**🖼️ Diagram: CNN vs Transformer Encoding**

┌──────────────────────────────┐

│ CNN Encoder │

└──────────────────────────────┘

Input: [word1] [word2] [word3] [word4] ...

│ │ │ │

▼ ▼ ▼ ▼

┌──────────┐ ┌──────────┐ ┌──────────┐

│Embedding │ │Embedding │ │Embedding │ → vectors

└──────────┘ └──────────┘ └──────────┘

│ │ │ │

▼ ▼ ▼ ▼

┌────────────────────────────────────┐

│ 1D Convolutions (sliding filters)│

│ Capture local n-gram patterns │

└────────────────────────────────────┘

│

▼

┌───────────────────────┐

│ Max/Average Pooling │ → sentence vector

└───────────────────────┘

│

▼

[Final Encoding]

👉 CNNs learn local features (like n-grams). Long-range dependencies require stacking many layers or large filters.

┌──────────────────────────────┐

│ Transformer Encoder │

└──────────────────────────────┘

Input: [word1] [word2] [word3] [word4] ...

│ │ │ │

▼ ▼ ▼ ▼

┌──────────┐ ┌──────────┐ ┌──────────┐

│Embedding │ │Embedding │ │Embedding │ → vectors

└──────────┘ └──────────┘ └──────────┘

│ │ │ │

▼ ▼ ▼ ▼

┌───────────────────────────────────────┐

│ + Positional Encoding (add positions) │

└───────────────────────────────────────┘

│

▼

┌───────────────────────────────────────┐

│ Multi-Head Self-Attention │

│ Every token attends to every other │

└───────────────────────────────────────┘

│

▼

┌───────────────────────────────────────┐

│ Feed-Forward Networks (per token) │

└───────────────────────────────────────┘

│

▼

┌───────────────────────────────────────┐

│ Contextual Representations (matrix) │

└───────────────────────────────────────┘

│

▼

[Final Encoding]

👉 Transformers skip CNNs. They rely on:

* **Embedding table** + **Positional encoding** for input vectors.
* **Self-attention** for global context (any word ↔ any word directly).

✅ **Summary:**

* **CNN** = sliding filters, local n-gram features, then pool.
* **Transformer** = lookup embeddings + positional encoding + global self-attention.