**به نام خدا**

**۱.۱ ساختار LSTM به زبان ساده**

**LSTM (Long Short-Term Memory)**  یک نوع **پیشرفته از شبکه‌های عصبی** **بازگشتی** (RNN) است که برای پردازش داده‌های **متوالی** (مانند متن، سری‌های زمانی، گفتار) طراحی شده. در اینجا به ساده‌ترین شکل توضیح می‌دهم:

### **مقایسه LSTM با RNN معمولی:**

| **ویژگی** | **RNN معمولی** | **LSTM** |
| --- | --- | --- |
| **حافظه** | فقط اطلاعات کوتاه‌مدت را حفظ می‌کند | هم حافظه کوتاه‌مدت و هم بلندمدت دارد |
| **مشکل Vanishing Gradient** | دارد | حل شده است |
| **ساختار** | ساده | دارای "دریچه" (Gates) برای کنترل جریان اطلاعات |

### **ساختار LSTM در یک نگاه:**

هر سلول LSTM سه دریچه دارد:

1. **دریچه فراموشی (Forget Gate)**: تصمیم می‌گیرد چه اطلاعاتی را دور بریزد.
2. **دریچه ورودی (Input Gate)**: تصمیم می‌گیرد چه اطلاعات جدیدی ذخیره شود.
3. **دریچه خروجی (Output Gate)**: تصمیم می‌گیرد چه اطلاعاتی به خروجی بفرستد.

### **💡 مثال کاربردی:**

**پیش‌بینی کلمه بعدی در جمله**:

* جمله: "امروز هوا آفتابی است و من می‌خواهم به \_\_\_\_ بروم"
* LSTM با توجه به **کل جمله** (نه فقط آخرین کلمه) پیش‌بینی می‌کند: "پارک"

### **پیاده‌سازی ساده:**

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import LSTM, Dense

model = Sequential([

LSTM(64, input\_shape=(10, 1)), *# 64 نورون، 10 گام زمانی، 1 ویژگی*

Dense(1) *# خروجی*

])

### ✅ **مزایای کلیدی**:

1. **حفظ اطلاعات بلندمدت** (مثلاً در ترجمه ماشینی).
2. **مقابله با مشکل Vanishing Gradient**.
3. **یادگیری وابستگی‌های پیچیده** در داده‌های متوالی.

### 📌 **کاربردهای رایج**:

* ترجمه ماشینی (Google Translate)
* پیش‌بینی قیمت سهام
* تولید متن (چت‌بات‌ها)
* تشخیص گفتار (Siri, Alexa)
* **مقایسه با RNN ساده:**
  + RNN: تنها یک حالت مخفی (h\_t)
  + LSTM: **سه گیت** (Input/Forget/Output) + **حافظه بلندمدت** (Cell State)

### **🎯 جمع‌بندی:**

LSTM مثل یک آدم با **حافظه خوب** است که:

* می‌داند چه چیزهایی را باید فراموش کند،
* چه چیزهای جدیدی را یاد بگیرد،
* و چه زمانی اطلاعات را استفاده کند!