AI Security

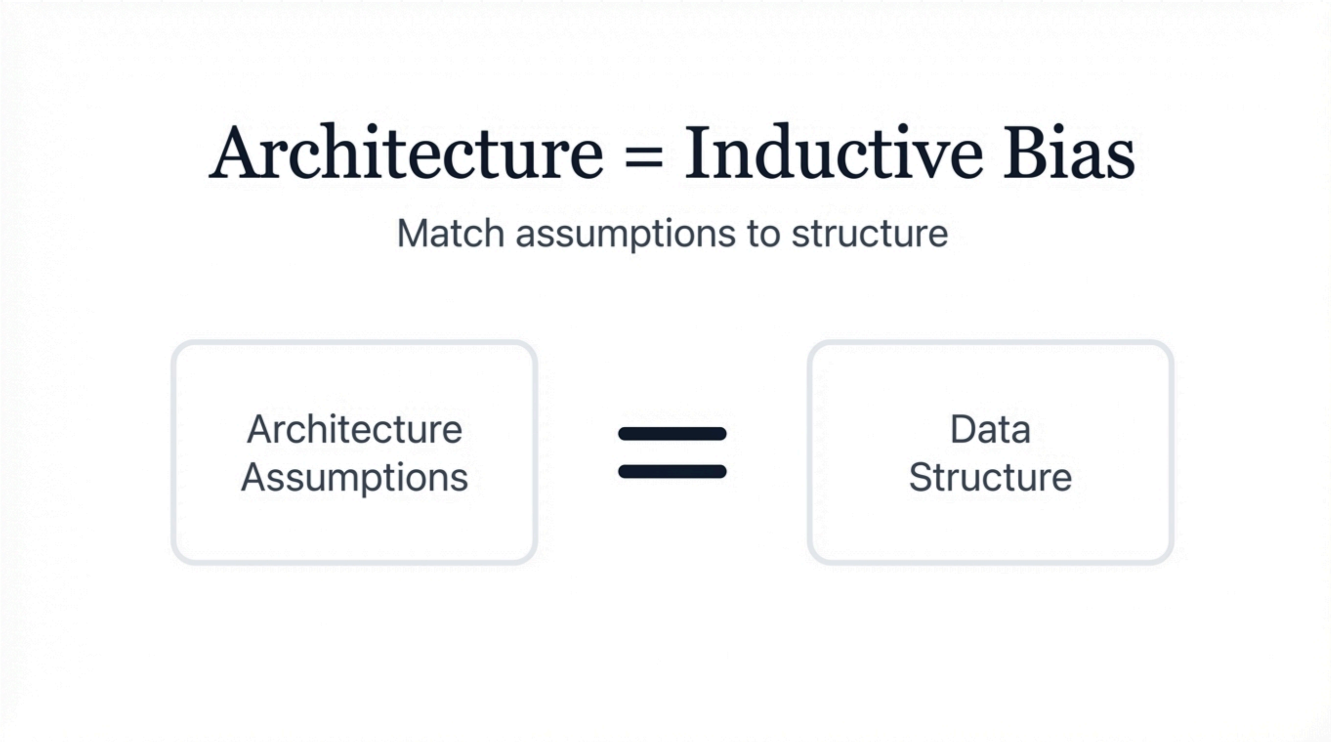
Neural Network Architecture Selection Cheat Sheet

Neural Network Architecture Selection Cheat Sheet

Author: Scott Thornton, perfecXion.ai **Published:** January 25, 2026 **Read Time:** 10 minutes

https://perfecxion.ai

The Core Principle

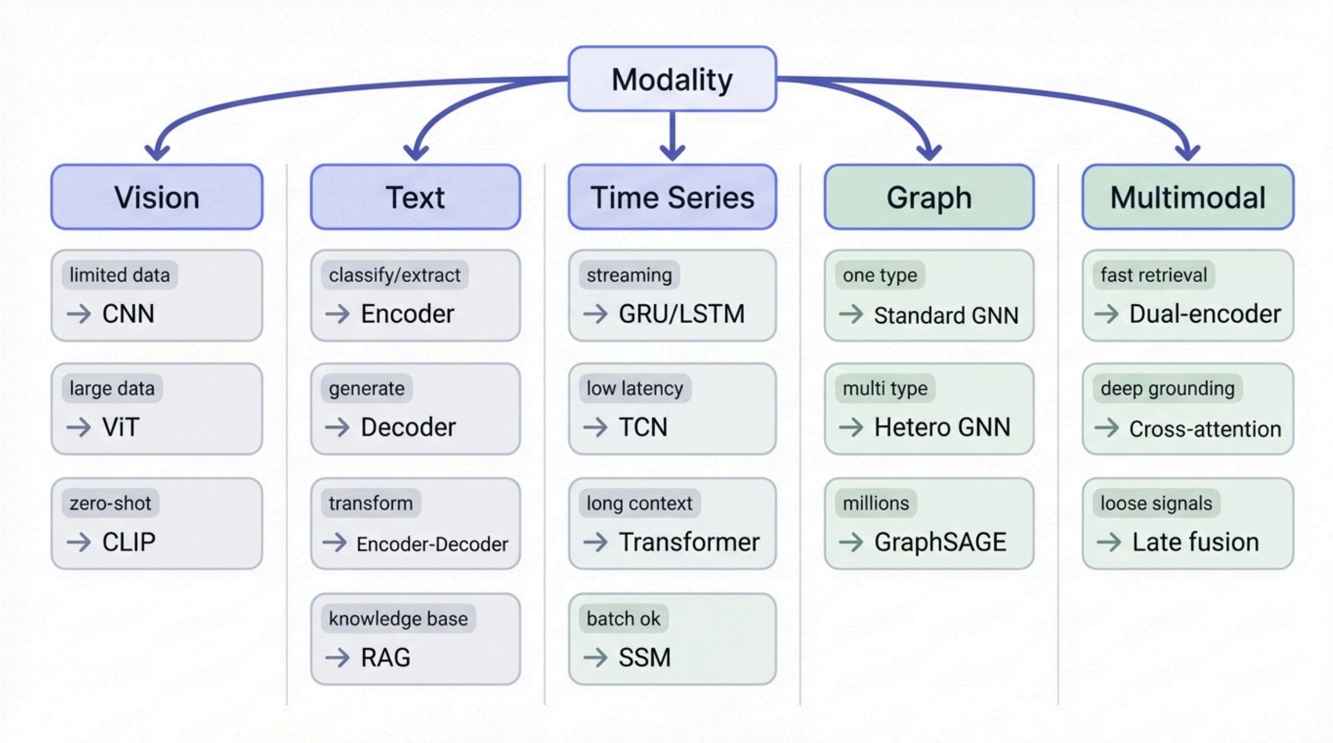


Inductive Bias Match

Architecture = Inductive Bias. Match your architecture's assumptions to your data's structure.

Data Structure	Architecture Assumes	Use	Watch For
Grid/Spatial (images)	Nearby elements correlate	**CNN**	Misses global context
Sequential (text, time)	Order matters	**Transformer/RNN**	Cost/latency explosion
Relational (networks)	Explicit relationships	**GNN**	Graph construction errors
Tabular (spreadsheets)	Minimal structure	**Trees → MLP**	High data requirements
Multiple types	Separate encoders needed	**Multimodal**	Complexity without gain

Quick Decision by Modality



Quick Decision by Modality

Vision

- **Limited data / edge deployment** → CNN (ResNet, EfficientNet)
- **Large data + pretrained available** → ViT
- **Zero-shot / retrieval / similarity** → CLIP-style embeddings

Text

- **Understand / classify / extract** → Encoder (BERT-style)
- **Generate / complete** → Decoder (GPT-style)
- **Transform (translate, summarize)** → Encoder-Decoder (T5)
- **Large knowledge base** → RAG (retrieval + generation)

Time Series

- **Streaming + low latency** → GRU/LSTM or TCN

- **Batch OK + long context** → Transformer or SSM

Graph

- **One node/edge type** → Standard GNN (GCN, GAT)
- **Multiple types** → Heterogeneous GNN
- **Millions of nodes** → GraphSAGE with sampling

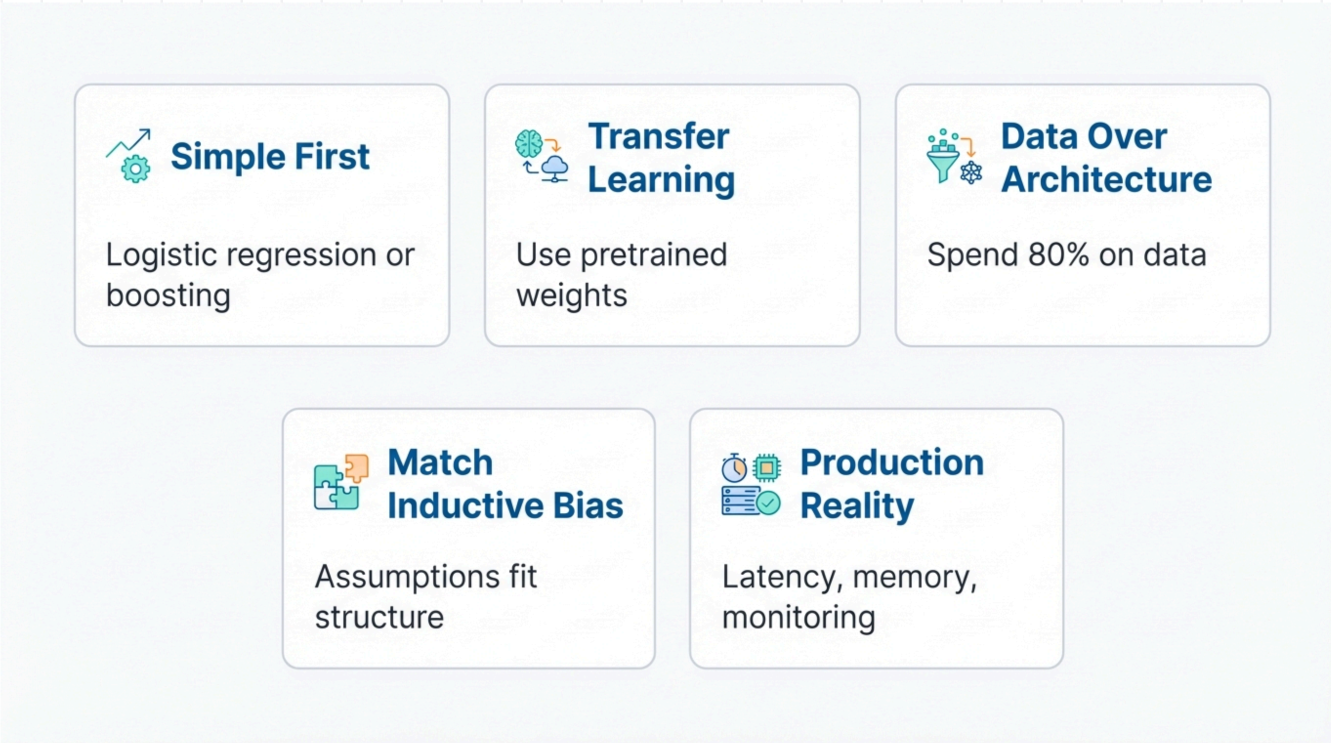
Multimodal

- **Fast retrieval/matching** → Dual-encoder (CLIP-style)
- **Deep grounding (VQA)** → Cross-attention
- **Loosely coupled signals** → Late fusion

Data Quantity Rules of Thumb

Samples	Approach
< 1,000	Classical ML. Heavy transfer learning if neural.
1,000–10,000	Transfer learning essential. Fine-tune pretrained.
10,000–100,000	Most architectures viable with pretrained start.
100,000+	Training from scratch becomes reasonable.

The Five Principles

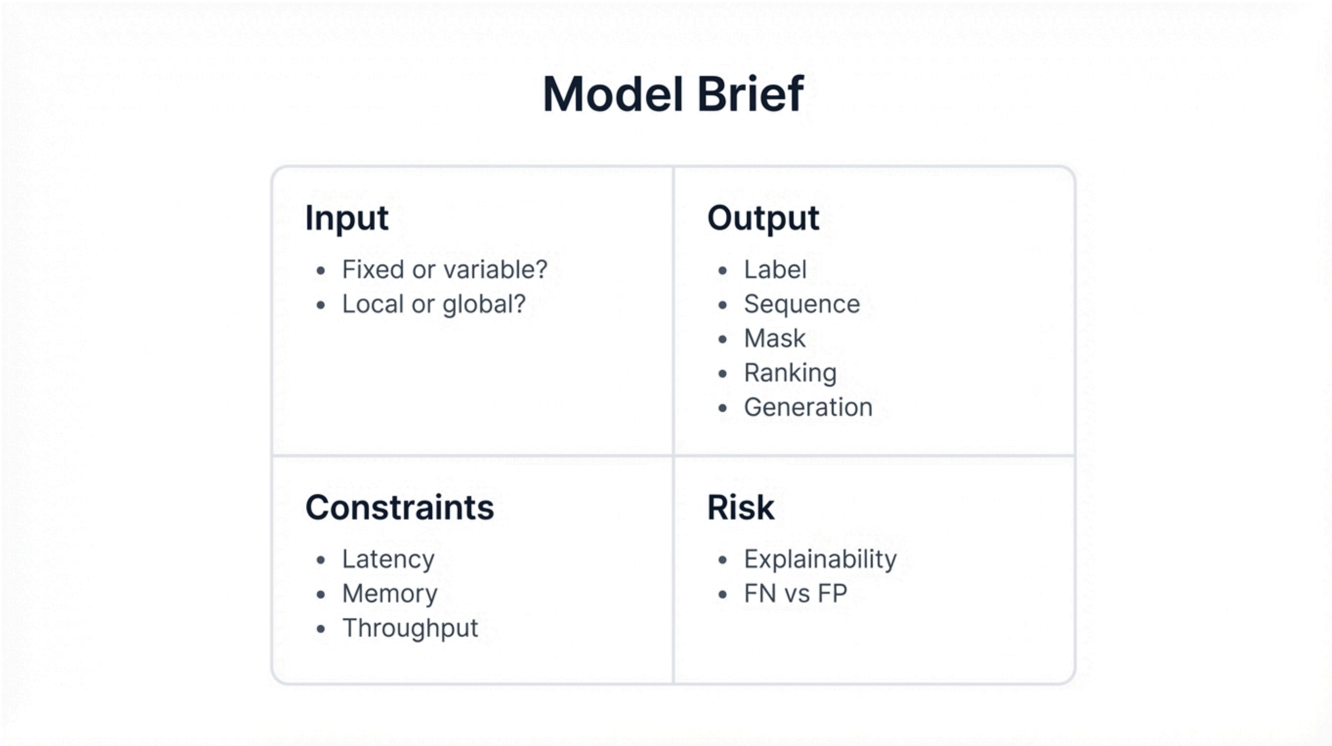


Five Principles Cards

1. **Simple First** — Try logistic regression or gradient boosting before neural networks.
2. **Transfer Learning Default** — Never train from scratch if pretrained weights exist.
3. **Data Over Architecture** — The best architecture can't fix bad data. Spend 80% on data quality.
4. **Match Inductive Bias** — Choose architectures whose assumptions match your data's true structure.
5. **Production Reality** — Consider latency, memory, and monitoring from the start.

Before You Start: Model Brief

Answer these before choosing:



Model Brief Checklist

Input: Fixed or variable size? Local or global signal?

Output: Label, sequence, mask, ranking, or generation?

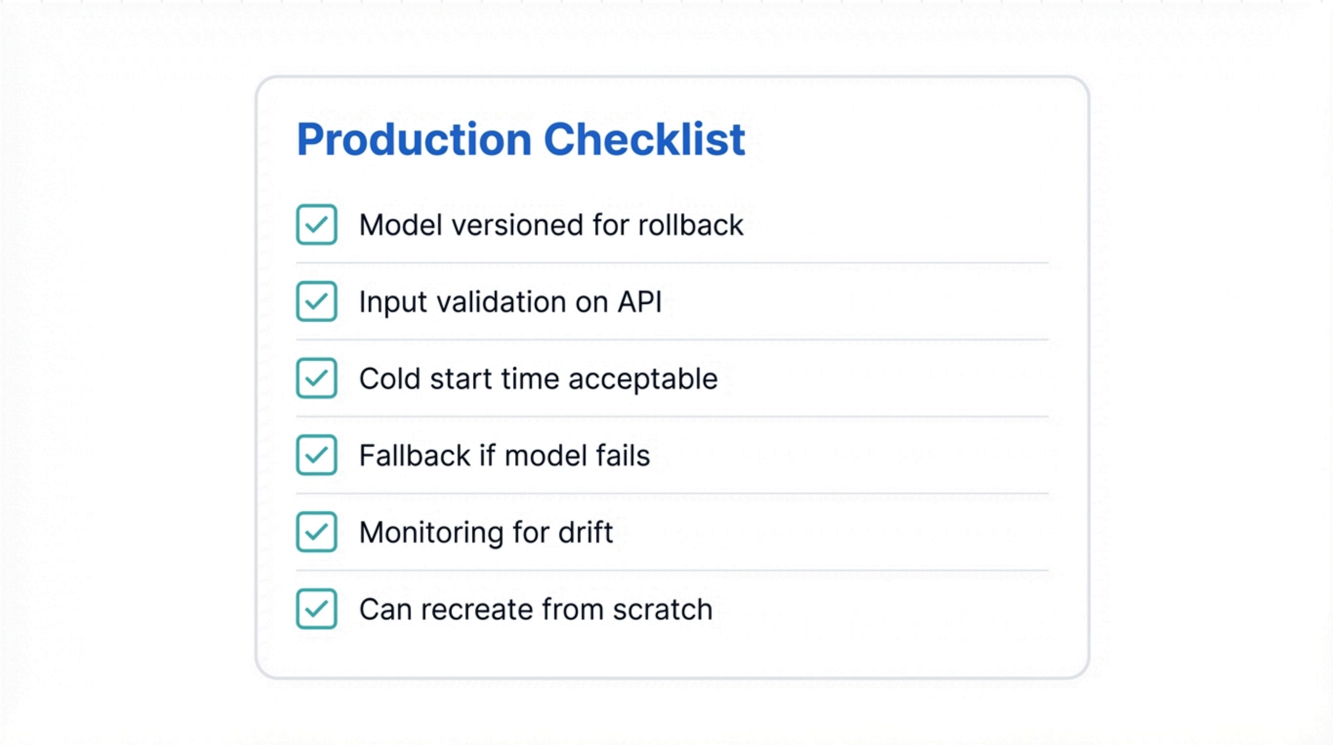
Constraints: Latency requirement? Memory budget? Throughput needs?

Risk: Explainability required? False negative vs false positive tolerance?

Common Mistakes

Mistake	Reality
"Transformers are always best"	Wasteful for tabular, small vision, edge
Ignoring classical ML for tabular	XGBoost/LightGBM often wins
Training from scratch	Fine-tuning needs 100x less data
Deeper = better	Diminishing returns, overfitting risk
Adding modalities "because it might help"	Complexity without signal = noise

Production Checklist



Production Checklist

- ☐ Model versioned for rollback?
- ☐ Input validation on API?
- ☐ Cold start time acceptable?
- ☐ Fallback if model fails?
- ☐ Monitoring for drift?
- ☐ Can recreate from scratch if needed?

Optimization Options

Technique	Result
Quantization (float32 → int8)	4x smaller, 2-4x faster
Pruning	Remove near-zero weights
Distillation	Small student mimics large teacher

Full Guide

For comprehensive coverage with worked examples and deep-dives into each architecture family:

- The Practitioner's Guide to Choosing Neural Network Architectures (/articles/neural-network-architecture-selection-guide.html)

perfecXion.ai

Thank You for Reading

Explore more AI security research at **perfecxion.ai**

This document was generated from perfecXion.ai
For the latest updates, visit the online version