Artificial Neural Network - Long Short-Term Memory

Sarit Maitra

Alliance University, Bengaluru, India

Biological Intelligence

Feature	Description
Incremental Processing	Processes sensory inputs and information in real time, step-by-step.
Internal Model	Maintains a constantly evolving mental representation of the environment and context.
Learning	Continuous and online, updates models from new experiences instantly or gradually.
Robustness	Adapts well to noisy, ambiguous, or novel situations.
Memory Integration	Seamlessly integrates long-term, short-term, and working memory.
Energy Efficiency	Extremely energy-efficient, using around 20 watts to power the brain.

Artificial Intelligence

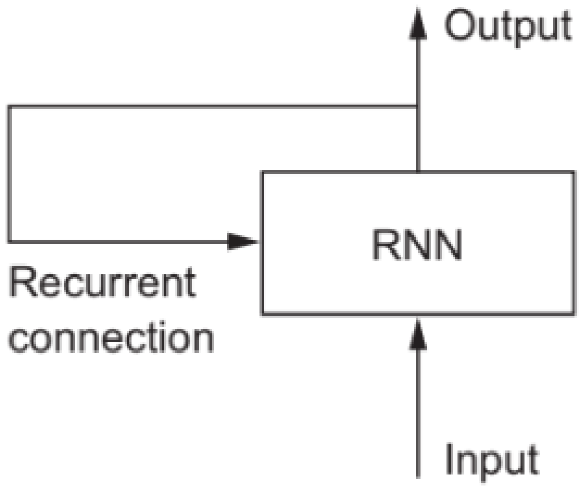
Feature	**Description**
Incremental Processing	Most models (like GPT or CNNs) process data in batches or fixed-length inputs. Real-time or streaming models are rarer and more complex.
Internal Model	Deep learning models have internal representations (weights, embeddings), but they are not always interpretable or dynamically updated on the fly.
Learning	Typically done in offline mode—a model is trained on a large dataset, and then deployed. To learn new things, it often requires retraining.
Robustness	Can struggle with out-of-distribution data, noise, or ambiguity unless specifically trained to handle it.
Memory Integration	Lacks human-like memory systems. Some efforts (e.g., retrieval-augmented models or memory networks) aim to imitate this.
Energy Efficiency	Computationally expensive, especially during training (e.g., large GPUs, data centers).

Key Differences

Aspect	Biological	Artificial
Learning mode	Online, adaptive	Mostly offline, retrain-heavy
Context awareness	Strong, integrated	Often limited or shallow
Memory handling	Rich, layered	Primitive, task-specific
Flexibility	General-purpose	Narrow/specialized (unless multi-modal)

RNN

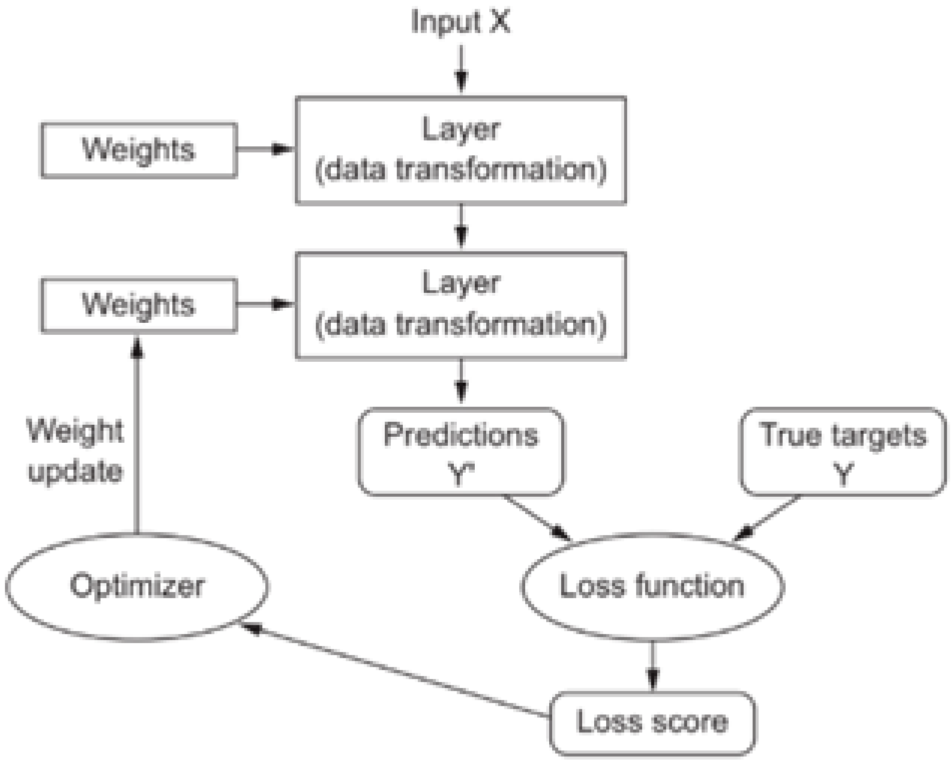
- RNN processes sequences by iterating through the sequence elements
- Maintaining a state containing information relative to what it has seen so far.
- RNN neural network has an internal loop.
- State of the RNN is reset between processing two different, independent sequences.
- The network internally loops over sequence elements.



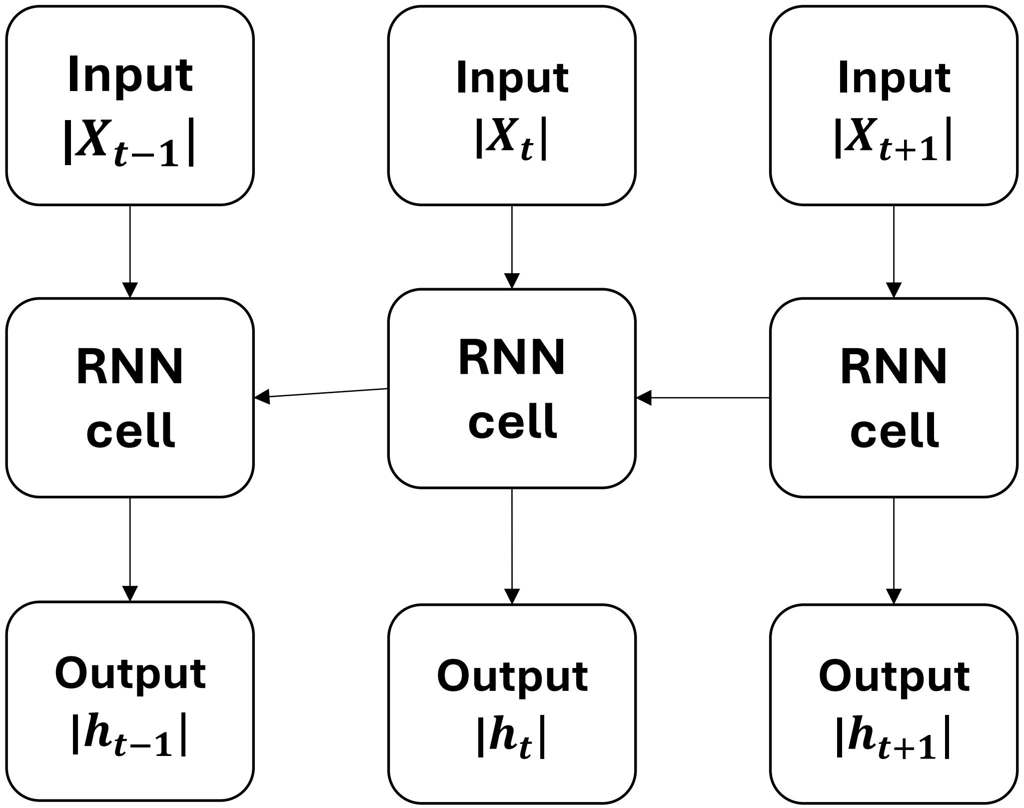
Anatomy of NN

Training loop

- Long Short-Term Memory (LSTM) algorithm fundamental was developed in 1997 and has barely changed since.
- NN revolves around the following objects:
  - Layers are combined into a model
  - The input data and corresponding targets
  - Loss function defines the feedback signal used for learning
  - Optimizer (SGD) determines how learning proceeds
- DNN involves multiple layers



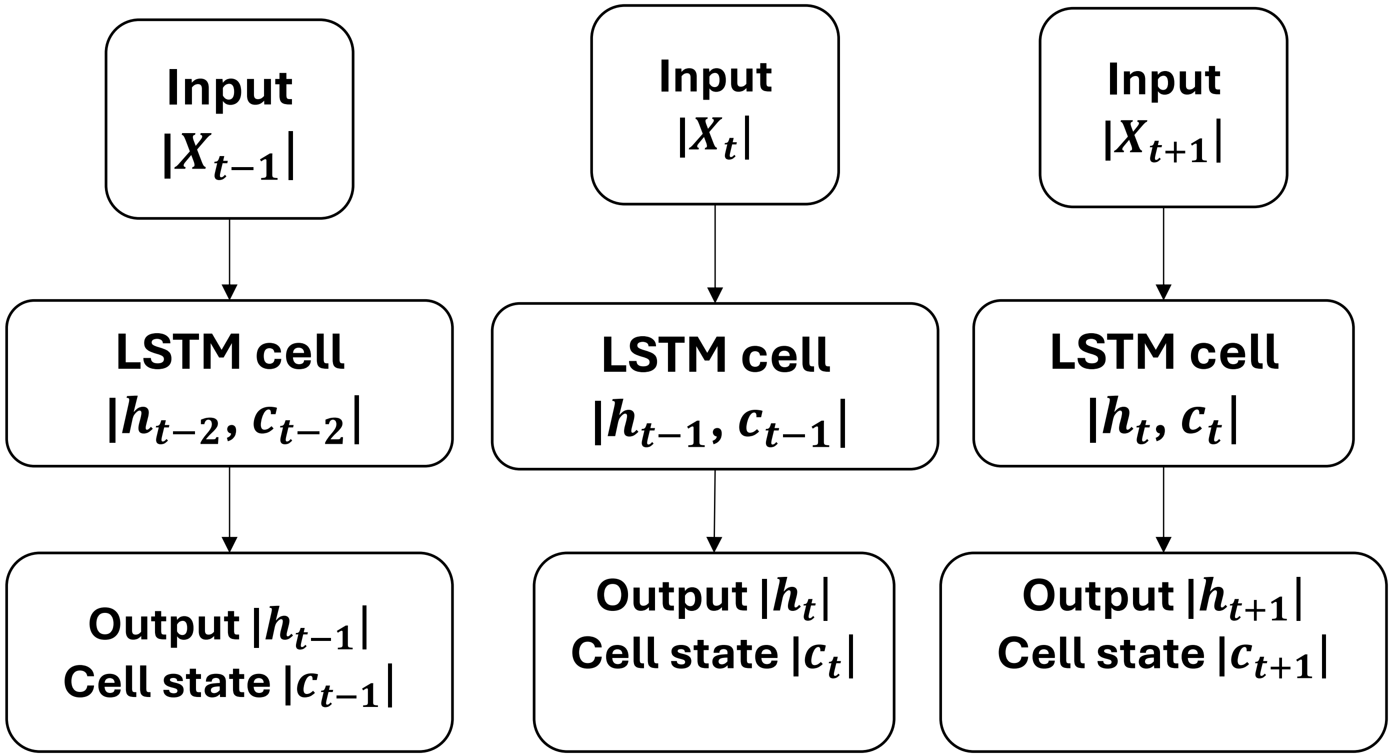
Simple RNN



- Each RNN Cell receives the current input (X_t) and previous hidden state (h_{t-1})
- Produces new hidden state (h_t)→ which can also be the output depending on the task
- Repeated over each time step

LSTM & GRU

- Simple RNN has a major issue
  - Theoretically retain information about inputs, in practice, such long-term dependencies are impossible to learn → vanishing gradient problem.
  - Hochreiter and Schmidhuber (1997) → LSTM



https://en.wikipedia.org/wiki/Long_short-term_memory#

~END~