Proyecto 1. Redes Neuronales Recurrentes: Predicción de variables meteorológicas usando una red neuronal LSTM

Luz Itzel Álvarez Cruz

Diplomado Ciencia de Datos (BUAP)
Abril 2025

1 Introducción

El presente trabajo tiene como objetivo predecir variables meteorológicas utilizando redes neuronales recurrentes tipo LSTM (Long Short-Term Memory). Se empleó un conjunto de datos meteorológicos provenientes de la ciudad de San Diego, con el fin de pronosticar temperatura, humedad, precipitación y velocidad del viento con un horizonte temporal de 24 horas.

2 Descripción del conjunto de datos

El conjunto de datos utilizado fue obtenido de Kaggle y está disponible en Weather Data – Kaggle Dataset. Este dataset contiene registros meteorológicos de varias ciudades, con variables registradas de manera horaria.

Las columnas disponibles incluyen:

- `Location`: nombre de la ciudad
- `Date_Time`: fecha y hora del registro
- `Temperature_C`: temperatura en grados Celsius
- `Humidity_pct`: humedad relativa en porcentaje
- `Precipitation_mm`: precipitación en milímetros
- `Wind_Speed_kmh`: velocidad del viento en km/h

Para este proyecto, se filtraron únicamente los datos correspondientes a la ciudad de San Diego, con el objetivo de construir un modelo específico por ciudad. Se seleccionaron las variables numéricas mencionadas para su predicción. El dataset tiene un formato de serie temporal multivariada, lo cual lo hace ideal para ser modelado mediante redes neuronales recurrentes (RNN), especialmente LSTM, que permiten capturar dependencias temporales en los datos. Se realizó una división del conjunto en entrenamiento y prueba, con una proporción del 80% y 20%, respectivamente.

- Tamaño del conjunto de entrenamiento: (79800, 24, 4)
- Tamaño del conjunto de prueba: (19950, 24, 4)

Cada muestra representa una secuencia de 24 pasos temporales (24 horas) con 4 variables, y se busca predecir esas mismas variables en el siguiente paso.

Variable	MAE	MSE	R^2
Temperatura (°C)	0.2510	0.0839	-0.0001
Humedad (%)	0.2491	0.0829	-0.0000
Precipitación (mm)	0.2506	0.0836	0.0000
Velocidad del viento	0.2498	0.0833	-0.0004
Promedio global	**0.2501**	**0.0834**	**-0.0001**

Table 1: Métricas de desempeño del modelo LSTM para cada variable meteorológica.

3 Preposesamiento

Las variables fueron normalizadas utilizando `MinMaxScaler` para mejorar el desempeño del modelo. Además, se definió una función para crear secuencias de entrada (X) y etiquetas (y), con una ventana deslizante de 24 pasos.

4 Construcción del modelo

Se implementó un modelo secuencial de Keras con la siguiente arquitectura:

Layer (**type**)	Output Shape	Param #
LSTM (64 unidades)	(None, 64)	17,664
Dropout (0.2)	(None, 64)	0
Dense (4 salidas)	(None, 4)	260

Total de parametros: 17,924

La capa LSTM permite capturar la dependencia temporal entre datos secuenciales, mientras que la capa densa produce una predicción para las cuatro variables meteorológicas.

5 Entrenamiento del modelo

Se entrenaron dos versiones del modelo:

- Versión 1: 10 épocas
- Versión 2: 20 épocas

Ambas con función de pérdida `mean_squared_error` y optimizador `adam`. Se usó una validación cruzada del 20% del conjunto de entrenamiento.

6 Evaluación del modelo

Se evaluó el modelo con el conjunto de prueba, utilizando las métricas: Error Absoluto Medio (MAE), Error Cuadrático Medio (MSE) y el coeficiente de determinación (R^2). Los resultados se presentan en la Tabla 1.

7 Resultados y análisis

A pesar de que el modelo logra converger con estabilidad durante el entrenamiento y presenta una pérdida constante en validación, los valores de R^2 cercanos a cero o negativos indican que el modelo no logra capturar adecuadamente la varianza de los datos. Esto sugiere que una red LSTM simple con una sola capa y sin ingeniería de características adicional puede no ser suficiente para modelar la complejidad de los datos meteorológicos.

8 Conclusiones

El experimento demuestra que es posible entrenar un modelo LSTM para predecir múltiples variables meteorológicas en paralelo. Sin embargo, para mejorar la precisión de las predicciones, se recomienda: Incrementar la complejidad del modelo (más capas LSTM o bidireccionales). Incorporar variables exógenas como hora del día, estación o condiciones pasadas. Realizar un análisis más profundo de las correlaciones entre variables.

9 Justificación del modelo

El uso de una RNN con capa LSTM se justifica por la naturaleza temporal de los datos. Las LSTM están diseñadas para manejar secuencias largas de datos gracias a su capacidad para recordar información durante varios pasos de tiempo, mitigando el problema del desvanecimiento del gradiente característico de las RNN tradicionales. Esto es especialmente útil en series meteorológicas, donde las condiciones actuales pueden depender de valores pasados.

El modelo fue optimizado usando el algoritmo Adam, debido a su capacidad para ajustar dinámicamente la tasa de aprendizaje durante el entrenamiento. Adam es una opción ampliamente utilizada por su eficiencia computacional y buen rendimiento en tareas de aprendizaje profundo.

10 Posibles mejoras

Aunque las métricas de error son bajas, el valor del R^2 Score cercano a cero indica que el modelo no supera una predicción trivial basada en la media. Esto sugiere que el modelo podría beneficiarse de: Incluir más capas LSTM o una arquitectura bidireccional. Ajustar la ventana temporal (mayor o menor a 24 pasos). Experimentar con normalizaciones específicas para cada variable. Incorporar variables externas (por ejemplo, día del año o fenómenos climáticos relevantes). Este análisis puede tomarse como una primera aproximación para evaluar la capacidad de las LSTM en tareas de predicción meteorológica.