# Long Short-Term Memory (LSTM) Networks

**What is LSTM?**

**LSTM (Long Short-Term Memory)** is a type of **Recurrent Neural Network (RNN)** used for **sequential data** — like time series, text, or audio. Unlike basic RNNs, LSTMs are good at remembering information over long sequences.

**Why LSTM?**

Regular RNNs struggle with "long-term dependencies", they forget earlier data too quickly. LSTM solves this using a special structure called a **cell state** and **gates**.

**2. LSTM Cell Structure**

An LSTM cell has **three gates** and a **cell state**:

- **Forget Gate**: Decides what to forget
- **Input Gate**: Decides what to store
- **Output Gate**: Decides what to output
- **Cell State**: Carries memory

## 🔷 3. Mathematical Formulations of LSTM

Let's define:

- $x_t$: input at time $t$
- $h_{t-1}$: previous hidden state
- $C_{t-1}$: previous cell state
- $W$, $b$: weight matrices and biases (different for each gate)
- $\sigma$: sigmoid activation
- $\tanh$: tanh activation

### Step-by-step Formulas:

**1. Forget Gate:**

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

**2. Input Gate:**

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

**3. Update Cell State:**

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

**4. Output Gate:**

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

---

## 🔷 4. Numerical Example (Simplified)

Let's walk through one time step. Suppose:

- $x_t = 0.5$
- $h_{t-1} = 0.1$
- $C_{t-1} = 0.2$
- All weights and biases are 1 for simplicity.

We concatenate $h_{t-1}$ and $x_t$:

$$[h_{t-1}, x_t] = [0.1, 0.5]$$

**Step 1: Forget Gate**

$$f_t = \sigma(1 * 0.1 + 1 * 0.5 + 1) = \sigma(1.6) \approx 0.832$$

**Step 2: Input Gate**

$$i_t = \sigma(1 * 0.1 + 1 * 0.5 + 1) = \sigma(1.6) \approx 0.832$$

$$\tilde{C}_t = \tanh(1 * 0.1 + 1 * 0.5 + 1) = \tanh(1.6) \approx 0.921$$

**Step 3: Cell State Update**

$$C_t = 0.832 * 0.2 + 0.832 * 0.921 \approx 0.166 + 0.766 \approx 0.932$$

**Step 4: Output Gate**

$$o_t = \sigma(1 * 0.1 + 1 * 0.5 + 1) = \sigma(1.6) \approx 0.832$$

$$h_t = 0.832 * \tanh(0.932) \approx 0.832 * 0.731 \approx 0.608$$

---

## 🔷 Summary of Results

| Variable | Value |
|---|---|
| $f_t$ | 0.832 |
| $i_t$ | 0.832 |
| $\tilde{C}_t$ | 0.921 |
| $C_t$ | 0.932 |
| $o_t$ | 0.832 |
| $h_t$ | 0.608 |

## 🔷 Key Insights

- **Cell state** $C_t$ keeps memory
- **Gates** help decide what to **keep**, **add**, or **output**
- LSTM is more stable than basic RNNs for long sequences