

Redes LSTM: Memoria a Largo y Corto Plazo

Arquitectura y Funcionamiento Detallado

Presentación sobre Redes Neuronales Recurrentes

2025-10-01

¿Qué son las LSTM?

- **Extensión de las RNN** diseñadas para capturar dependencias temporales de secuencias largas
- Las RNN básicas **luchan por capturar** dependencias largas en el tiempo
- Introducen el concepto de “**celda de memoria**” que puede mantener, escribir o leer información
- Controladas por **estructuras llamadas puertas** que gestionan el flujo de información

Problemas que Resuelven las LSTM

Limitaciones de las RNN Básicas

- **Problema del gradiente de fuga**
- Dificultad para considerar **secuencias de entrada largas**
- **Pérdida de contexto** en dependencias temporales largas

Soluciones de las LSTM

- Las alteraciones en los LSTM **abordan el problema del gradiente de fuga**
- Permiten considerar **secuencias de entrada mucho más largas**
- **Preservan el contexto** de manera más efectiva

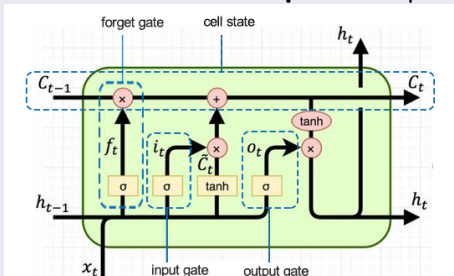
Arquitectura LSTM: Las Tres Puertas

Componentes Fundamentales

- 1 **Puerta de entrada** - Decide qué valores son importantes
- 2 **Puerta de olvido** - Descarta información innecesaria
- 3 **Puerta de salida** - Decide qué información pasar al siguiente paso

Características Comunes

- Consideran las **entradas del paso de tiempo anterior**
- Modifican la **memoria del modelo** y los **pesos de entrada**
- Utilizan **funciones de activación específicas** para cada puerta



Puerta de Olvido

Función Principal

Descarta **información que el modelo considera innecesaria** para tomar decisiones. Ecuación Matemática

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

Componentes

- W_f : **peso de la puerta del olvido**
- b_f : **sesgo de la puerta de olvido**
- σ : **función sigmoidea** (valores 0-1)

Interpretación

- **0**: Olvidar completamente
- **1**: Retener completamente
- Determina qué información del estado anterior conservar

Puerta de Entrada

Función Principal

Decide **qué valores son importantes** y deben transmitirse por el modelo.

Ecuaciones

Decidir qué actualizar:

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

Generar valores candidatos:

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

Funciones de Activación

- **Sigmoidea**: decide qué valores transmitir (0 = reducir, 1 = conservar)
- **TanH**: decide importancia de valores (-1 a 1)

Célula de Memoria Candidata

Función

Genera **nueva información potencial** para almacenar en el estado de la celda.

Ecuación

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

Características

- Genera **valores candidatos** para agregar al estado
- Función **tanh** asegura valores entre -1 y 1
- Representa **información potencial** para almacenar

Actualización del Estado de la Celda

Proceso de Actualización

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

Explicación del Proceso

- 1 Estado anterior (C_{t-1}) \times Puerta de olvido (f_t)
- 2 Valores candidatos (\tilde{C}_t) \times Puerta de entrada (i_t)
- 3 **Suma** para formar nuevo estado (C_t)

Interpretación

Combina información conservada con nueva información importante.

Puerta de Salida

Función Principal

Decide **qué valores pasar al siguiente paso de tiempo**.

Ecuación

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

Proceso de Decisión

- Analiza valores y asigna importancia (-1 a 1)
- Regula datos antes del siguiente cálculo
- Decide **salida final** del estado actual

Actualización del Estado Oculto

Ecuación Final

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

Funcionalidad

- Estado oculto se actualiza según estado de celda y puerta de salida
- Se usa como **salida para paso actual**
- Sirve como **entrada para siguiente paso**

Flujo Completo LSTM

Proceso Secuencial

- 1 **Puerta de Olvido:** Qué información conservar (sigmoidea 0-1)
- 2 **Puerta de Entrada:** Qué nueva información incorporar (sigmoidea + tanh)
- 3 **Actualización Estado:** $C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$
- 4 **Puerta de Salida:** Qué información emitir (sigmoidea)
- 5 **Estado Oculto:** $h_t = o_t \cdot \tanh(C_t)$

Aplicaciones Prácticas

Campos de Uso

- **Procesamiento de lenguaje natural**
- **Reconocimiento de voz**
- **Predicción de series temporales**
- **Sistemas de recomendación**
- **Análisis de datos secuenciales**

Beneficios Clave

- **Manejan dependencias largas** eficientemente
- **Resuelven problema del gradiente de fuga**
- **Preservan contexto** en secuencias largas
- **Flexibles** para diversos datos temporales

Impacto

Las LSTM representan **evolución significativa** sobre RNN tradicionales, permitiendo manejo efectivo de **dependencias temporales largas**.

¿Qué es una capa LSTM en el contexto de inundaciones?

Es una capa de red neuronal recurrente especializada en procesar **datos secuenciales temporales**:

- Series de tiempo de **niveles de ríos**
- Datos de **precipitación** acumulada
- **Humedad del suelo** histórica
- Variables meteorológicas temporales

Diseño con “puertas” le permite: - Recordar **patrones importantes** (picos de lluvia antecedentes) - Olvidar **información irrelevante** (días sin lluvia) - Predecir **eventos extremos** con horas/días de anticipación

Parámetros principales (en contexto de inundaciones)

Configuración clave para hidrología

units → Número de celdas LSTM. Más unidades capturan patrones complejos

Ejemplo: 50 unidades para predecir crecidas en una cuenca grande

return_sequences → Si es True, devuelve salidas en cada paso temporal
Útil para conectar otra LSTM después

return_state → Devuelve estados internos

Para modelos avanzados de atención o inicialización

activation → Función de activación

Por defecto tanh para estabilidad

input_shape → Forma de la entrada: (pasos_temporales, características)

Ejemplo: (30, 5) para 30 días con 5 variables meteorológicas

Ejemplo conceptual con datos reales

Secuencia de 7 días de datos hidrológicos

Entrada (7 días × 4 características):

Día	Precipitación	Nivel Río	Humedad Suelo	Temperatura
1	5mm	2.0m	60%	20°C
2	10mm	2.2m	70%	18°C
3	25mm	2.5m	80%	16°C
4	40mm	2.8m	85%	15°C
5	35mm	3.0m	90%	14°C
6	45mm	3.2m	92%	15°C
7	50mm	3.5m	95%	15°C

Procesamiento LSTM

Una LSTM con **units=3** procesará esta secuencia día a día, actualizando su “**memoria**” con la evolución de las variables.

Ejemplo interpretativo para predicción de inundaciones

Configuración del modelo

```
LSTM(units=10, return_sequences=False, input_shape=(30, 6))
```

a) Entrada

30 días históricos con 6 variables: - Precipitación - Nivel del río
- Humedad del suelo - Temperatura - Velocidad del viento - Presión atmosférica

b) Proceso

units=10 → La capa tiene 10 celdas LSTM para capturar patrones complejos

Ejemplo: correlaciones entre lluvia acumulada y subida del río

return_sequences=False → Solo devuelve la salida del último día

Para predecir inundación al día siguiente

Puertas de olvido descartan datos no relevantes

Ejemplo: días secos antiguos

Analogía intuitiva

Piensa en la LSTM como un **experto en hidrología** que analiza un informe diario:

Forget gate

Como cuando decide que una **lluvia leve de hace 20 días** ya no es relevante para el riesgo actual.

Input gate

Cuando subraya datos importantes: **“¡Lluvia de 100mm en 3 días!”**

Output gate Cuando emite un alerta gradual: **“Río subiendo 0.5m/día
→ riesgo en 48h”**

Evolución del análisis

- **Capas simples:** Variables individuales
- **Capas múltiples:** Interacciones complejas
- **Ejemplo avanzado:** “Suelo saturado + lluvia intensa = inundación repentina”

Mini-ejemplo numérico simplificado

Entrada de 3 días para predecir crecida

Día	Precipitación	Nivel Río	Normalizado
1	10mm	1.5m	[0.1, 0.15]
2	30mm	1.7m	[0.3, 0.17]
3	60mm	2.0m	[0.6, 0.2]

Procesamiento paso a paso

Estado inicial: $h = [0, 0]$, $c = [0, 0]$

Paso 1 (Día 1): - LSTM detecta lluvia moderada \rightarrow actualiza ligeramente su estado - $h = [0.02, 0.01]$, $c = [0.03, 0.02]$

Paso 2 (Día 2): - Lluvia alta \rightarrow puerta de entrada guarda esta información - $h = [0.15, 0.1]$, $c = [0.2, 0.12]$

Paso 3 (Día 3): - Lluvia extrema + nivel subiendo \rightarrow puerta de olvido mantiene todo el historial reciente - **Salida:** $h = [0.8, 0.6]$ (indica alto riesgo)

¿Por qué es útil para inundaciones?

Captura la no linealidad de las cuencas

Ejemplo: el suelo se satura después de X días de lluvia

Modela retrasos temporales

Ejemplo: la lluvia tarda horas en afectar el río aguas abajo

Puede integrar múltiples fuentes de datos

- Satélites
- Sensores terrestres
- Pronósticos meteorológicos

Usada en sistemas de alerta temprana

Como el **EFAS** (European Flood Awareness System)

Resumen de ventajas

Para predicción hidrológica

- **Memoria persistente** para eventos lejanos críticos
- **Adaptabilidad** a diferentes tipos de cuencas
- **Robustez** con datos imperfectos o incompletos

Para gestión de emergencias

- **Alertas tempranas** con mayor anticipación
- **Mejor precisión** en predicción de eventos extremos
- **Integración** de múltiples fuentes de información

Aplicaciones en sistemas reales

Casos de implementación

- **Sistemas de alerta temprana** nacionales y regionales
- **Monitoreo de cuencas** críticas
- **Predicción de crecidas** repentinas
- **Gestión de embalses** y recursos hídricos

Beneficios demostrados

- **Reducción de daños** económicos
- **Protección de vidas** humanas
- **Optimización** de recursos de emergencia
- **Mejora** en la toma de decisiones