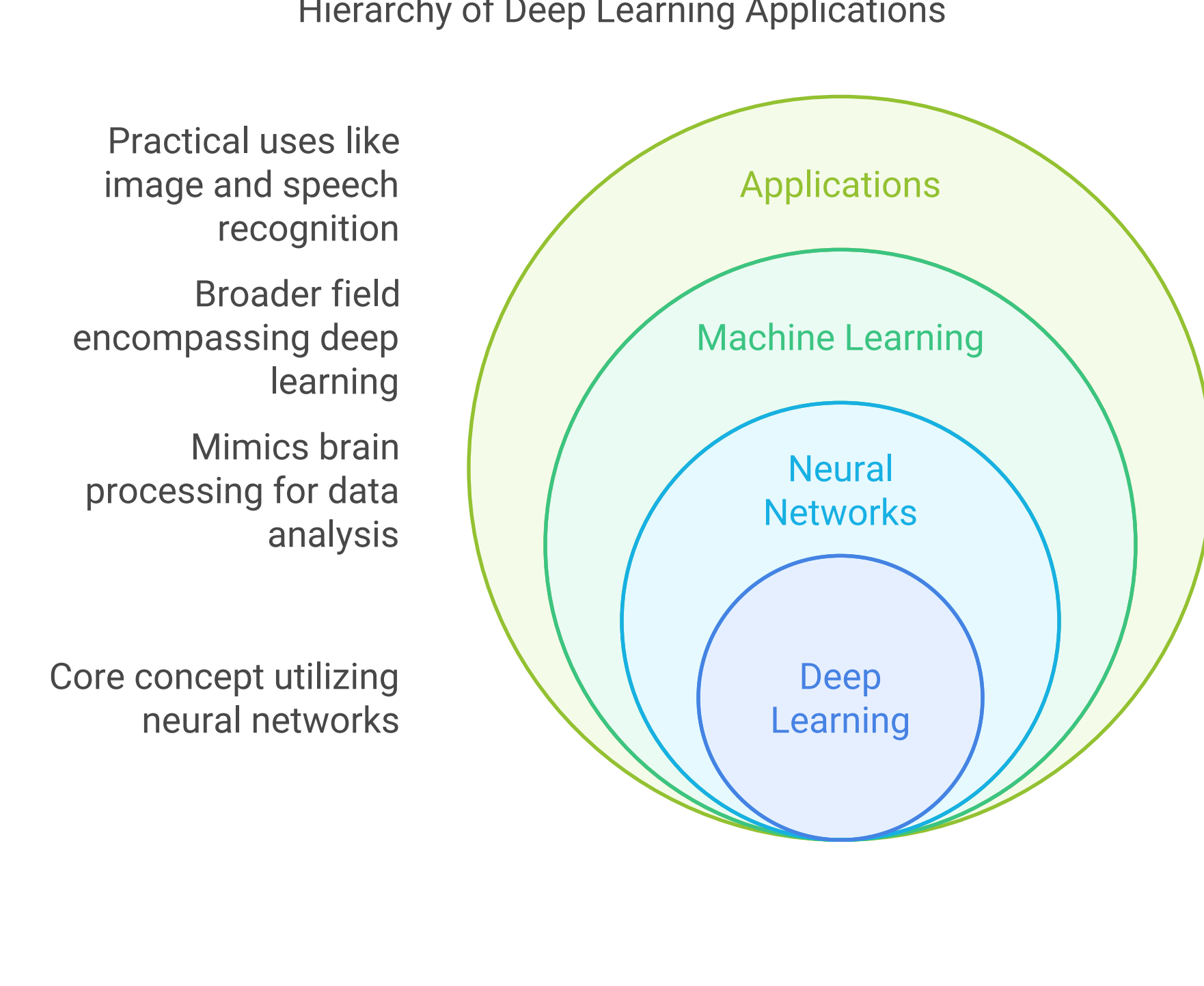


Introduction to deep learning with PyTorch

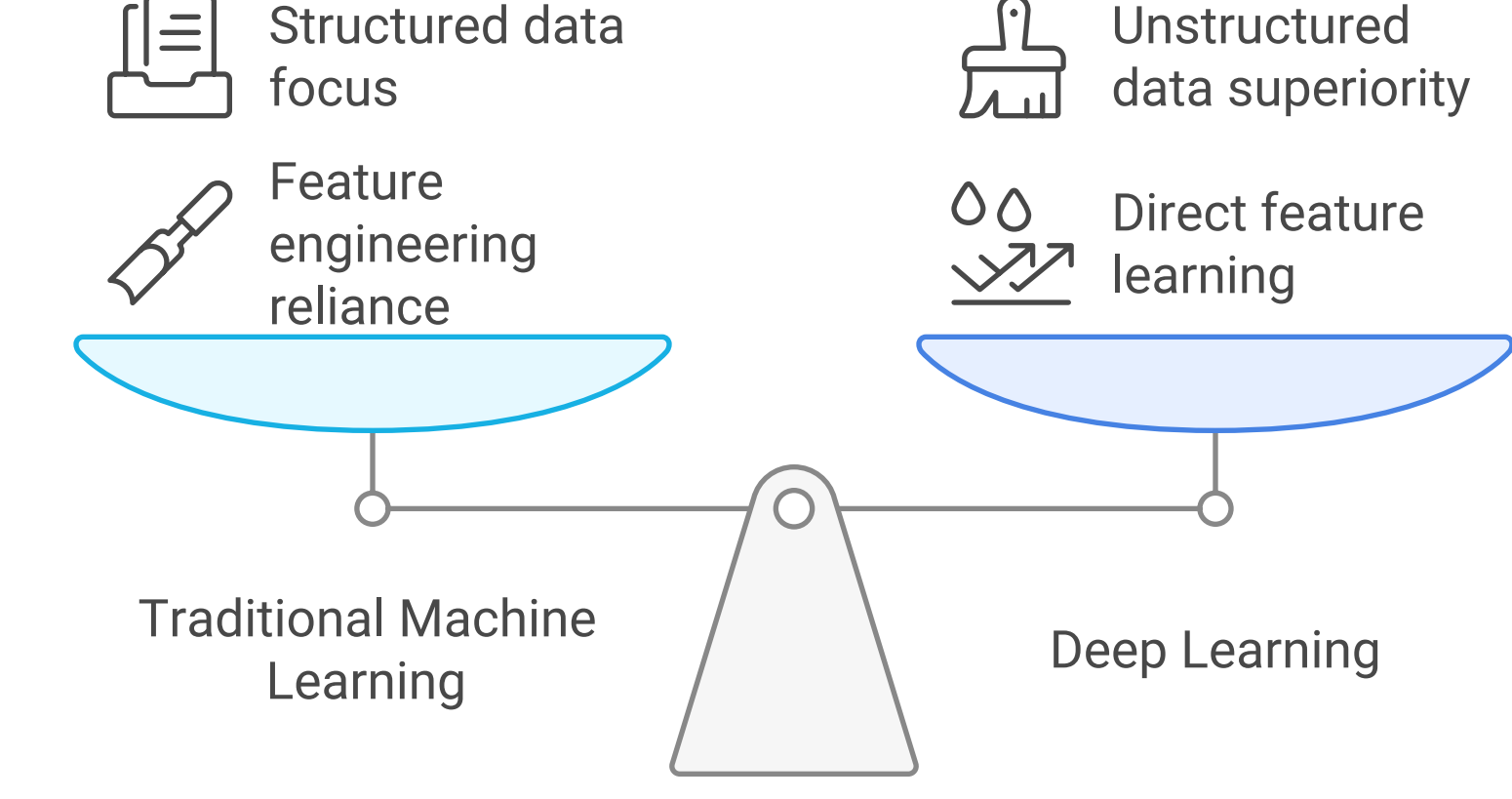
What is Deep Learning?



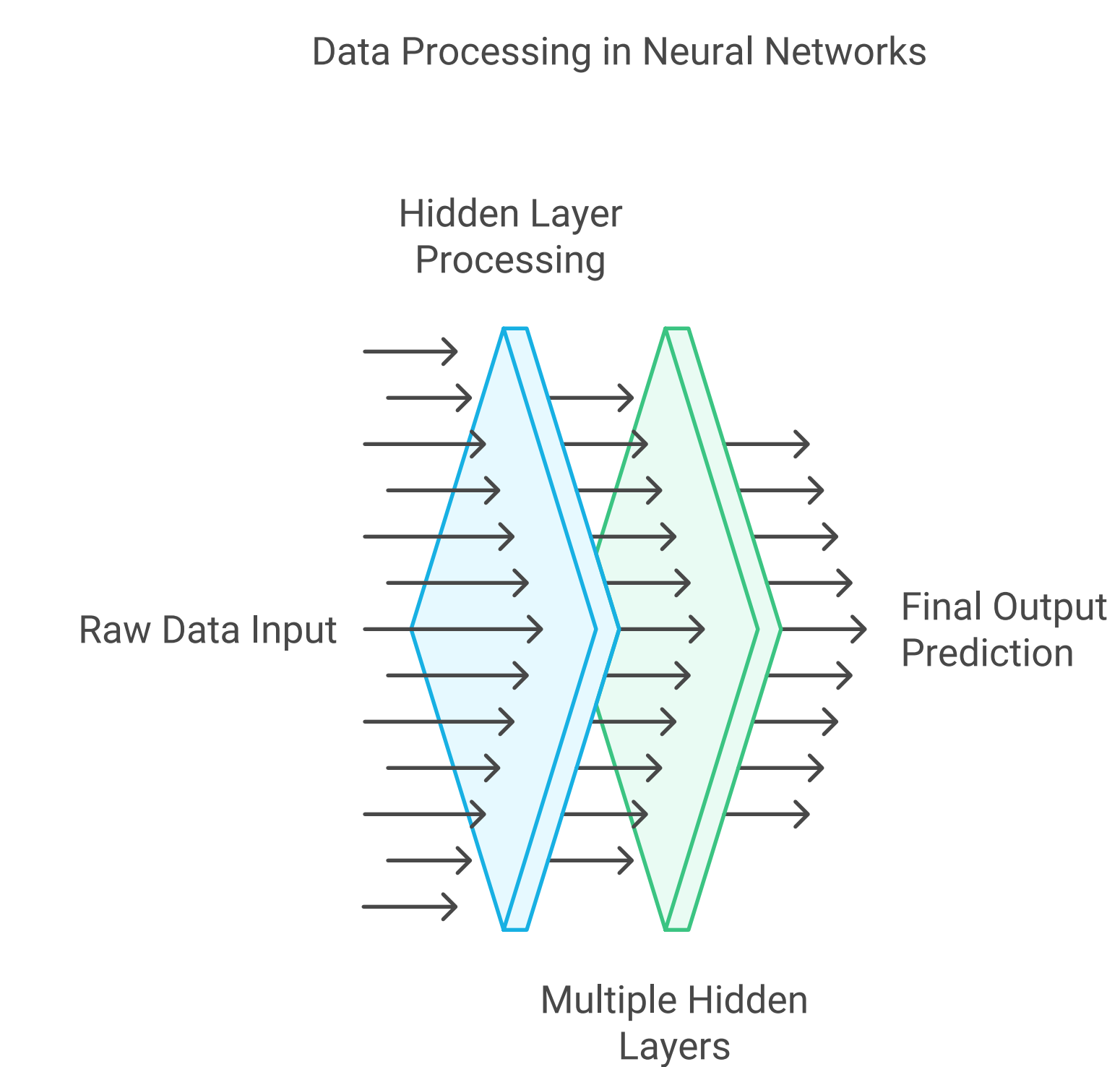
Applications: Language translation, self-driving cars, medical diagnostics, chatbots.

Deep Learning:

- Utilizes "layered" network structures.
- Performs feature learning (representation learning) directly from raw data.
- Outperforms traditional machine learning in unstructured data analysis.



Deep Learning Fundamentals



Definition: A subset of machine learning with networks of:

- **Inputs:** Data fed into the model.
- **Hidden Layers:** Computational layers extract patterns.
- **Outputs:** predictions or classifications.

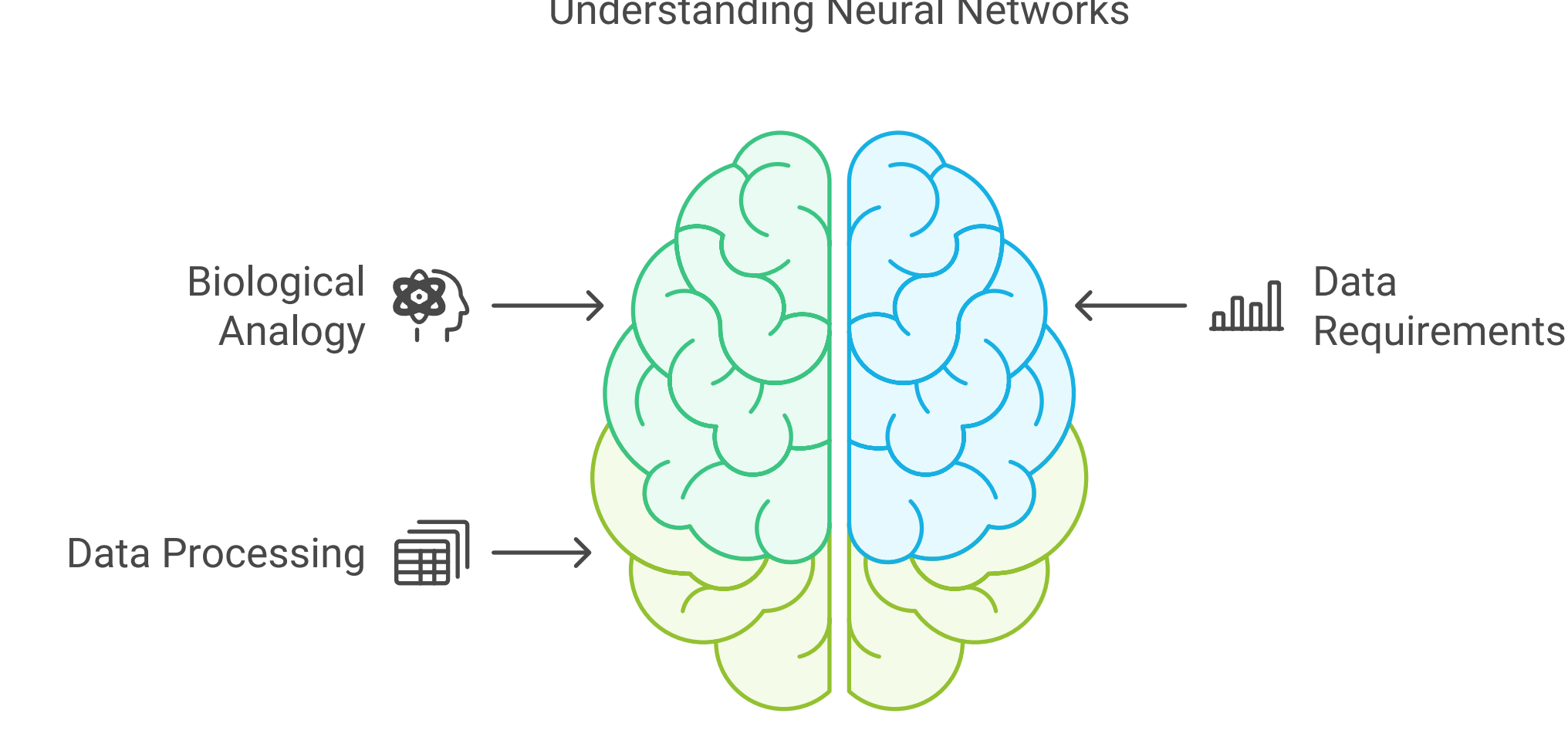
Layer Depth: Networks may have one or multiple hidden layers.

Inspiration Behind Deep Learning

Biological Analogy: Modeled after the human brain's interconnected neurons.

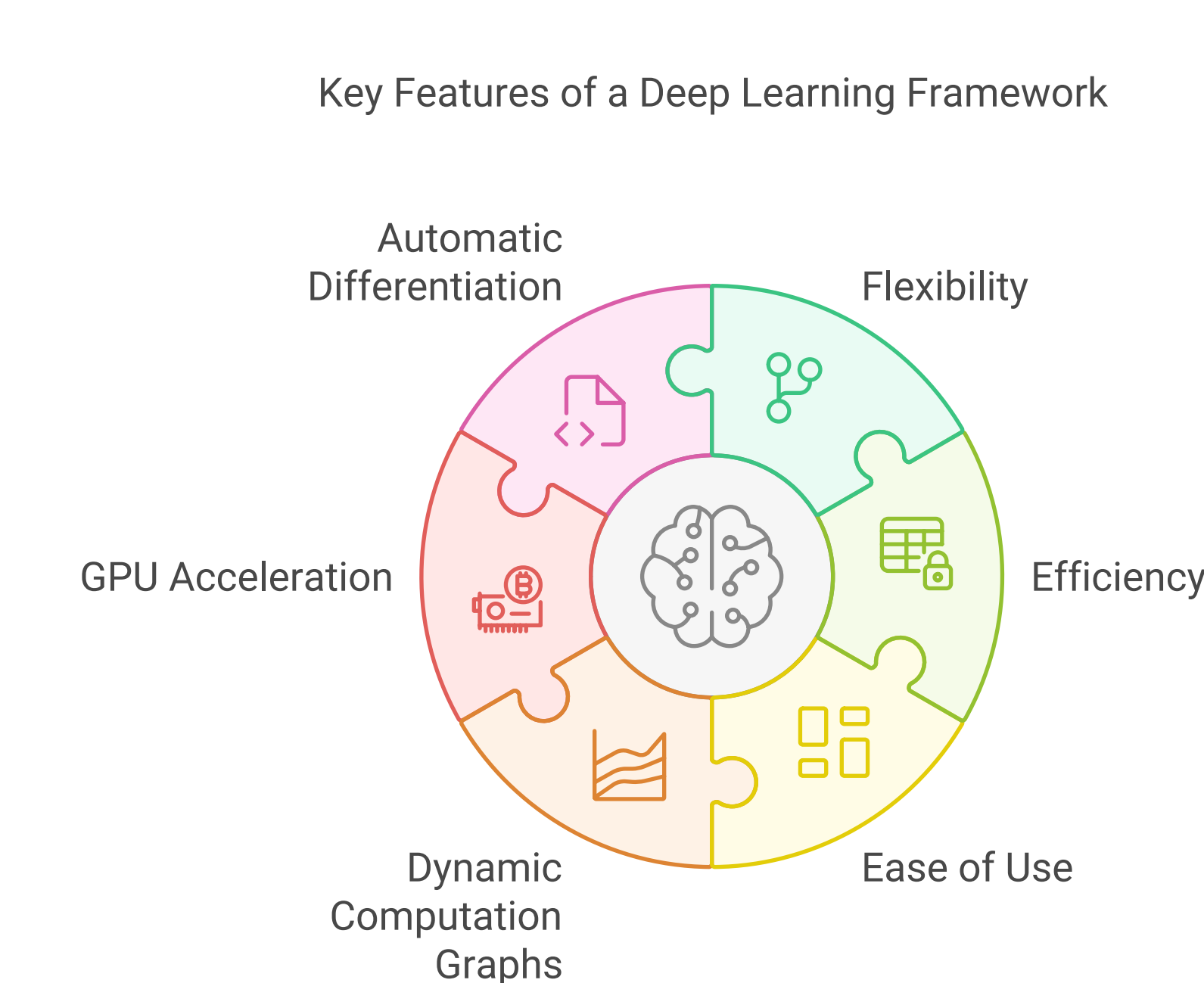
Neural Networks:

- Require substantial data for training (hundreds of thousands of points).
- Process unstructured data effectively.



What is PyTorch?

- **Definition:** PyTorch is an open-source deep learning framework developed by Facebook's AI Research lab.
- **Purpose:**
 - Designed to be flexible, efficient, and easy to use for building and training deep learning models.
 - Provides support for dynamic computation graphs, GPU acceleration, and automatic differentiation.



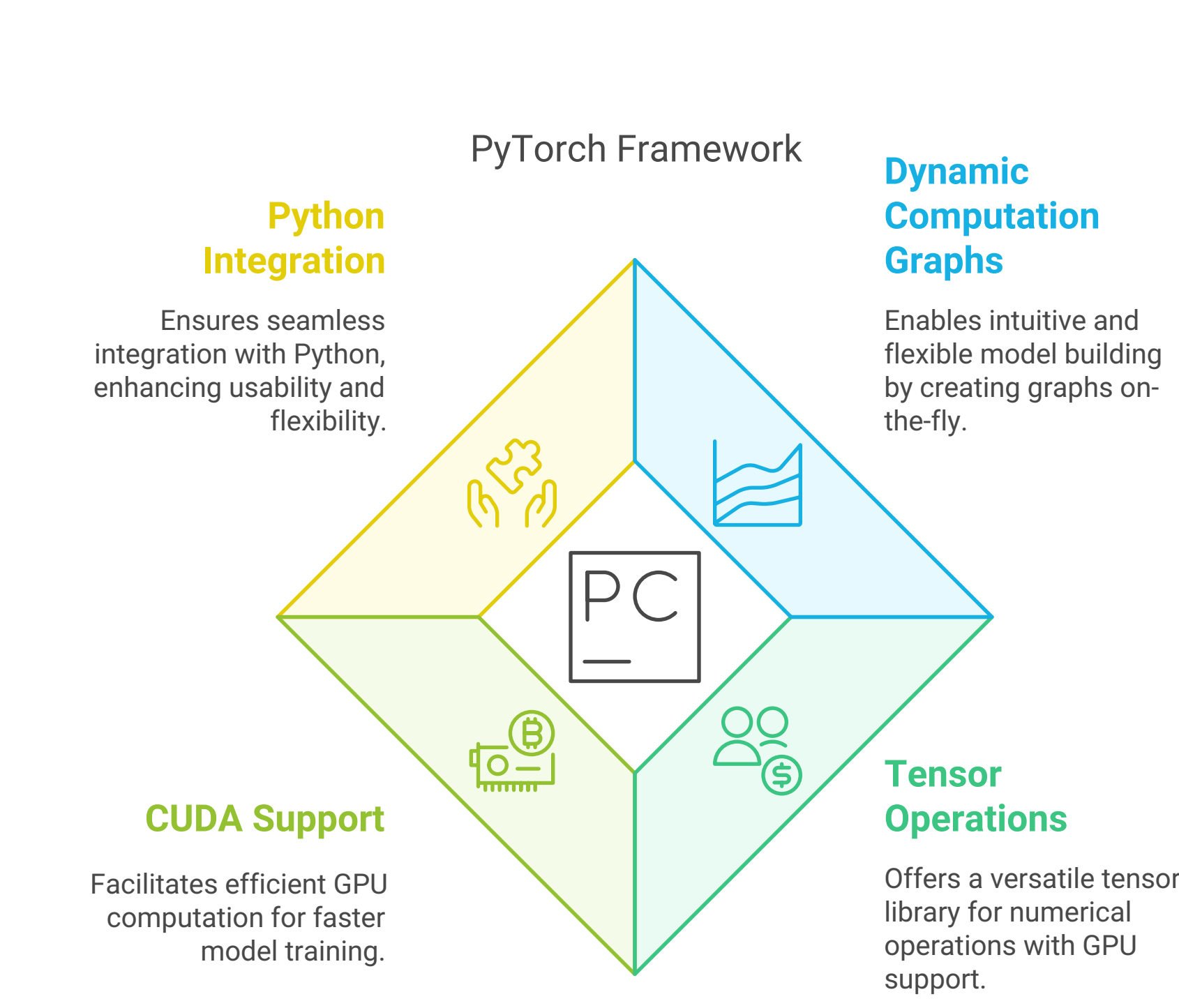
Key Features Of PyTorch

Dynamic Computation Graphs: Unlike static computation graphs, PyTorch builds graphs as operations are performed, making debugging easier and more intuitive.

Tensor Operations: PyTorch provides a flexible tensor library for performing numerical operations, much like NumPy, but with additional support on GPU acceleration.

CUDA Support: PyTorch allows seamless computation on GPUs via CUDA, providing faster model training times.

Integration with Python: PyTorch integrates well with Python and is highly favored in the research community for its simplicity and flexibility.



Why Use PyTorch?

- **Ease of Use:**
 - Intuitive design, especially for Python users.
 - Syntax and functionality align closely with NumPy.
- **Dynamic Computation Graphs:**
 - Allows on-the-fly graph construction, making debugging easier.
- **Flexibility:**
 - Ideal for research and experimentation due to its adaptable framework.
- **Community and Resources:**
 - Supported by a robust developer community.
 - Extensive documentation and resources available.
- **Industry and Academia:**
 - Widely used by engineers and researchers.
 - Many state-of-the-art models and papers utilize PyTorch.
- **GPU Support:**
 - Optimized for training on GPUs, offering faster computation for deep learning tasks.

