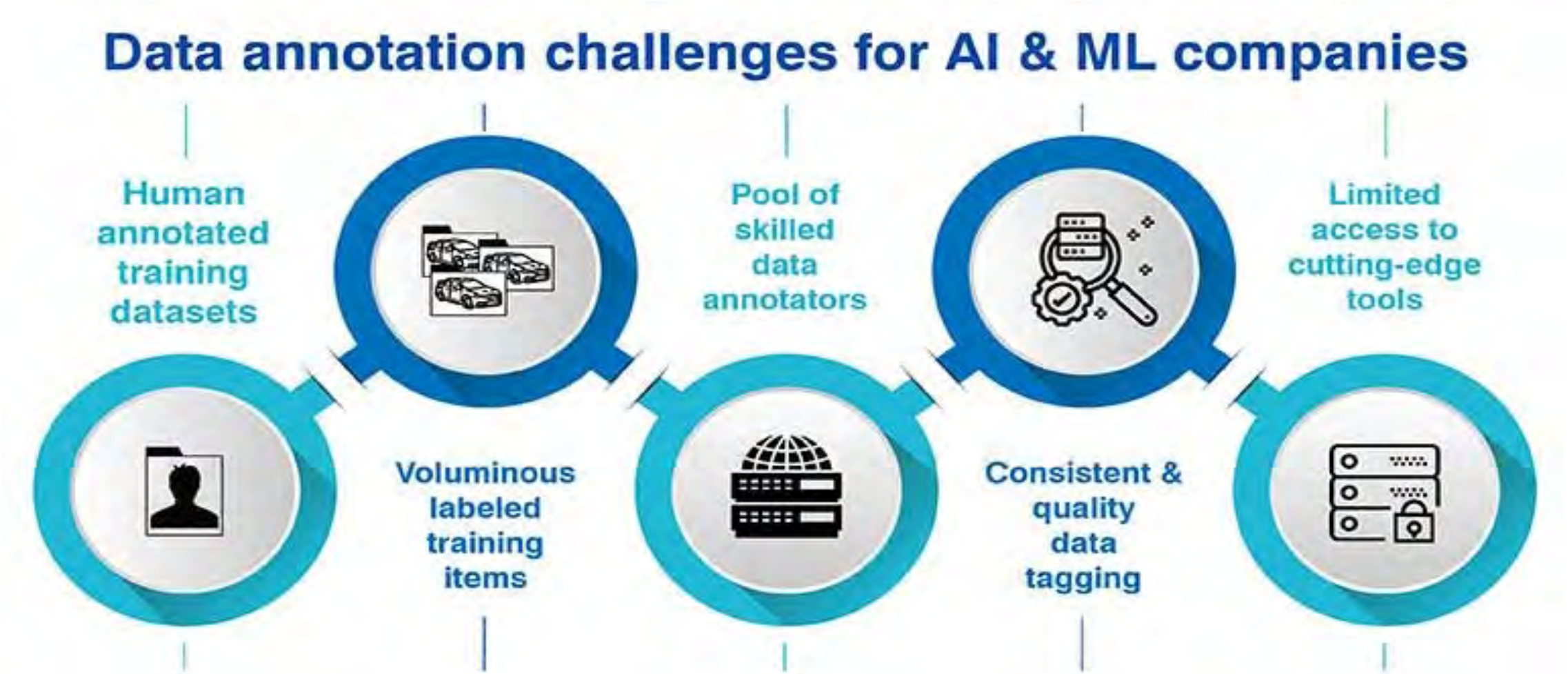
Anamoly Detection @ at a glance



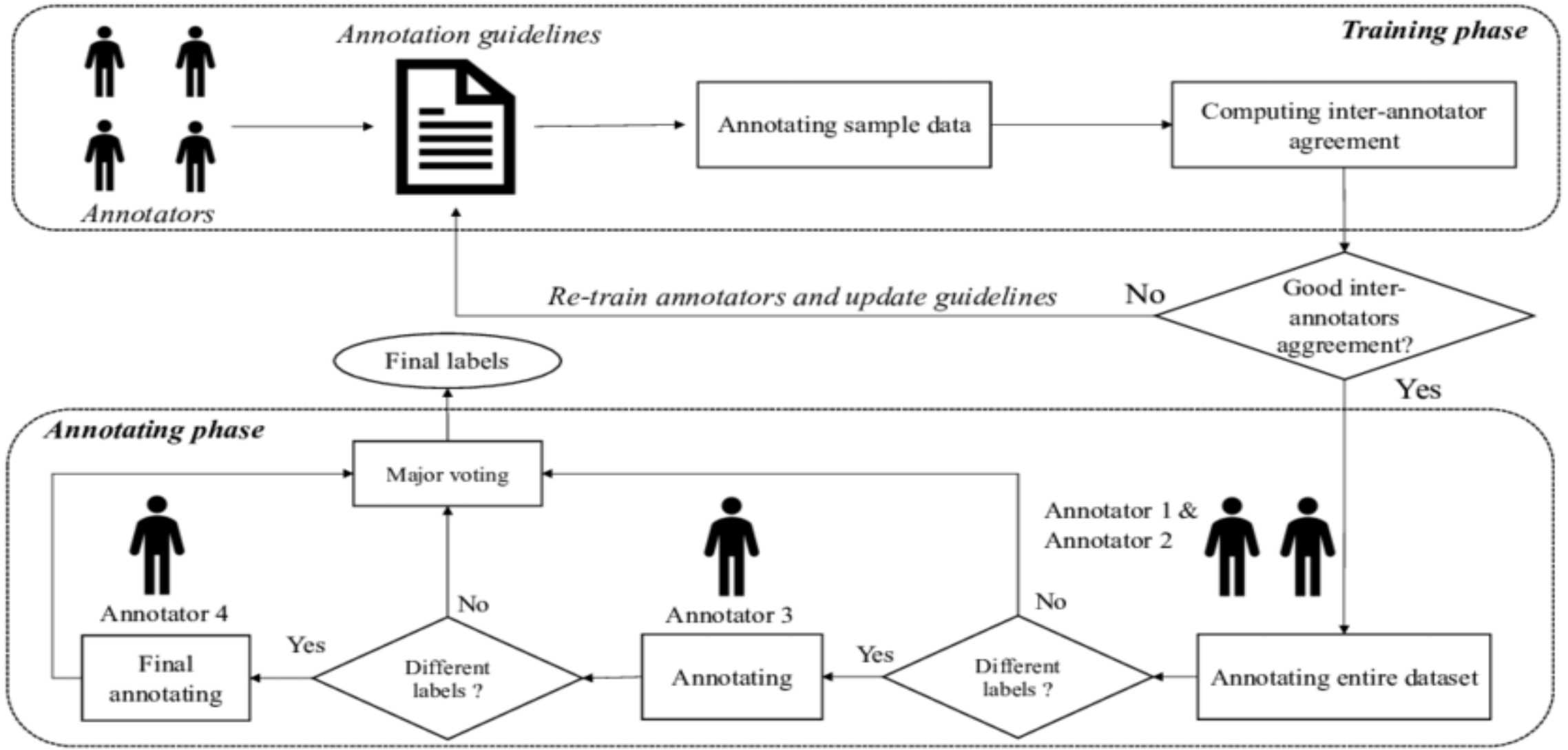
ML Algorithm Development Process



Why Data Annotation ?



Data Annotation ...



A comprehensive AI impl journey

- Develop a **balanced, well-labelled data** set
- Implement **robust** data **management** and **security** procedures
- Create and segregate **Training, Validation** and **Test Dataset** for Model training and Validation
- **Understand data context** by capturing human knowledge

- Build-in **repeatability** and **auditability** as part of system design
- Plan for **separate training environment** for AI model development and to support ongoing retraining
- Plan for **risk management framework** to mitigate 'Black-box' effect such as ensemble model approach

- Evaluate performance on **statistical model metrics and business acceptance criteria**
- Validate on Test Data reserved during model build
- Perform **Human Intelligence testing** with Business and Domain SMEs
- Incase of retrained model, re-validate on the reserved test data to ensure model performance is superior or same

- **Real world study** – Parallel run with human review
- Monitor algorithm **on performance metrics**: No. of Flagged Records, False Positive & False Negative rate.
  - Perform periodic manual review of un-flagged records to identify false negatives
- Establish Change control for **retraining/model updates**
- Re-validate the system when major changes are implemented impacting the model build and architecture

Anomaly Identification Framework

- Uses a deep ensemble of text classifiers:

  Multiple BERT and BioBERT models trained with different random seeds

  ⇒ If **any** of these models predicts text as adverse event, forward it to human review

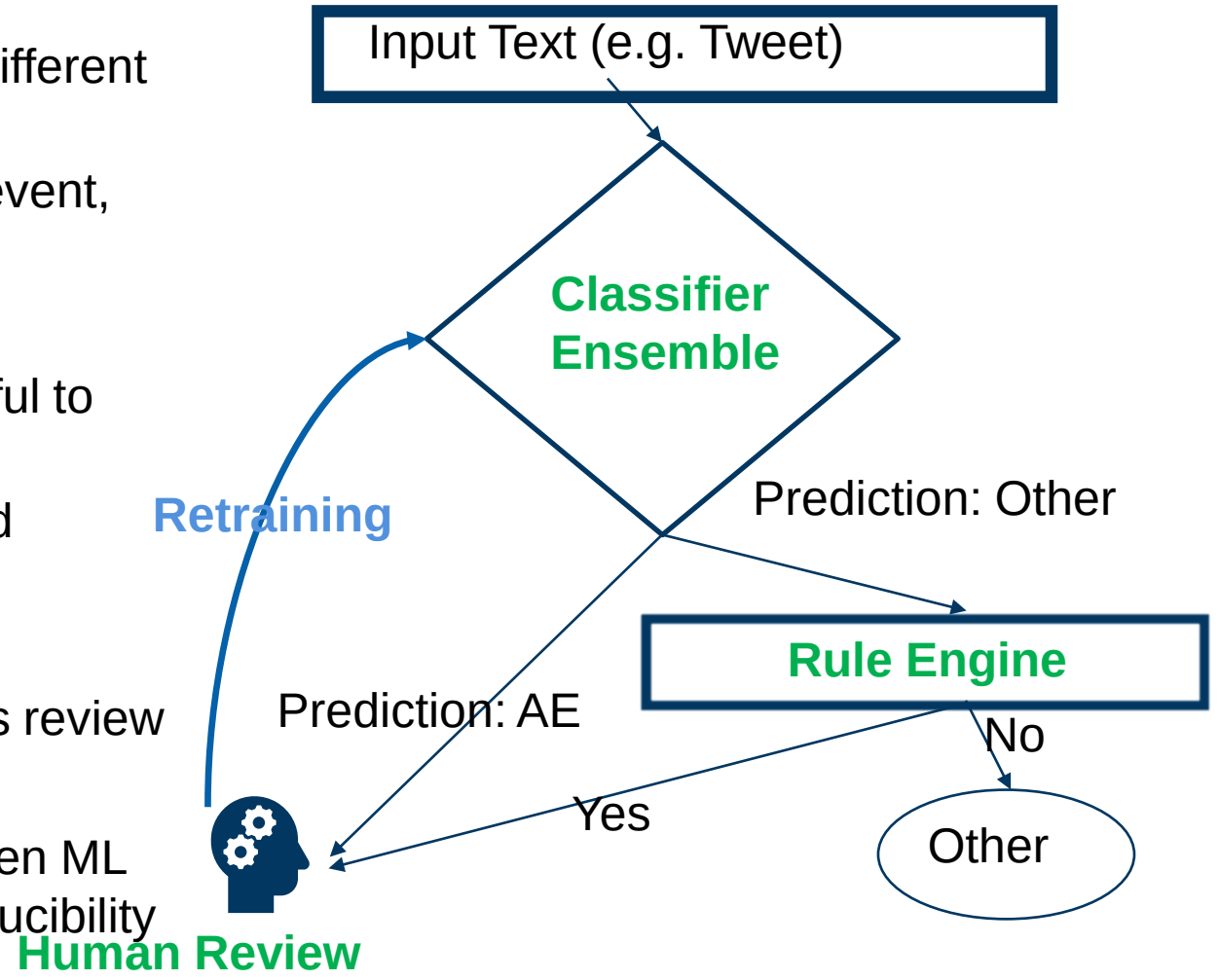
- Additionally:

  Review any text that fits predefined rules (e.g. useful to capture additional edge cases)

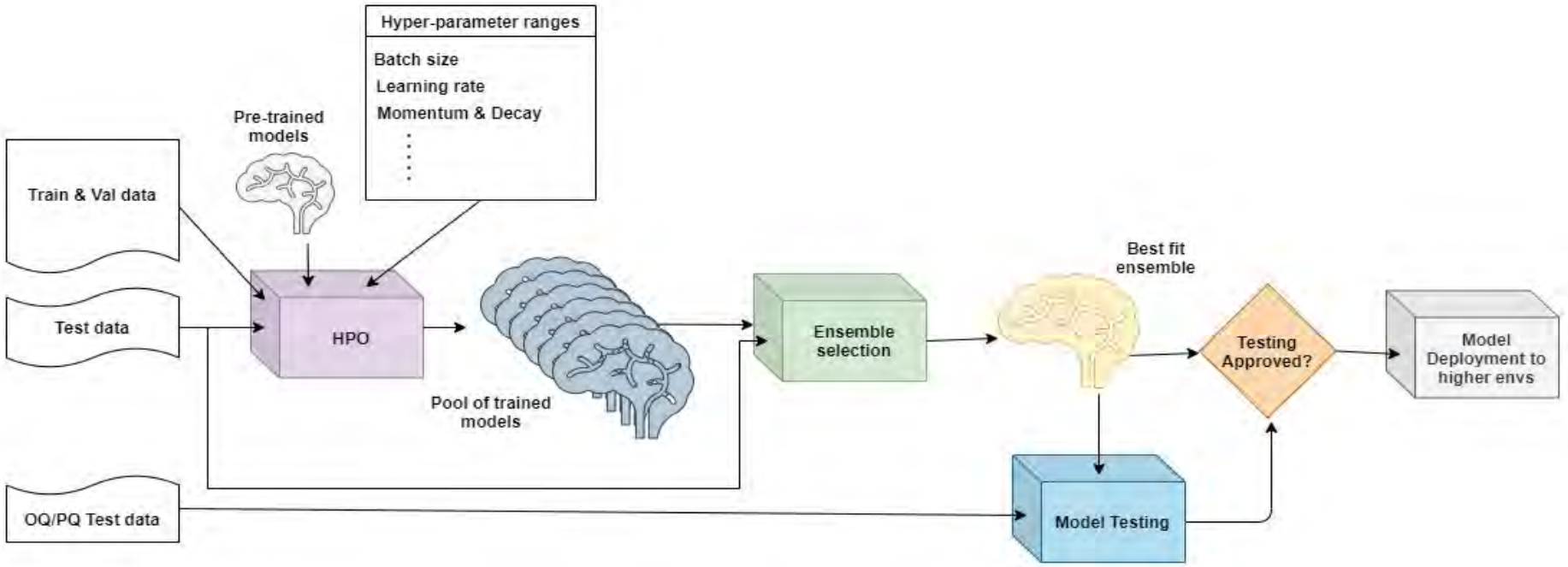
  The rules are currently based on Product and Medical Event/Anomaly Identification

- Un-flagged records are then sub-sampled using "SOP"approved sampling methodology for business review to ensure Algorithm performance is not degrading

- Incase of safety critical use cases, ensure the chosen ML framework/computational resources support reproducibility (eg Pytorch vs Tflow)

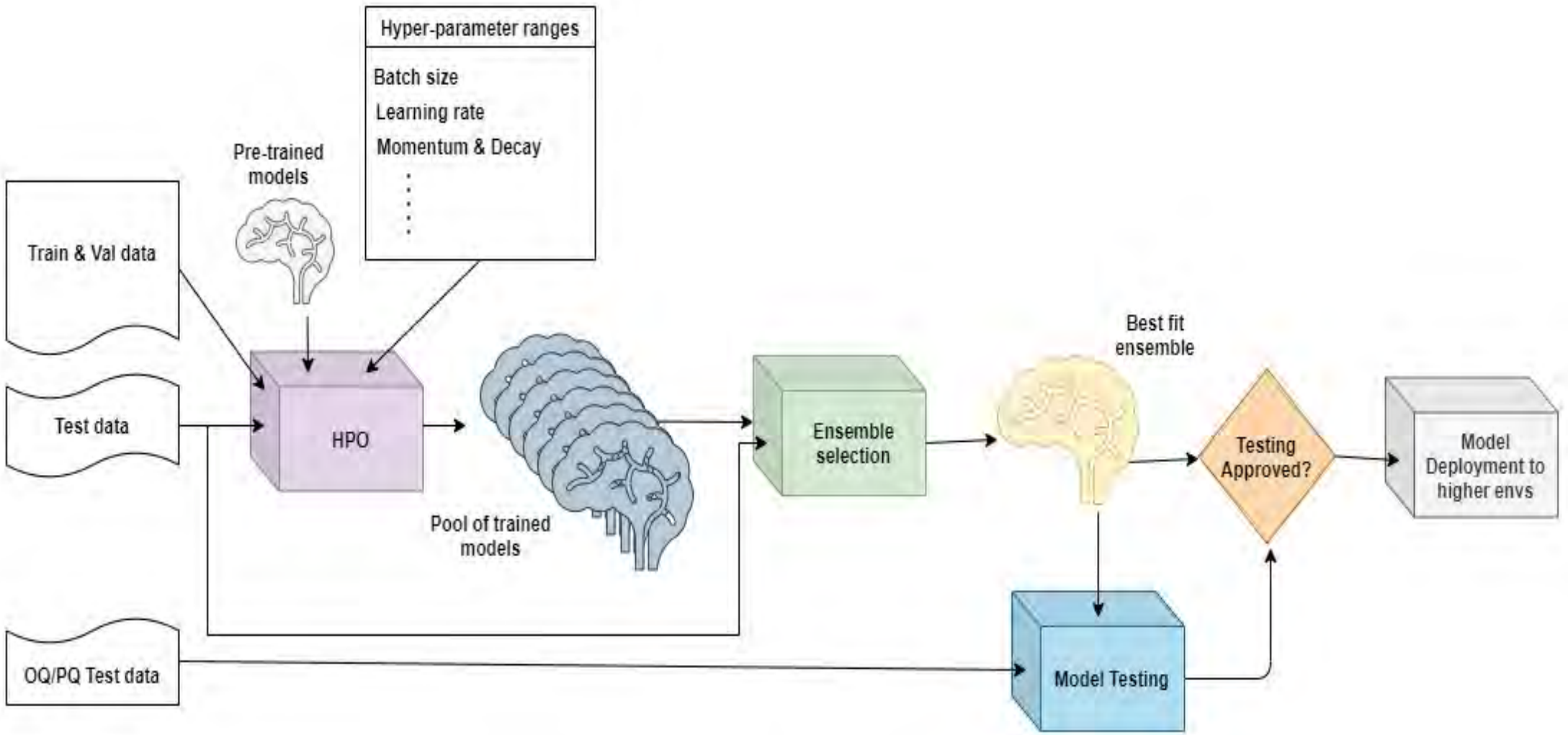


Machine learning Model Framework – Continuous Training for optimized performance



- We generate a pool of trained models and select the best fitting (on test set) ensemble. This ensemble validated using OQ/PQ data.
- We use `**PyTorch**` as the underlying framework, **'HuggingFace` Transformer** models as the pre-trained models. All models are trained on **8x *Tesla K80* GPUs in a 'Data Parallel' manner**.
- We also use `**MLflow**` for experiment tracking and model registry.

LLM + Guardrails



Challenges in BERT models

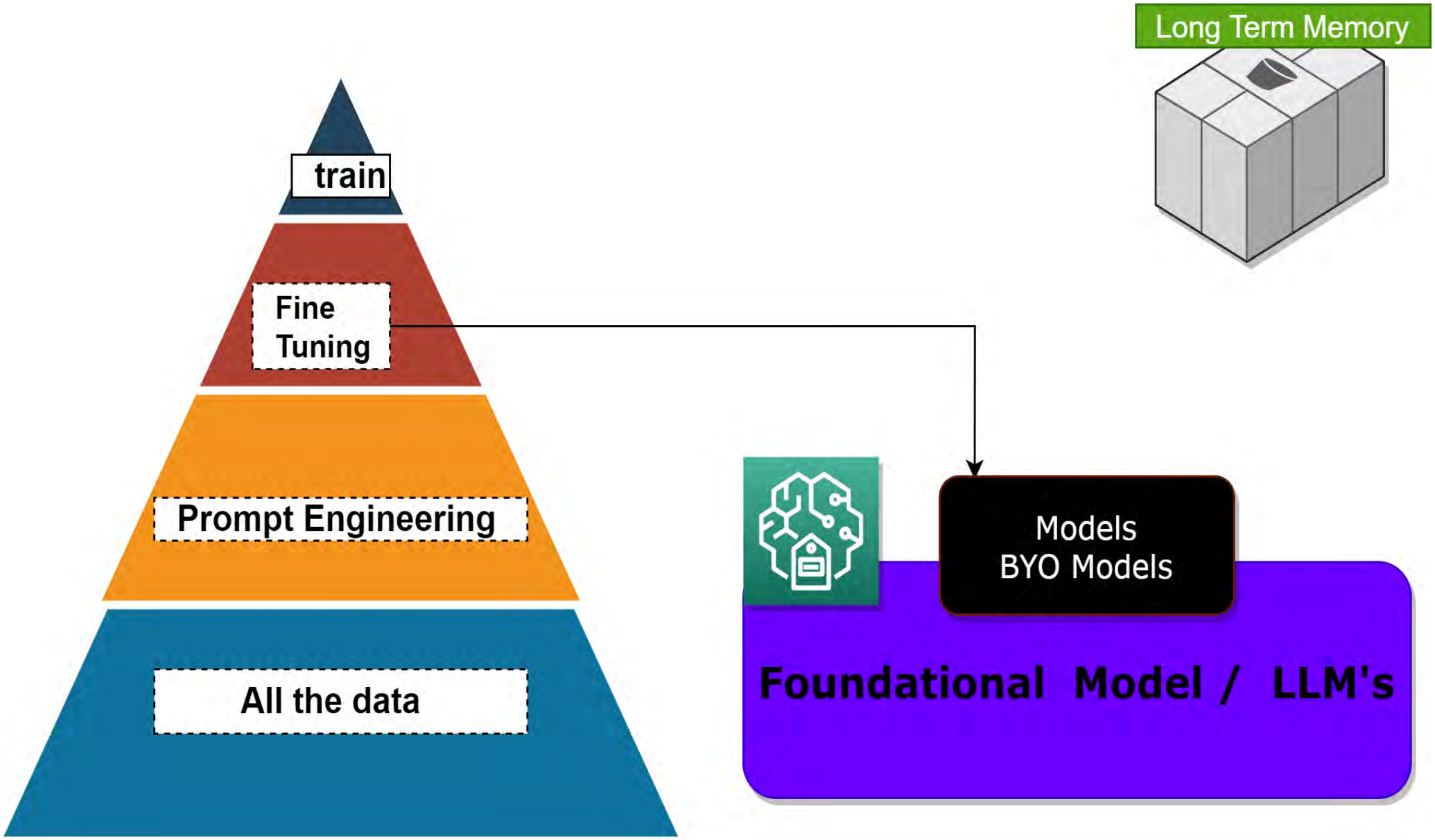
1. Fine-tuning pre-trained models like BERT (from Google) and GPT (from OpenAI) is a common practice in NLP tasks. Fine-tuning involves adjusting the pre-trained model to better suit the specific task at hand. While BERT and GPT-4 are both transformer-based models, they have different architectures and use cases, which can lead to different complications during fine-tuning.
2. BERT is designed to understand the context of every word in a sentence by looking at the words that come before and after it. This makes it particularly useful for tasks that require understanding the context of words in a sentence, such as Named Entity Recognition (NER) or question-answering tasks.
3. Complications in fine-tuning BERT can include:

- **Training Time**: BERT models have a lot of parameters, which can make the fine-tuning process computationally expensive and time-consuming.
- **Overfitting**: Due to its complexity and capacity, BERT can easily overfit on smaller datasets.
- **Understanding BERT's Representations**: It can be challenging to understand why BERT makes certain predictions, as its internal representations are complex and not easily interpretable.

Generative AI and Prompt Engineering

- Through the utilization of a foundational model, we have the capacity to craft more specialized and advanced models that are specifically designed for particular domains or use cases. For instance, generative AI can utilize foundation models as a core for creating large language models. By leveraging the knowledge learned from training on vast amounts of text data, generative AI can perform the identification of anomaly detection.
- Prompt Engineering in Generative AI is an advanced tool that leverages the capabilities of AI language models. It optimizes the performance of language models by developing tactical prompts, and the model is given clear and specific instructions.

Gen AI – LLM's approach




THANK
YOU