

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**

**FACULTAD DE CIENCIAS E INGENIERÍA**



**DESARROLLO DE UN MODELO DE PREDICCIÓN  
ESPACIOTEMPORAL DE VARIABLES ATMOSFÉRICAS EN ZONAS  
DE GLACIARES USANDO REDES NEURONALES ARTIFICIALES**

**Tesis para obtener el título profesional de Ingeniero Informático**

**AUTOR:**

Christian Andre Carhuancho Rodriguez

**ASESOR:**

Dr. Edwin Rafael Villanueva Talavera


Lima, 2026

## Informe de Similitud

Yo, Edwin Rafael Villanueva Talavera, docente de la Facultad de Ciencias e Ingeniería de la Pontificia Universidad Católica del Perú, asesor del trabajo de investigación titulado *DESARROLLO DE UN MODELO DE PREDICCIÓN ESPACIOTEMPORAL DE VARIABLES ATMOSFÉRICAS EN ZONAS DE GLACIARES USANDO REDES NEURONALES ARTIFICIALES*, de Christian Andre Carhuancho Rodriguez, dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 21%. Así lo consigna el reporte de similitud emitido por el software *Turnitin* el 11 de abril de 2026.
- He revisado con detalle dicho reporte y la Tesis o Trabajo de Suficiencia Profesional, y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha: Lima, 12 de abril de 2026.

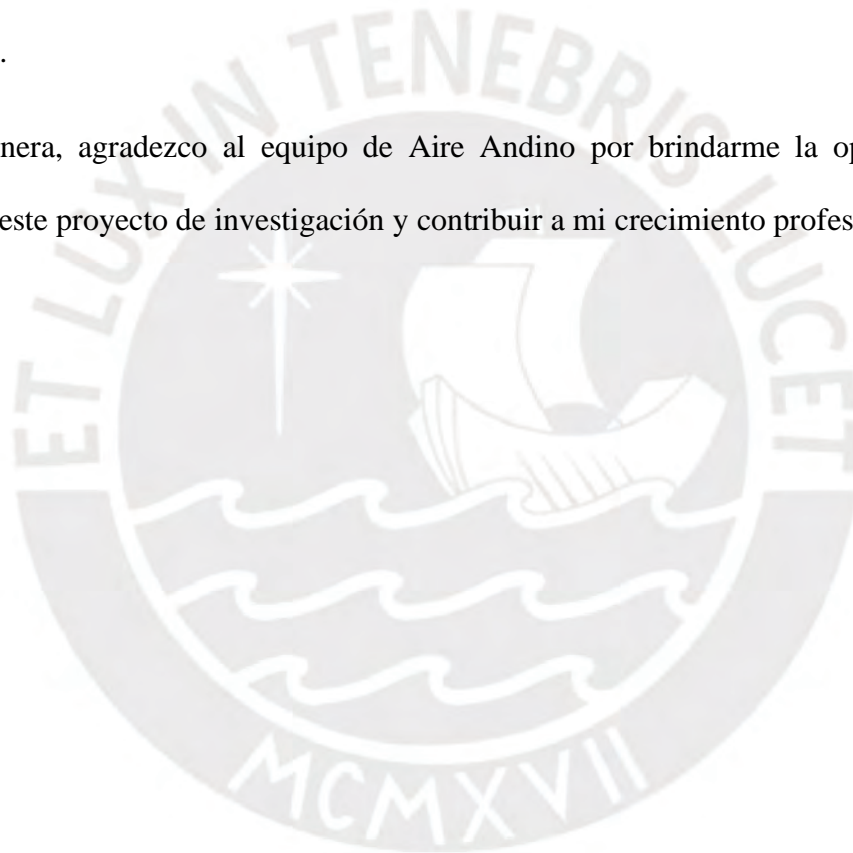
Apellidos y nombres del asesor / de la asesora: <u>Villanueva Talavera, Edwin Rafael</u>	
DNI: 29714308	Firma 
ORCID: 0000-0002-6540-1230	

## **Dedicatoria**

El presente proyecto de tesis está dedicado a mi familia, especialmente a mis padres, Ciro y Domnina, y a mi abuela Eugenia, quienes me han brindado su apoyo incondicional a lo largo de toda mi vida y durante mi formación universitaria. Gracias a su ejemplo, esfuerzo y valores, han sido una guía fundamental en la persona que aspiro a ser.

Asimismo, expreso mi sincero agradecimiento a mi asesor, Edwin Villanueva, por su orientación constante, paciencia y compromiso durante todo el proceso de desarrollo de esta investigación.

De igual manera, agradezco al equipo de Aire Andino por brindarme la oportunidad de participar en este proyecto de investigación y contribuir a mi crecimiento profesional.



## Resumen

La generación de datos climáticos de alta resolución en regiones con topografía compleja, como los Andes, representa un desafío significativo para la modelación meteorológica. Los modelos climáticos regionales tradicionales, como el Weather Research and Forecasting (WRF), enfrentan limitaciones relacionadas con altos costos computacionales y tiempos de procesamiento prolongados, especialmente cuando se requieren resoluciones espaciotemporales elevadas (menores a 27 km y horarias). Este estudio presenta una alternativa basada en inteligencia artificial para replicar las salidas del modelo WRF en la Cordillera Blanca, Perú, caracterizada por elevaciones que alcanzan los 6768 m sobre el nivel del mar y marcada variabilidad climática.

La investigación tiene como objetivo desarrollar y evaluar modelos de aprendizaje profundo capaces de generar datos atmosféricos de alta resolución (3 km) con menor costo computacional. Se fundamenta en la capacidad de las redes neuronales recurrentes convolucionales para capturar dependencias espaciotemporales complejas en datos meteorológicos, utilizando variables del Global Forecast System (GFS) y datos de elevación de alta resolución.

Se implementan y comparan dos arquitecturas: un modelo basado en capas ConvLSTM2D con doble entrada y un modelo encoder-decoder. Mediante experimentación numérica se evalúa el desempeño en la predicción de temperatura, humedad relativa, y componentes del viento.

Los resultados demuestran que el modelo ConvLSTM2D posee un desempeño superior, reduciendo significativamente el tiempo de inferencia y requisitos de memoria comparado con WRF, mientras que mantiene una precisión adecuada. El modelo desarrollado ofrece una herramienta práctica para pronóstico meteorológico en tiempo real en regiones montañosas, contribuyendo al avance de metodologías basadas en inteligencia artificial para el modelamiento climático regional.

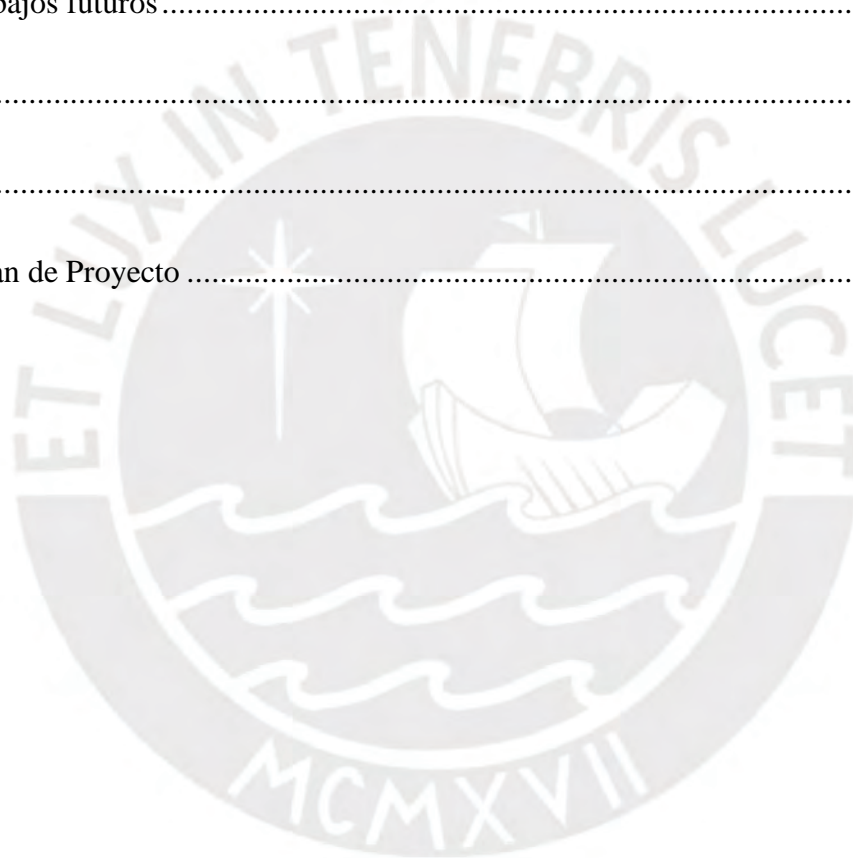
## Tabla de Contenido

Resumen .....	i
Tabla de Contenido .....	ii
Índice de Figuras.....	vi
Índice de Tablas .....	viii
Capítulo 1. Generalidades .....	1
1.1 Problemática .....	1
1.1.1 Árbol de Problemas.....	1
1.1.2 Descripción.....	2
1.1.3 Problema seleccionado.....	4
1.2 Objetivos .....	4
1.2.1 Objetivo general.....	4
1.2.2 Objetivos específicos .....	4
1.2.3 Resultados esperados .....	5
1.2.4 Mapeo de objetivos, resultados y verificación .....	6
1.3 Métodos y Procedimientos .....	7
1.3.1 WRF:.....	7
1.3.2 Python: .....	8
1.3.3 Tensorflow / Keras:.....	9
1.3.4 Github.....	9
1.3.5 Preprocesamiento de datos .....	10

1.3.6	Optimización de hiperparámetros:.....	11
1.3.7	Weights & Biases: .....	11
1.3.8	Matplotlib:.....	12
Capítulo 2. Marco Conceptual.....		12
2.1	Introducción.....	12
2.2	Conceptos asociados a las variables atmosféricas .....	13
2.2.1	Variables de estudio .....	13
2.3	Conceptos asociados al modelamiento de Variables Atmosféricas.....	14
2.3.1	Modelos explicativos: .....	14
2.3.2	Modelos predictivos:.....	14
Capítulo 3. Estado del Arte .....		17
3.1	Introducción.....	17
3.2	Objetivos de revisión .....	17
3.3	Preguntas de revisión .....	17
3.4	Estrategia de búsqueda.....	18
3.4.1	Motores de búsqueda a usar .....	18
3.4.2	Cadenas de búsqueda a usar .....	18
3.4.3	Documentos encontrados .....	20
3.4.4	Criterios de inclusión/exclusión .....	21
3.4.5	Documentos revisados .....	22
3.5	Formulario de extracción de datos.....	26

3.6	Resultados de la revisión.....	27
3.6.1	Respuesta a pregunta: ¿Qué modelos espacio temporales basados en aprendizaje profundo son utilizados para predecir variables atmosféricas y qué resultados, ventajas y limitantes se obtienen de cada uno?.....	27
3.6.2	Respuesta a pregunta: ¿En qué ubicación geográfica se han aplicado los modelos espacio temporales y cuál es el periodo que logran predecir?.....	30
3.6.3	Respuesta a pregunta: ¿Cuáles son las métricas utilizadas por los modelos para su evaluación?.....	31
3.6.4	Respuesta a pregunta: ¿Qué fuentes de datos son utilizadas por los modelos espacio temporales para realizar las predicciones?.....	31
3.6.5	Respuesta a pregunta: ¿Los modelos espacio temporales emplean de alguna forma modelos numéricos? ¿Cómo fueron utilizados en el estudio? .....	32
3.7	Conclusiones.....	33
Capítulo 4.	Presentación de los resultados esperados.....	34
R1.	Implementar el modelo WRF para la predicción de las variables de estudio.....	34
R2.	Conjunto de datos completo y estructurado .....	41
R3.	Revisión sistemática sobre arquitecturas de modelos a implementar .....	46
R4.	Código fuente de implementación del flujo de entrenamiento de dos modelos algorítmicos.....	52
R5.	Dos modelos algorítmicos entrenados.....	59
R6.	Estudio comparativo de evaluación de desempeño entre modelos algorítmicos.....	63

R7. Estrategia de evaluación numérica para comparar el modelo seleccionado en O2 y WRF.....	68
R8. Estudio comparativo del desempeño predictivo y tiempo de procesamiento entre los modelos algorítmicos y WRF.....	69
Capítulo 5. Conclusiones y trabajos futuros.....	76
5.1 Conclusiones.....	76
5.2 Trabajos futuros.....	78
Referencias .....	80
Anexos.....	89
Anexo A: Plan de Proyecto .....	91



## Índice de Figuras

Figura 1 Comparación entre neurona biológica y neurona artificial adaptado de (Neha, 2013) .....	16
Figura 2 Código de implementación de la instalación y configuración inicial del modelo numérico WRF en formato bash.....	36
Figura 3 Ejemplo de parámetros de configuración del modelo numérico WRF en el archivo namelist.input para ejecutar 24 horas de simulación para la fecha 14/09/2022 .....	40
Figura 4 Dominios de modelamiento WRF del área de estudio .....	41
Figura 5 Gráfico de elevación de terreno del área de estudio .....	44
Figura 6 Estructura de archivos del conjunto de datos para el entrenamiento de los modelos basados en aprendizaje profundo .....	45
Figura 7 Estructura de un elemento de entrada al modelo IA [elaboración propia] .....	48
Figura 8 Comparación entre salidas del modelo GFS y WRF para el dominio D2 y un mismo paso temporal.....	49
Figura 9 Estrategia de generación de muestras de entrenamiento con ventana deslizante con 3 puntos temporales de entrada (azul) y tres puntos de salida (rojo). [elaboración propia].....	50
Figura 10 Diagrama de estructura de capas convLSTM [elaboración propia].....	51
Figura 11 Diagrama de estructura de capas encoder-decoder [elaboración propia].....	51
Figura 12 Diagramas de caja del RMSE promedio para siete grupos de variables (G1–G7) evaluados con un modelo base Convolutacional Recurrente. ....	55
Figura 13 Código fuente de la definición de la estructura del modelo inicial de prueba de la arquitectura ConvLSTM en Python .....	57

Figura 14 Código fuente de la definición de la estructura del modelo inicial de prueba de la arquitectura encoder-decoder en Python .....	58
Figura 15 División regional del área de estudio en la Cordillera Blanca (R1–R3) según gradiente altitudinal y ubicación geográfica [elaboración propia] .....	60
Figura 16 Mapas de calor para un tiempo determinado para las predicciones de las arquitecturas Encoder-Decoder, ConvLSTM comparadas con WRF .....	63
Figura 17 Frecuencias de RMSE del mejor modelo por corrida, arquitectura y salida total ..	65
Figura 18 Mejores configuraciones de hiperparámetros del modelo CNN-LSTM obtenidas mediante la búsqueda aleatoria.....	70
Figura 19 Gráficos de dispersión del modelo ConvLSTM para el conjunto de prueba por variable y paso temporal.....	72
Figura 20 Mapas de calor de salidas y error absoluto del modelo ConvLSTM y WRF por variable de estudio y paso temporal.....	72

## Índice de Tablas

Tabla 1	Árbol de problemas .....	1
Tabla 2	Documentos encontrados por motor de búsqueda.....	20
Tabla 3	Documentos encontrados por motor de búsqueda luego de aplicar el filtro.....	21
Tabla 4	Documentos seleccionados para la revisión sistemática .....	22
Tabla 5	Campos del formulario de extracción.....	26
Tabla 6	Variables atmosféricas relevantes presentes en los datos del modelo GFS.....	45
Tabla 7	Grupos de variables atmosféricas utilizados para evaluar la relevancia en la predicción .....	53
Tabla 8	Parámetros de entrenamiento de arquitectura de modelos propuestos.....	60
Tabla 9	Resultados de entrenamiento del modelo con arquitectura Encoder-Decoder para los conjuntos de entrenamiento y validación .....	62
Tabla 10	Resultados de entrenamiento del modelo con arquitectura ConvLSTM para los conjuntos de entrenamiento y validación .....	62
Tabla 11	Valores de RMSE del mejor modelo por corrida, arquitectura y salida.....	64
Tabla 12	Espacio de búsqueda de hiperparámetros del modelo ConvLSTM .....	68
Tabla 13	Espacio de búsqueda de hiperparámetros del modelo ConvLSTM .....	70
Tabla 14	Resultados de entrenamiento del modelo con arquitectura ConvLSTM optimizado para los conjuntos de entrenamiento, validación y prueba.....	73
Tabla 15	Entorno computacional empleado para la corrida de experimentos.....	74
Tabla 16	Comparación de desempeño computacional entre el modelo IA optimizado y WRF .....	75

# Capítulo 1. Generalidades

## 1.1 Problemática

En el presente capítulo se describe la problemática del presente estudio de investigación empleando un árbol de problemas y se presenta el problema central a solucionar.

### 1.1.1 Árbol de Problemas

En la Tabla 1 se presenta el árbol de problemas, el cual contiene los problemas causa de la problemática, el problema central o principal del presente estudio y los problemas efectos que surgen a partir de la problemática.

Tabla 1 Árbol de problemas

<b>Problemas efectos</b>	Es difícil realizar estudios para conocer la distribución de variables atmosféricas en diferentes áreas geográficas, como zonas urbanas y zonas glaciares.	Al no conocer la información detallada de variables atmosféricas, no se tiene información precisa para tomar acciones de control ambiental efectivas.	
<b>Problema central</b>	Los modelos numéricos son actualmente las herramientas predilectas para el modelado espaciotemporal de variables atmosféricas dada su reconocida precisión; sin embargo, su aplicación suele ser problemática para estudios con alta resolución espacial o en grandes periodos de simulación debido a su elevada demanda de poder de cómputo.		
<b>Problemas causales</b>	No existe un cuerpo de datos públicamente disponible de simulaciones espaciotemporales de modelos numéricos que pueda servir como base para el entrenamiento	No existe un modelo algorítmico basado en aprendizaje profundo para emular las predicciones espaciotemporales generadas por modelos numéricos a tiempos de	No se conoce el desempeño predictivo y ahorro en costo computacional al que se puede llegar con modelos algorítmicos basados en aprendizaje profundo para la

	de modelos de aprendizaje profundo.	procesamiento reducidos.	predicción de variables atmosféricas.
--	-------------------------------------	--------------------------	---------------------------------------

### 1.1.2 Descripción

Los modelos climáticos sirven para una amplia variedad de propósitos en diversas disciplinas. En climatología y meteorología, son fundamentales para predecir eventos extremos peligrosos, cuya frecuencia e intensidad han aumentado significativamente en las últimas décadas y se proyecta que continúen en aumento en el futuro cercano (Seneviratne et al., 2023). En el sector de la salud, estas simulaciones son esenciales para mitigar los efectos adversos de los altos niveles de contaminación atmosférica (Cifuentes et al., 2021; López-Noreña et al., 2022; Spiridonov et al., 2019), así como para predecir los efectos de grandes olas de calor (Gupta et al., 2024) y evaluar la vulnerabilidad a la contaminación causada por incendios forestales (Gallo et al., 2023), entre otras aplicaciones.

En regiones vulnerables a eventos climáticos, como los Andes, los pronósticos de alta resolución son esenciales debido a la abrupta variabilidad del terreno en cortas distancias. Por ello, los modelos climáticos regionales (RCMs) se utilizan frecuentemente para generar datos detallados que mejoran la representación del clima en zonas montañosas (Doblas-Reyes et al., 2023; Gutierrez et al., 2024).

Aunque existen varios tipos de RCMs, una evaluación reciente realizada por Gutierrez et al. (2024) no encontró diferencias significativas en su desempeño. No obstante, el modelo WRF ha surgido como la opción preferida para generar datos espaciales y temporales de alta resolución. WRF ha sido ampliamente aplicado en regiones andinas, incluyendo Venezuela (Martinez et al., 2022), Colombia (Cifuentes et al., 2021), Ecuador (Chimborazo et al., 2022; Chimborazo & Vuille, 2021; Ochoa et al., 2014), Perú (Mourre et al., 2016; Potter et al., 2023; Saavedra et al., 2020), Bolivia (Mamani & Hendrick, 2021), Chile (Yáñez-Morrón et al., 2018) y Argentina (Schumacher et al., 2020). Estos estudios han demostrado consistentemente la

capacidad de WRF para capturar con precisión las condiciones climáticas de la región altoandina.

A pesar de su amplio uso y precisión reconocida, el modelo WRF enfrenta importantes desafíos cuando se aplica a casos reales. Generar datos meteorológicos con alta resolución espacial y temporal o para períodos de simulación prolongados requiere recursos computacionales sustanciales, lo que conlleva altos costos y tiempos de ejecución prolongados. Estas limitaciones destacan la necesidad de infraestructuras computacionales avanzadas y, en muchos casos, costosas (Chimborazo & Vuille, 2021; Spiridonov et al., 2019). Existe una estrecha relación entre la resolución espacial y los requerimientos computacionales. Por ejemplo, reducir el tamaño de la grilla de 24 km a 3 km incrementa el tiempo de procesamiento de WRF en un factor de 512 (Tie et al., 2010).

Ante la demanda del volumen de datos a procesar y estudiar en las diferentes áreas de la ciencia existentes, los modelos algorítmicos basados en aprendizaje profundo han cobrado gran relevancia en diversas disciplinas científicas debido a su excepcional capacidad para procesar grandes volúmenes de datos. Estas técnicas destacan por su habilidad para aprender automáticamente patrones complejos directamente a partir de datos sin procesar, minimizando la necesidad de intervención humana (Sharma et al., 2021). El aprendizaje profundo ha demostrado ser altamente efectivo en un amplio rango de aplicaciones, desde recomendaciones personalizadas en comercio electrónico hasta la solución de desafíos médicos complejos, incluyendo la gestión de la pandemia de COVID-19.

A pesar que un modelo de aprendizaje profundo para la predicción espaciotemporal de calidad del aire podría ofrecer tiempos de respuesta reducidos y requerir menores capacidades computacionales que un modelo numérico, la principal limitante para su construcción consiste en la falta de un cuerpo de datos con información de simulaciones realizadas por modelos numéricos que se puedan utilizar para entrenar al modelo de aprendizaje profundo y, ante la

falta de modelos de este tipo, no se conoce el desempeño predictivo que este puede tener ni el ahorro en costo computacional comparado con los modelos numéricos que se pueda lograr. Es por ello que el presente estudio busca resolver estos problemas.

### **1.1.3 Problema seleccionado**

El problema principal identificado puede resumirse de la siguiente manera: “Los modelos numéricos son actualmente las herramientas preferidas para el modelado espaciotemporal de variables atmosféricas debido a su reconocida precisión; sin embargo, su aplicación resulta limitada en estudios que requieren alta resolución espacial o largos periodos de simulación, debido a su elevada demanda de recursos computacionales.” Esta limitación, junto con los problemas mencionados en el apartado anterior, evidencia la necesidad de desarrollar enfoques alternativos. En este contexto, se plantea la construcción de un modelo de aprendizaje profundo que sea capaz de reproducir con alta fidelidad los resultados de los modelos numéricos, pero con un costo computacional considerablemente menor.

## **1.2 Objetivos**

### **1.2.1 Objetivo general**

Construir un modelo basado en aprendizaje profundo que permita predecir las variables atmosféricas de Temperatura (T), Humedad relativa (RH), componentes zonal (U) y meridional (V) del viento, con una precisión y desempeño comparable con los resultados obtenidos del modelo numérico WRF y con un menor costo computacional.

### **1.2.2 Objetivos específicos**

- O 1. Construir un conjunto de datos sobre simulaciones espaciotemporales del modelo numérico WRF sobre las variables de estudio (T, RH, U, V).
- O 2. Construir y comparar dos modelos algorítmicos basados en aprendizaje profundo capaces de emular las predicciones del modelo numérico WRF sobre

variables atmosféricas, con el fin de seleccionar el modelo con mejor desempeño predictivo.

- O 3. Evaluar el desempeño predictivo y costo computacional del modelo escogido basado en aprendizaje profundo con respecto al modelo numérico WRF.

### **1.2.3 Resultados esperados**

- O 1. Construir un conjunto de datos sobre simulaciones espaciotemporales del modelo numérico WRF sobre las variables de estudio (T, RH, U, V).
  - R 1. Implementar el modelo WRF para la predicción de las variables de estudio.
  - R 2. Conjunto de datos completo y estructurado.
- O 2. Construir y comparar dos modelos algorítmicos basados en aprendizaje profundo capaces de emular las predicciones del modelo numérico WRF sobre variables atmosféricas, con el fin de seleccionar el modelo con mejor desempeño predictivo.
  - R 3. Revisión sistemática sobre arquitecturas de modelos a implementar.
  - R 4. Código fuente de implementación del flujo de entrenamiento de dos modelos algorítmicos.
  - R 5. Dos modelos algorítmicos entrenados.
  - R 6. Estudio comparativo de evaluación de desempeño entre modelos algorítmicos.
- O 3. Evaluar el desempeño predictivo y costo computacional del modelo escogido basado en aprendizaje profundo con respecto al modelo numérico WRF.
  - R 7. Estrategia de evaluación numérica para comparar el modelo seleccionado en O2 y WRF.
  - R 8. Estudio comparativo del desempeño predictivo y tiempo de procesamiento entre el modelo algorítmico seleccionado y WRF.

### 1.2.4 Mapeo de objetivos, resultados y verificación

<b>Objetivo:</b> Construir un conjunto de datos sobre simulaciones espaciotemporales del modelo numérico WRF sobre las variables de estudio (T, RH, U, V).			
<b>Resultado</b>	<b>Medio de verificación</b>	<b>Indicador objetivamente verificable</b>	<b>Herramientas o métodos</b>
Implementar el modelo WRF para la predicción de las variables de estudio	- Reporte sobre instalación y configuración	- Aprobación de un experto en WRF	- WRF
Conjunto de datos completo y estructurado	- Reporte sobre obtención de datos y estadísticas descriptivas	- Aprobación del reporte por el asesor	- GitHub - Python - Matplotlib
<b>Objetivo:</b> Construir y comparar dos modelos algorítmicos basados en aprendizaje profundo capaces de emular las predicciones del modelo numérico WRF sobre variables atmosféricas, con el fin de seleccionar el modelo con mejor desempeño predictivo.			
<b>Resultado</b>	<b>Medio de verificación</b>	<b>Indicador objetivamente verificable</b>	<b>Herramientas o métodos</b>
Revisión sistemática sobre arquitecturas de modelos a implementar	- Reporte sobre elección de arquitecturas	- Aprobación del experto en inteligencia artificial	- Revisión sistemática
Código fuente de implementación del flujo de entrenamiento de dos modelos algorítmicos	- Repositorio en GitHub	- Dos modelos a implementar	- Python - Tensorflow Keras - GitHub - Preprocesamiento de datos
Dos modelos algorítmicos entrenados	- Métricas de evaluación de desempeño del entrenamiento y pruebas	- Métricas mejor que el valor medio obtenido de WRF	- Python - Tensorflow Keras - Fine Tuning - Weights & Biases

Estudio comparativo de evaluación de desempeño entre modelos algorítmicos	- Reporte del estudio comparativo	- Aprobación del experto en inteligencia artificial	- Python - Matplotlib - Seaborn - Estadísticas comparativas
<b>Objetivo:</b> Evaluar el desempeño predictivo y costo computacional del modelo escogido basado en aprendizaje profundo con respecto al modelo numérico WRF.			
<b>Resultado</b>	<b>Medio de verificación</b>	<b>Indicador objetivamente verificable</b>	<b>Herramientas o métodos</b>
Estrategia de evaluación numérica para comparar el modelo seleccionado en O2 y WRF	- Reporte de selección de estrategia de evaluación numérica	- Aprobación por el experto en inteligencia artificial	- Revisión sistemática
Estudio comparativo del desempeño predictivo y tiempo de procesamiento entre los modelos algorítmicos y WRF	- Reporte del estudio comparativo	- Aprobación por el experto en inteligencia artificial	- Python - Matplotlib - Seaborn - Estadísticas comparativas

### 1.3 Métodos y Procedimientos

A continuación, se presentan los métodos y procedimientos utilizados para la obtención de los resultados esperados del presente proyecto de fin de carrera.

#### 1.3.1 WRF:

WRF, abreviatura de Weather Research and Forecasting, es un modelo de simulación y predicción espaciotemporal atmosférico diseñado tanto para investigación como para aplicaciones operativas de pronóstico. Fue desarrollado mediante un esfuerzo colaborativo entre el Centro Nacional de Investigación Atmosférica (NCAR), la Administración Nacional Oceánica y Atmosférica (NOAA) y otras instituciones de investigación (Powers et al., 2017).

Desde su lanzamiento público inicial en el año 2000, el modelo WRF se ha convertido en el modelo atmosférico más utilizado del mundo. Esto se evidencia en métricas de usuarios registrados y publicaciones. El número acumulado de registros de WRF supera actualmente los 36,000, distribuidos en 162 países. La amplia aceptación de WRF se debe en parte a que se proporciona sin costo, sin restricciones de derechos de autor ni limitaciones para su modificación (Powers et al., 2017).

El diseño modular del modelo permite a los investigadores implementar y probar varios esquemas de física atmosférica, haciéndolo altamente adaptable para diversas aplicaciones, incluyendo pronóstico del tiempo, estudios de calidad del aire, investigación climática y modelado acoplado atmósfera-océano (Powers et al., 2017).

### **1.3.2 Python:**

Python es un lenguaje de programación de alto nivel y versátil que ha ganado una amplia popularidad en el campo de la computación y el deep learning (Python Organization, n.d.). Se caracteriza por su sintaxis clara y legible, lo que facilita su aprendizaje y comprensión, además de contar con una amplia comunidad de desarrolladores que ha creado una gran cantidad de bibliotecas y frameworks especializados que hacen de Python una opción poderosa para una variedad de aplicaciones (Python Organization, n.d.).

En el campo de la computación, Python se utiliza ampliamente para el desarrollo de aplicaciones de escritorio, aplicaciones web, análisis de datos y automatización de tareas (Python Organization, n.d.). Es conocido por su facilidad de uso y su capacidad para abordar una amplia gama de problemas gracias a las populares bibliotecas como NumPy, Pandas y Matplotlib que permiten el procesamiento y análisis eficiente de datos, mientras que frameworks como Django y Flask facilitan el desarrollo de aplicaciones web robustas (Python Organization, n.d.).

Python se ha convertido en el lenguaje preferido para muchas tareas relacionadas con el aprendizaje profundo (McKinney, 2017). Frameworks como TensorFlow y PyTorch, que son ampliamente utilizados en la investigación y la industria del aprendizaje máquina, tienen una interfaz de programación en Python (McKinney, 2017). Estas bibliotecas ofrecen una amplia gama de herramientas y funciones para la creación y entrenamiento de modelos de aprendizaje profundo, lo que ha contribuido a la popularidad de Python en este campo (McKinney, 2017).

### **1.3.3 Tensorflow / Keras:**

TensorFlow Keras es una biblioteca de aprendizaje profundo de código abierto y de alto nivel que se basa en el popular framework TensorFlow, además de proporcionar una interfaz de programación sencilla y flexible para diseñar, entrenar y evaluar modelos de aprendizaje profundo (Chollet, 2017).

Keras se caracteriza por su facilidad de uso y su enfoque en la modularidad y la extensibilidad, lo que permite a los desarrolladores construir redes neuronales de forma rápida y eficiente mediante la combinación de capas y módulos predefinidos, lo que facilita también la experimentación con diferentes arquitecturas de redes y algoritmos de optimización (Chollet, 2017).

Keras también ofrece herramientas para visualizar y evaluar el rendimiento de los modelos entrenados, lo cual facilita el proceso de entrenamiento y verificación de resultados empleando métricas de evaluación (Chollet, 2017).

### **1.3.4 Github**

GitHub es una plataforma de desarrollo colaborativo de código abierto que permite a los desarrolladores alojar, revisar y compartir proyectos de software (GitHub Inc., n.d.). Proporciona un sistema de control de versiones distribuido, lo que facilita el seguimiento de

los cambios realizados en los archivos de código fuente a lo largo del tiempo (GitHub Inc., n.d.).

GitHub juega un papel fundamental como un repositorio central de proyectos y bibliotecas de código abierto relacionados con el aprendizaje profundo. Los investigadores y desarrolladores utilizan GitHub para compartir y colaborar en el desarrollo de algoritmos, modelos pre-entrenados e implementaciones de redes neuronales.

### **1.3.5 Preprocesamiento de datos**

Son las técnicas aplicadas a los datos antes de ser utilizados para entrenar modelos de aprendizaje profundo y consiste en la manipulación y transformación de los datos de entrada con el objetivo de mejorar la calidad, la uniformidad y la representación adecuada para su uso en los algoritmos de aprendizaje profundo (Géron, 2019).

El preprocesamiento de datos en el aprendizaje profundo puede involucrar varias etapas y técnicas, como la limpieza y normalización de datos, la reducción de dimensionalidad, el equilibrio de clases, la codificación de variables categóricas y el particionamiento de datos en conjuntos de entrenamiento, validación y prueba (Géron, 2019).

Estas técnicas de preprocesamiento de datos son fundamentales para garantizar un rendimiento óptimo del modelo de aprendizaje profundo y ayudan a mitigar problemas como datos faltantes, valores atípicos, desequilibrio de clases y diferencias en las escalas de las variables, usualmente presentes en conjuntos de datos grandes y que pueden afectar negativamente la capacidad del modelo para aprender patrones y generalizar correctamente (Géron, 2019).

El preprocesamiento de datos también es esencial para mejorar la eficiencia computacional y reducir el tiempo de entrenamiento de los modelos de aprendizaje profundo, al reducir la complejidad y el ruido en los datos (Géron, 2019).

### **1.3.6 Optimización de hiperparámetros:**

La selección de hiperparámetros es una etapa fundamental en el desarrollo de modelos de aprendizaje profundo, ya que permite explorar de forma sistemática el espacio de configuraciones posibles con el objetivo de identificar aquellas que maximizan el desempeño predictivo del modelo (Bergstra & Bengio, 2012). Este proceso resulta especialmente relevante en arquitecturas de redes neuronales complejas, donde la interacción entre distintas capas o componentes puede influir considerablemente en la capacidad del modelo para generalizar (Feurer & Hutter, 2019).

Existen diversas estrategias para la búsqueda de hiperparámetros, siendo la búsqueda aleatoria (random search) una de las más eficientes en espacios de alta dimensionalidad. A diferencia de la búsqueda por malla (grid search), otra técnica comúnmente empleada, la búsqueda aleatoria permite explorar un mayor número de combinaciones con menos evaluaciones, lo que se traduce en una mejor cobertura del espacio de búsqueda en menor tiempo (Bergstra & Bengio, 2012).

### **1.3.7 Weights & Biases:**

Es una biblioteca de Python diseñada para ayudar en el seguimiento, la visualización y la colaboración en proyectos de aprendizaje automático y profundo mediante herramientas poderosas para el registro de experimentos, la visualización de métricas y la comparación de resultados (Weights & Biases, n.d.).

Weights & Biases permite a los desarrolladores y científicos de datos realizar un seguimiento detallado de los experimentos de aprendizaje automático, mediante sus funcionalidades como el registro y almacenamiento de información sobre los hiperparámetros, las métricas de rendimiento, los modelos entrenados y otros detalles relevantes, además de brindar la

posibilidad de organizar y etiquetar los experimentos, lo que facilita la gestión y la reproducción de los resultados (Weights & Biases, n.d.).

Una de las características clave de Weights & Biases es su capacidad para visualizar de manera interactiva los resultados de los experimentos gracias a que proporciona gráficos y visualizaciones intuitivas para comprender mejor el rendimiento del modelo, analizar tendencias y comparar diferentes ejecuciones (Weights & Biases, n.d.).

### **1.3.8 Matplotlib:**

Matplotlib es una biblioteca de Python ampliamente utilizada para crear visualizaciones gráficas de alta calidad, gracias a la amplia gama de herramientas y funciones para crear gráficos estáticos, gráficos interactivos, diagramas de dispersión, histogramas, gráficos de barras, gráficos de líneas y muchos otros tipos de visualizaciones que proporciona (Hunter, 2007).

Matplotlib es una herramienta esencial para la visualización de datos en el campo de la ciencia de datos, la ingeniería, la investigación y otras disciplinas que requieren análisis visual (Hunter, 2007). Además, es compatible con otras bibliotecas populares de Python, como NumPy y Pandas, lo que facilita su uso en combinación con operaciones de manipulación y análisis de datos (Hunter, 2007).

## **Capítulo 2. Marco Conceptual**

### **2.1 Introducción**

En el presente capítulo, se presenta un conjunto de definiciones sobre conceptos relevantes para el estudio. Se mencionan definiciones de conceptos relacionados a las variables atmosféricas elegidas para el estudio y de aprendizaje profundo, ya que en torno a estos conceptos se desarrolla el problema central planteado.

## 2.2 Conceptos asociados a las variables atmosféricas

### 2.2.1 Variables de estudio

El presente trabajo se enfoca en cuatro variables atmosféricas clave: temperatura (T), humedad relativa (RH), y los componentes zonal (U) y meridional (V) del viento. Cada una de estas variables desempeña un papel fundamental en la caracterización y predicción del comportamiento atmosférico, especialmente en regiones geográficamente complejas como los Andes.

La temperatura (T) representa la energía térmica de las moléculas de aire y es esencial para describir procesos meteorológicos como la formación de nubes, la convección y la distribución de precipitaciones. Además de ser una variable de entrada en modelos atmosféricos, también funciona como condición de frontera en simulaciones numéricas. La variación vertical de la temperatura con la altitud es un factor determinante de la estabilidad atmosférica y está estrechamente relacionada con fenómenos de calentamiento dependiente de la elevación en zonas montañosas, como los Andes (Chimborazo et al., 2022; Yarleque et al., 2018). Por ello, el seguimiento de su evolución temporal es crucial para comprender la variabilidad y el cambio climático (Gong et al., 2022; Johnstone & Sulungu, 2021; Tyagi et al., 2016).

La humedad relativa (RH) mide el contenido de vapor de agua en el aire y tiene un papel determinante en la formación de patrones meteorológicos, en los procesos de condensación, y en el desarrollo de precipitación y niebla. Su relación directa con la temperatura influye significativamente en la formación de nubes y en el balance energético de la atmósfera. Una modelación precisa de la RH no solo mejora la predicción del tiempo a corto plazo, sino que también permite analizar con mayor detalle las tendencias climáticas de largo plazo (Lovell-Smith et al., 2016; Tu & Lu, 2020).

Los componentes del viento, tanto zonal (U) como meridional (V), representan el movimiento del aire en los ejes este-oeste y norte-sur, respectivamente. Estos vectores permiten identificar patrones de circulación atmosférica que son fundamentales para comprender fenómenos meteorológicos locales y regionales, como las advecciones de masa de aire, frentes y sistemas convectivos. La predicción precisa de estos componentes ha mejorado gracias al uso de técnicas de aprendizaje automático, y su integración con el análisis de temperatura y humedad incrementa notablemente la calidad del modelado climático y meteorológico (Chu et al., 2022; Sri Preethaa et al., 2023).

## **2.3 Conceptos asociados al modelamiento de Variables Atmosféricas**

### **2.3.1 Modelos explicativos:**

Son modelos estadísticos y numéricos cuyo fin es probar teoría causal. Están conformados por un conjunto de leyes matemáticas, físicas y químicas que utilizan un conjunto de datos en forma de variables para probar hipótesis causales asociadas a otro conjunto de variables (Shmueli, 2010).

Un ejemplo de modelos explicativos es el modelo WRF, un modelo numérico que depende de data precisa de variables meteorológicas para realizar buenas predicciones; sin embargo, la calidad de predicciones en general es excelente con la limitante de requerir de alta capacidad computacional y tiene un elevado tiempo de ejecución (K. Zhang et al., 2023).

### **2.3.2 Modelos predictivos:**

Un modelo predictivo es una abstracción matemática utilizada para predecir nuevas o futuras observaciones. El objetivo generalmente es predecir el valor de una o un conjunto de variables de salida para un conjunto de nuevas observaciones en forma de variables de entrada (Shmueli, 2010).

Un ejemplo de modelo predictivo es el uso de redes de aprendizaje profundo para estimar el volumen de precipitación diaria. A partir de variables meteorológicas como temperatura, humedad relativa, presión atmosférica, velocidad del viento, entre otras, y considerando la fecha de la predicción, el modelo puede estimar valores como la precipitación mínima, máxima y promedio esperada para ese día.

- **Modelos predictivos espaciotemporales:**

Son modelos predictivos que realizan predicciones tanto temporales como espaciales, es decir, predicen un conjunto de variables para un determinado lapso de tiempo futuro en base a un conjunto de observaciones pasadas (B. Zhang et al., 2022) y, para el caso espacial, se realiza la predicción de un conjunto de variables para un área geográfica en un mismo tiempo dadas las observaciones de las mismas variables en otras ubicaciones geográficas (Stasch et al., 2014).

Un ejemplo de modelo predictivo espaciotemporal es el uso de modelos basados en aprendizaje profundo para la predicción de variables atmosféricas, como la temperatura, la humedad relativa o los componentes del viento, en una determinada región geográfica. Estos modelos utilizan datos históricos en forma de secuencias de imágenes o mapas de variables meteorológicas para aprender patrones espaciotemporales, lo que les permite generar predicciones en regiones cercanas o en el mismo lugar, pero en un periodo de tiempo futuro.

- **Redes neuronales artificiales**

Son sistemas computacionales inspirados en redes neuronales biológicas que consisten en un gran conjunto de procesadores simples denominados neuronas con muchas interconexiones entre ellos (Neha, 2013). Estas estructuras consisten en un conjunto de capas de neuronas donde se asignan pesos a cada neurona de una capa a manera de representar el flujo de información a

través de ellas y utilizan una función matemática que determina la activación de una neurona artificial y otra función que computa la salida de una neurona artificial (Neha, 2013).

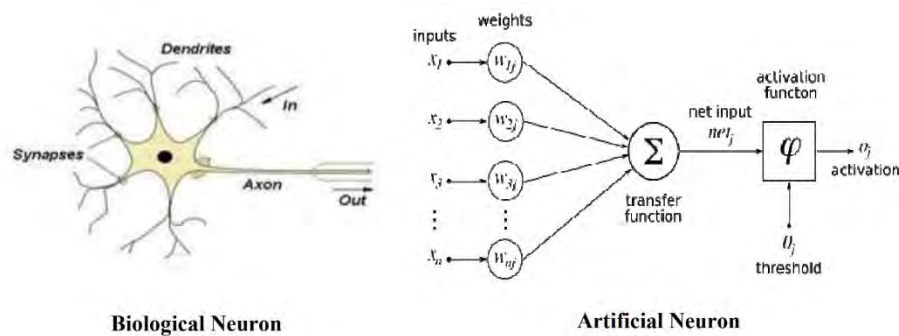


Figura 1 Comparación entre neurona biológica y neurona artificial adaptado de (Neha, 2013)

En la Figura 1 se puede apreciar la comparación directa entre una neurona biológica y una neurona artificial, donde en ambas existe un flujo de información que llega a la neurona y ella se encarga de procesar dicha información para devolver una salida vista que corresponde a nueva información que deberá ser procesada por la siguiente neurona. La diferencia entre una neurona artificial y una neurona biológica es el procesamiento mediante funciones matemáticas que se realiza para poder generar la salida. Las funciones matemáticas empleadas varían acorde al problema a resolver y a un conjunto de prácticas conocidas en el uso de estas.

El proceso en el cual se realiza el ajuste de los valores de pesos asignados se denomina entrenamiento y busca obtener la salida deseada, aunque para una cantidad de cientos de miles de neuronas resulta complicado, por lo que existen algoritmos que permiten ajustar los valores de los pesos cuyo objetivo es alcanzar el objetivo o salida deseada con el mínimo error posible (Neha, 2013).

El uso de redes neuronales artificiales se ha incrementado en los últimos 10 años debido al buen desempeño que tiene para el análisis de data en varias disciplinas como la medicina, sociología, seguridad de la información, big data, entre otros (Abiodun et al., 2018). Las

principales aplicaciones de redes neuronales se dan como solución a problemas de clasificación, clustering, reconocimiento de patrones y predicción (Abiodun et al., 2018).

## **Capítulo 3. Estado del Arte**

### **3.1 Introducción**

En el presente capítulo se presenta el estado del arte, el cual tiene como fin realizar la revisión sistemática mediante los objetivos de revisión, el planteamiento de las preguntas de revisión para conocer los principales modelos de aprendizaje profundo utilizados para el modelamiento espaciotemporal de variables meteorológicas y conocer la relación que estos tienen con los modelos numéricos.

### **3.2 Objetivos de revisión**

Se realiza la revisión sistemática con el objetivo de conocer los diferentes modelos espacio temporales y métricas utilizadas para sus respectivas evaluaciones en la predicción y simulación de variables atmosféricas en los diferentes espacios en los que fueron estudiados. Además, se busca conocer las fuentes de datos, como sensores o cámaras satelitales, la locación y contexto del estudio, y si estos utilizan o se comparan con otros modelos numéricos como WRF.

El tipo de revisión a realizar es del estado del arte.

### **3.3 Preguntas de revisión**

- P1. ¿Qué modelos espacio temporales basados en aprendizaje profundo son utilizados para predecir variables atmosféricas y qué resultados, ventajas y limitantes se obtienen de cada uno?
- P2. ¿En qué ubicación geográfica se han aplicado los modelos espacio temporales y cuál es el periodo que logran predecir?

- P3. ¿Cuáles son las métricas utilizadas por los modelos para su evaluación?
- P4. ¿Qué fuentes de datos son utilizadas por los modelos espacio temporales para realizar las predicciones?
- P5. ¿Los modelos espacio temporales emplean de alguna forma modelos numéricos?  
¿Cómo fueron utilizados en el estudio?

### **3.4 Estrategia de búsqueda**

Se realizará la búsqueda de documentos en motores de búsqueda provistos por los repositorios y bases de datos con acceso desde la Pontificia Universidad Católica del Perú.

#### **3.4.1 Motores de búsqueda a usar**

Se utilizan los motores de búsqueda Web of Science y Scopus, ya que ambos indexan material científico de reconocida calidad. El material utilizado corresponde a publicaciones en revistas y congresos internacionales.

#### **3.4.2 Cadenas de búsqueda a usar**

Se identifican los términos y palabras clave a utilizar para la revisión sistemática. Se realiza la siguiente clasificación:

- Términos relacionados a los modelos de aprendizaje profundo: Deep learning (incluye a redes neuronales en el término). spatio-temporal, spatiotemporal, sequence modeling.
- Términos relacionados a variables atmosféricas: Atmospheric variables, meteorological variables, weather variables, temperature, humidity, relative humidity, wind, zonal wind, meridional wind, wind components.
- Términos relacionados a modelos numéricos: WRF, numerical weather prediction, NWP, climate models, reanalysis data.

Se plantean dos cadenas de búsqueda para cada motor, la primera se centra en las cuatro primeras preguntas y la segunda relaja la búsqueda quitando el modelo espacio temporal para poder encontrar información relacionada a los modelos numéricos que pudieron ser utilizados en estudios de otros componentes del aire y responder a la quinta pregunta:

- Cadenas en Web of Science:

- TS=("deep learning" OR "neural network\*" OR "ConvLSTM" OR "LSTM")  
AND  
TS=("atmospheric variable\*" OR "meteorological variable\*" OR "weather variable\*" OR temperature OR "air temperature" OR "relative humidity" OR humidity OR "specific humidity" OR "zonal wind" OR "meridional wind" OR "wind components" OR "wind speed" OR "wind direction")  
AND  
TI=("spatio-temporal" OR spatiotemporal)
- TS=("deep learning" OR "neural network\*" OR "ConvLSTM" OR "LSTM")  
AND  
TS=("atmospheric variable\*" OR "meteorological variable\*" OR "weather variable\*" OR temperature OR "air temperature" OR "relative humidity" OR humidity OR "specific humidity" OR "zonal wind" OR "meridional wind" OR "wind components" OR "wind speed" OR "wind direction")  
AND  
TS=("WRF" OR "numerical weather prediction" OR "climate model\*" OR "reanalysis data")

- Cadenas en Scopus:

- (TITLE-ABS-KEY("deep learning" OR "neural network\*" OR "ConvLSTM" OR "LSTM"))

AND

(TITLE-ABS-KEY("atmospheric variable\*" OR "meteorological variable\*" OR "weather variable\*" OR temperature OR "air temperature" OR "relative humidity" OR humidity OR "specific humidity" OR "zonal wind" OR "meridional wind" OR "wind components" OR "wind speed" OR "wind direction"))

AND

(TITLE("spatio-temporal" OR spatiotemporal))

- (TITLE-ABS-KEY("deep learning" OR "neural network\*" OR "ConvLSTM" OR "LSTM"))

AND

(TITLE-ABS-KEY("atmospheric variable\*" OR "meteorological variable\*" OR "weather variable\*" OR temperature OR "air temperature" OR "relative humidity" OR humidity OR "specific humidity" OR "zonal wind" OR "meridional wind" OR "wind components" OR "wind speed" OR "wind direction"))

AND

(TITLE-ABS-KEY("WRF" OR "numerical weather prediction" OR "climate model\*" OR "reanalysis data"))

### 3.4.3 Documentos encontrados

Los resultados luego de consultar cada motor con cada cadena se muestran en la Tabla 2.

*Tabla 2 Documentos encontrados por motor de búsqueda*

Scopus		Web of Science	
C1	C2	C1	C2
384	751	128	217

Se puede ver que, en ambas búsquedas, el motor Scopus devuelve la mayor cantidad de resultados por cadena. Además, se puede notar que la segunda cadena devuelve una cantidad pequeña de resultados.

### 3.4.4 Criterios de inclusión/exclusión

Los artículos encontrados pasaron por un conjunto de criterios de inclusión y exclusión con el fin de considerar aquellos trabajos relevantes para responder las preguntas de revisión.

Los criterios de inclusión utilizados son los siguientes:

- El estudio detalla técnicas de aprendizaje profundo en la implementación de los modelos.
- El estudio pertenece a las áreas de ciencias de la computación e ingeniería y ciencias ambientales.
- El año de publicación del estudio es mayor al 2019

Los estudios excluidos cumplían con los siguientes criterios:

- El estudio no utiliza técnicas de aprendizaje profundo, debido a que no está relacionado con el tema de estudio.
- El estudio pertenece a un área diferente a ciencias de la computación e ingeniería y ciencias ambientales, por ejemplo, geología o telecomunicaciones.
- El estudio no responde a ninguna pregunta planteada para la revisión.

Luego de aplicar los criterios, se obtienen los siguientes resultados:

*Tabla 3 Documentos encontrados por motor de búsqueda luego de aplicar el filtro*

Scopus		Web of Science	
C1	C2	C1	C2
46	25	33	54

### 3.4.5 Documentos revisados

Luego de aplicar los criterios de inclusión y exclusión se obtuvieron 27 artículos para la revisión sistemática. Se presenta la Tabla 4 con los artículos revisados y su referencia en formato APA:

*Tabla 4 Documentos seleccionados para la revisión sistemática*

N° de artículo	Título del artículo	Referencia
1	Estimation of surface ozone concentration over Jiangsu province using a high-performance deep learning model	Mu, X., Wang, S., Jiang, P., & Wu, Y. (2023). Estimation of surface ozone concentration over jiangsu province using a high-performance deep learning model. <i>Journal of Environmental Sciences (China)</i> , 132, 122-133. doi:10.1016/j.jes.2022.09.032
2	Spatio-temporal air quality analysis and PM2.5 prediction over Hyderabad City, India using artificial intelligence techniques	Gokul, P. R., Mathew, A., Bhosale, A., & Nair, A. T. (2023). Spatio-temporal air quality analysis and PM2.5 prediction over hyderabad city, india using artificial intelligence techniques. <i>Ecological Informatics</i> , 76 doi:10.1016/j.ecoinf.2023.102067
3	Retrieval of hourly PM2.5 using top-of-atmosphere reflectance from geostationary ocean color imagers I and II	Choi, H., Park, S., Kang, Y., Im, J., & Song, S. (2023). Retrieval of hourly PM2.5 using top-of-atmosphere reflectance from geostationary ocean color imagers I and II. <i>Environmental Pollution</i> , 323 doi:10.1016/j.envpol.2023.121169
4	Dual-channel spatial-temporal difference graph neural network for PM 2.5 forecasting	Ouyang, X., Yang, Y., Zhang, Y., Zhou, W., & Guo, D. (2023). Dual-channel spatial-temporal difference graph neural network for PM 2.5 forecasting. <i>Neural Computing and Applications</i> , 35(10), 7475-7494. doi:10.1007/s00521-022-08036-0
5	Measuring and Modelling the Concentration of Vehicle-Related PM2.5 and PM10 Emissions Based on Neural Networks	Shepelev, V., Glushkov, A., Slobodin, I., & Cherkassov, Y. (2023). Measuring and modelling the concentration of vehicle-related PM2.5 and PM10 emissions based on neural networks. <i>Mathematics</i> , 11(5) doi:10.3390/math11051144

6	A hybrid deep learning model for regional O3 and NO2 concentrations prediction based on spatiotemporal dependencies in air quality monitoring network	Wu, C. -, He, H. -, Song, R. -, Zhu, X. -, Peng, Z. -, Fu, Q. -, & Pan, J. (2023). A hybrid deep learning model for regional O3 and NO2 concentrations prediction based on spatiotemporal dependencies in air quality monitoring network. <i>Environmental Pollution</i> , 320 doi:10.1016/j.envpol.2023.121075
7	Predicting hourly PM2.5 concentrations in wildfire-prone areas using a SpatioTemporal Transformer model	Yu, M., Masrur, A., & Blaszcak-Boxe, C. (2023). Predicting hourly PM2.5 concentrations in wildfire-prone areas using a SpatioTemporal transformer model. <i>Science of the Total Environment</i> , 860 doi:10.1016/j.scitotenv.2022.160446
8	Predicting spatiotemporal variations of PM2.5 concentrations during spring festival for county-level cities in China using VIIRS-DNB data	Erkin, N., Simayi, M., Ablat, X., Yahefu, P., & Maimaiti, B. (2023). Predicting spatiotemporal variations of PM2.5 concentrations during spring festival for county-level cities in china using VIIRS-DNB data. <i>Atmospheric Environment</i> , 294 doi:10.1016/j.atmosenv.2022.119484
9	Graph Neural Network for Air Quality Prediction: A Case Study in Madrid	Iskandaryan, D., Ramos, F., & Trilles, S. (2023). Graph neural network for air quality prediction: A case study in madrid. <i>IEEE Access</i> , 11, 2729-2742. doi:10.1109/ACCESS.2023.3234214
10	Multi-step forecast of PM2.5 and PM10 concentrations using convolutional neural network integrated with spatial-temporal attention and residual learning	Zhang, K., Yang, X., Cao, H., Thé, J., Tan, Z., & Yu, H. (2023). Multi-step forecast of PM2.5 and PM10 concentrations using convolutional neural network integrated with spatial-temporal attention and residual learning. <i>Environment International</i> , 171 doi:10.1016/j.envint.2022.107691
11	Estimation of daily ground-level PM2.5 concentrations over the Pearl River Delta using 1 km resolution MODIS AOD based on multi-feature BiLSTM	Miao, L., Tang, S., Ren, Y., Kwan, M. -, & Zhang, K. (2022). Estimation of daily ground-level PM2.5 concentrations over the pearl river delta using 1 km resolution MODIS AOD based on multi-feature BiLSTM. <i>Atmospheric Environment</i> , 290 doi:10.1016/j.atmosenv.2022.119362
12	RCL-Learning: ResNet and convolutional long short-term memory-based spatiotemporal air	Zhang, B., Zou, G., Qin, D., Ni, Q., Mao, H., & Li, M. (2022). RCL-learning: ResNet and convolutional long short-term memory-based spatiotemporal air pollutant concentration

	pollutant concentration prediction model	prediction model. Expert Systems with Applications, 207 doi:10.1016/j.eswa.2022.118017
13	Good environmental governance: Predicting PM2.5 by using Spatiotemporal Matrix Factorization generative adversarial network	Zhang, A., Chen, S., Zhao, F., & Dai, X. (2022). Good environmental governance: Predicting PM2.5 by using spatiotemporal matrix factorization generative adversarial network. Frontiers in Environmental Science, 10 doi:10.3389/fenvs.2022.981268
14	Spatiotemporal neural network for estimating surface NO2 concentrations over north China and their human health impact	Zhang, C., Liu, C., Li, B., Zhao, F., & Zhao, C. (2022). Spatiotemporal neural network for estimating surface NO2 concentrations over north china and their human health impact. Environmental Pollution, 307 doi:10.1016/j.envpol.2022.119510
15	An integrated 3D CNN-GRU deep learning method for short-term prediction of PM2.5 concentration in urban environment	Faraji, M., Nadi, S., Ghaffarpasand, O., Homayoni, S., & Downey, K. (2022). An integrated 3D CNN-GRU deep learning method for short-term prediction of PM2.5 concentration in urban environment. Science of the Total Environment, 834 doi:10.1016/j.scitotenv.2022.155324
16	Predicting PM2.5 atmospheric air pollution using deep learning with meteorological data and ground-based observations and remote-sensing satellite big data	Muthukumar, P., Cocom, E., Nagrecha, K., Comer, D., Burga, I., Taub, J., . . . Pourhomayoun, M. (2022). Predicting PM2.5 atmospheric air pollution using deep learning with meteorological data and ground-based observations and remote-sensing satellite big data. Air Quality, Atmosphere and Health, 15(7), 1221-1234. doi:10.1007/s11869-021-01126-3
17	Spatio-temporal prediction of soil hydro-thermal response in embankment to the varying climatic conditions using a Two-Step LSTM-ML approach	An, N., Xie, E., Yu, Y., Yang, Y., Lv, Q., Zhang, S., ... Wu, Y. (2025). Spatio-temporal prediction of soil hydro-thermal response in embankment to the varying climatic conditions using a Two-Step LSTM-ML approach. Transportation Geotechnics, 55(101648), 101648. doi:10.1016/j.trgeo.2025.101648
18	A novel deep learning approach for regional high-resolution spatio-	Resifi, S., Aawar, E. A., Dasari, H. P., Jebari, H., & Hoteit, I. (2025). A novel deep learning approach for regional high-resolution spatio-temporal wind

	temporal wind speed forecasting for energy applications	speed forecasting for energy applications. <i>Energy</i> (Oxford, England), 328(136356), 136356. doi:10.1016/j.energy.2025.136356
19	A spatiotemporal CNN-LSTM deep learning model for predicting soil temperature in diverse large-scale regional climates	Farhangmehr, V., Imanian, H., Mohammadian, A., Cobo, J. H., Shirkhani, H., & Payeur, P. (2025). A spatiotemporal CNN-LSTM deep learning model for predicting soil temperature in diverse large-scale regional climates. <i>The Science of the Total Environment</i> , 968(178901), 178901. doi:10.1016/j.scitotenv.2025.178901
20	FusionFireNet: A CNN-LSTM model for short-term wildfire hotspot prediction utilizing spatio-temporal datasets	Alizadeh, N., Mahdianpari, M., Hemmati, E., & Marjani, M. (2025). FusionFireNet: A CNN-LSTM model for short-term wildfire hotspot prediction utilizing spatio-temporal datasets. <i>Remote Sensing Applications Society and Environment</i> , 37(101436), 101436. doi:10.1016/j.rsase.2024.101436
21	MIESTC: A Multivariable Spatio-Temporal Model for Accurate Short-Term Wind Speed Forecasting	Li, S., Chen, M., Yi, L., Lu, Q., & Yang, H. (2025). MIESTC: A multivariable spatio-temporal model for accurate short-term wind speed forecasting. <i>Atmosphere</i> , 16(1), 67. doi:10.3390/atmos16010067
22	Estimation of daily XCO <sub>2</sub> at 1 km resolution in China using a spatiotemporal ResNet model	Wu, C., Yang, S., Jiao, D., Chen, Y., Yang, J., & Huang, B. (2024). Estimation of daily XCO <sub>2</sub> at 1 km resolution in China using a spatiotemporal ResNet model. <i>The Science of the Total Environment</i> , 954(176171), 176171. doi:10.1016/j.scitotenv.2024.176171
23	Estimating 1-km PM <sub>2.5</sub> concentrations based on a novel spatiotemporal parallel network STMSPNet in the Beijing-Tianjin-Hebei region	Zeng, Q., Li, M., Fan, M., Tao, J., Chen, L., Zhang, Y., ... Zhu, Y. (2024). Estimating 1-km PM <sub>2.5</sub> concentrations based on a novel spatiotemporal parallel network STMSPNet in the Beijing-Tianjin-Hebei region. <i>Atmospheric Environment</i> (Oxford, England: 1994), 338(120796), 120796. doi:10.1016/j.atmosenv.2024.120796
24	Forecasting sea surface temperature during typhoon events in the Bohai	He, H., Shi, B., Hao, Y., Feng, L., Lyu, X., & Ling, Z. (2024). Forecasting sea surface temperature during typhoon events in the Bohai Sea using

	Sea using spatiotemporal neural networks	spatiotemporal neural networks. Atmospheric Research, 309(107578), 107578. doi:10.1016/j.atmosres.2024.107578
25	Spatiotemporal Evolution Analysis of PM2.5 Concentrations in Central China Using the Random Forest Algorithm	Fang, G., Zhu, Y., & Zhang, J. (2024). Spatiotemporal evolution analysis of PM2.5 concentrations in Central China using the random forest algorithm. Sustainability, 16(19), 8613. doi:10.3390/su16198613
26	Residual Spatiotemporal Convolutional Neural Network Based on Multisource Fusion Data for Approaching Precipitation Forecasting	Zhang, T., Wang, D., Huang, L., Chen, Y., & Li, E. (2024). Residual spatiotemporal convolutional neural network based on multisource fusion data for approaching precipitation forecasting. Atmosphere, 15(6), 628. doi:10.3390/atmos15060628
27	A novel dynamic spatio-temporal graph convolutional network for wind speed interval prediction	Chen, Z., Zhang, B., Du, C., Meng, W., & Meng, A. (2024). A novel dynamic spatio-temporal graph convolutional network for wind speed interval prediction. Energy (Oxford, England), 294(130930), 130930. doi:10.1016/j.energy.2024.130930

### 3.5 Formulario de extracción de datos

Se plantea la siguiente tabla con los campos del formulario de extracción de información para los artículos obtenidos luego de realizar las búsquedas:

*Tabla 5 Campos del formulario de extracción*

Campo	Descripción	Pregunta
N° de artículo	Número identificador del artículo	General
Año de publicación	Año de publicación del artículo	General
Título del artículo	Título del artículo	General
Autores	Autor o autores del artículo	General
Nombre de la fuente	Nombre de la revista, congreso, libro o tesis del artículo	General
Resumen	Breve resumen del artículo	General

Modelos aplicados	¿Qué modelos basados en aprendizaje profundo se han aplicado para realizar las predicciones?	P1
Limitaciones existentes	¿Cuáles fueron las principales limitaciones al momento de aplicar los modelos elegidos?	P1
Ventajas encontradas	¿Cuáles fueron las principales ventajas de aplicar los modelos elegidos?	P1
Resultados obtenidos	¿Cuáles fueron los resultados obtenidos del estudio? ¿Fueron positivos o negativos?	P1
Ubicación geográfica del estudio	País o región en donde se realizó el estudio	P2
Periodo de predicción	¿Cuál es el periodo de predicción del modelo? Por ejemplo: Un día, una semana, un mes, etc	P2
Métricas utilizadas	¿Cuáles fueron las métricas utilizadas para evaluar el desempeño de predicción de los modelos elegidos?	P3
Fuente de datos del estudio	Fuentes de datos donde se obtuvieron los datos para el análisis. Por ejemplo: Sensores, cámaras satelitales.	P4
¿Utiliza un modelo numérico? ¿Cuál y cómo?	¿Utiliza o compara los resultados de predicción obtenidos con un modelo numérico como WRF?	P5
¿Presenta una implementación en código?	Conocer si el artículo presenta código de implementación disponible en algún repositorio.	General
Comentarios	Comentarios personales sobre el artículo revisado.	General

### 3.6 Resultados de la revisión

#### 3.6.1 Respuesta a pregunta: ¿Qué modelos espacio temporales basados en aprendizaje profundo son utilizados para predecir variables atmosféricas y qué resultados, ventajas y limitantes se obtienen de cada uno?

De los veintisiete trabajos revisados, veinte corresponden a modelos híbridos y siete son de arquitectura única o comparaciones entre múltiples modelos. Los modelos híbridos propuestos están compuestos predominantemente por una combinación de dos arquitecturas de aprendizaje profundo, donde destaca el modelo propuesto por (Wu et al., 2023) que utiliza una combinación

de tres arquitecturas (ResNet, GCN y BiLSTM), y el estudio de An et al. (2025) que implementa un enfoque secuencial de dos pasos combinando LSTM con tres algoritmos de aprendizaje automático (SVR, RF y ANN). Entre los estudios comparativos destaca (Gokul et al., 2023) que evalúan cinco algoritmos de aprendizaje máquina (regresión multilineal, árboles de decisión, K-Nearest Neighbors, Random Forest y Extreme Gradient Boost) junto con un modelo de aprendizaje profundo LSTM, mientras que (Fang et al., 2024) comparan tres enfoques (MLR, BPNN y RF) para análisis de PM2.5.

Entre las arquitecturas de modelos de aprendizaje profundo encontradas en la revisión, se observa que la más utilizada es la arquitectura LSTM, considerada como una red recurrente RNN, apareciendo en doce estudios ya sea en forma pura o como componente de modelos híbridos (Alizadeh et al., 2025; An et al., 2025; Chen et al., 2024; Farhangmehr et al., 2025; Gokul et al., 2023; Miao et al., 2022; Mu et al., 2023; Muthukumar et al., 2022; Resifi et al., 2025; Wu et al., 2023; B. Zhang et al., 2022; T. Zhang et al., 2024). Le siguen en frecuencia las redes basadas en grafos GCN (Graph Convolutional Networks), presentes en seis estudios (Chen et al., 2024; Iskandaryan et al., 2023; Muthukumar et al., 2022; Ouyang et al., 2023; Wu et al., 2023; A. Zhang et al., 2022), y finalmente las arquitecturas basadas en atención como GRU, también considerada una RNN, utilizada en cuatro estudios (Chen et al., 2024; Faraji et al., 2022; Iskandaryan et al., 2023; Li et al., 2025). Las arquitecturas convolucionales CNN aparecen en ocho estudios (Alizadeh et al., 2025; Faraji et al., 2022; Farhangmehr et al., 2025; Muthukumar et al., 2022; Wu et al., 2024; Zeng et al., 2024; K. Zhang et al., 2023; T. Zhang et al., 2024), frecuentemente combinadas con LSTM para aprovechar las fortalezas complementarias de extracción de características espaciales y modelamiento temporal.

La elección de arquitecturas RNN se debe a que modelos de este tipo permiten representar de una manera efectiva las dinámicas espacio temporales y tienen la ventaja de un buen aprendizaje en base a eventos pasados y patrones de corto plazo, así como la incorporación de

atención que permite discriminar data que no es necesaria o no debe ser recordada (Muthukumar et al., 2022). Lamentablemente, esta arquitectura sufre de la limitación de necesitar data con continuidad temporal, es decir, que los registros utilizados sean consecuentes en un tiempo determinado (Farhangmehr et al., 2025; Miao et al., 2022).

Los enfoques híbridos CNN-LSTM demostraron resultados sobresalientes al combinar extracción de características espaciales mediante CNN con modelamiento temporal mediante LSTM. Estos modelos alcanzaron correlaciones de entrenamiento del 99.18% al 99.69% y correlaciones de prueba del 93.72% al 99.24% (Farhangmehr et al., 2025), superando significativamente a algoritmos tradicionales como RF (NRMSE 6.93%-9.14%) y SVR (NRMSE 7.48%-9.69%).

Las arquitecturas Transformer y mecanismos de atención, aunque menos utilizadas, obtienen resultados excepcionales y representan tendencias emergentes. Los modelos ST-Transformer alcanzaron RMSE de 6.92 con capacidad de aplicación a múltiples problemas (calidad de agua, precipitaciones, radiación solar) (Yu et al., 2023). Sistemas basados en AFNO (Adaptive Fourier Neural Operator) con Vision Transformers exploraron tres enfoques (recursivo, downscaling, combinado), donde el enfoque combinado demostró ser el más preciso y computacionalmente eficiente (1.61s inferencia) con MAE de 0.4 a 0.71 m/s en 48h manteniendo  $R^2$  de 0.95 a 0.88, mientras que el enfoque recursivo puro sufrió errores relativos de hasta 40% (Resifi et al., 2025). Modelos con codificación independiente multivariable y correlación espaciotemporal (MIESTC) lograron mejoras del 14.60%, 8.64% y 10.41% comparado con ConvLSTM, UNET y SimVP, evitando introducción de ruido por diferencias entre variables (Li et al., 2025). Modelos con mecanismos de atención de bloque convolucional (ACFN) superaron consistentemente a ConvLSTM y PredRNN con tiempos de cálculo menores a 1 segundo versus modelos oceánicos numéricos más intensivos computacionalmente (He et al., 2024).

### **3.6.2 Respuesta a pregunta: ¿En qué ubicación geográfica se han aplicado los modelos espacio temporales y cuál es el periodo que logran predecir?**

De los veintisiete trabajos revisados, China representa la ubicación geográfica más frecuente con diecisiete estudios (63%), seguida por América del Norte con cinco estudios (18.5%), Europa con dos estudios (7.4%), y finalmente Medio Oriente y regiones marítimas con un estudio cada uno (3.7%). Esta concentración en territorio asiático refleja la urgencia de abordar problemas de contaminación atmosférica como la disponibilidad de extensas redes de monitoreo y datos satelitales de alta resolución en la región empleadas como fuentes de datos para el entrenamiento y/o validación de los modelos propuestos.

En cuanto al tipo de ubicación, las zonas urbanas dominan con un total de veinte estudios (74%), abarcando desde ciudades individuales como Hyderabad, Teherán y Madrid, hasta megaciudades y aglomeraciones urbanas como Shanghai, Los Ángeles y la región Beijing-Tianjin-Hebei. Las zonas suburbanas y rurales están representadas en tres estudios (11%), enfocándose principalmente en el norte de China y regiones agrícolas de China Central. Las aplicaciones en ambientes marinos aparecen en dos estudios (7.4%), mientras que regiones con topografía mixta (montañas, costas, planicies) están presentes en cuatro estudios (15%). Notablemente, ningún estudio revisado se enfocó específicamente en glaciares.

Los periodos de predicción muestran una clara tendencia hacia horizontes de corto plazo. Las predicciones horarias constituyen la modalidad más frecuente con dieciséis estudios (59%), siendo el horizonte temporal predominante para aplicaciones de calidad del aire y gestión operacional. Las predicciones diarias representan el segundo grupo más común con siete estudios (26%), utilizadas principalmente para análisis de contaminantes de acumulación lenta y evaluación de tendencias estacionales.

### **3.6.3 Respuesta a pregunta: ¿Cuáles son las métricas utilizadas por los modelos para su evaluación?**

Entre las métricas de evaluación de modelos más usadas en los estudios revisados, se encuentran cuatro:  $R^2$ , MAE, RMSE y MSE. Además, se utilizan múltiples métricas para un solo estudio, por lo que es importante mencionar que las comparaciones que se realizan son empleando todas las métricas utilizadas.

$R^2$  es el coeficiente de determinación e indica qué tan bueno fue el entrenamiento comparando las variables predichas con las variables de prueba. En la mayoría de casos, es un indicador confiable de rendimiento de un modelo.

En cuanto a MAE, es el error absoluto medio de un modelo para un conjunto de prueba y se calcula como el promedio del error individual de cada predicción.

La métrica RMSE es la raíz cuadrada del error cuadrático medio y sirve para realizar una comparación de término a término de la diferencia entre los valores estimados y reales. Además, provee información del desempeño a corto plazo de un modelo.

Finalmente, la métrica MSE corresponde al error cuadrático medio y sirve para saber qué tan cerca están los puntos de una línea de regresión. En conjunto con la métrica RMSE permiten tener una idea de la calidad de predicción del modelo.

### **3.6.4 Respuesta a pregunta: ¿Qué fuentes de datos son utilizadas por los modelos espacio temporales para realizar las predicciones?**

Las fuentes de datos utilizadas por los modelos revisados se clasifican en cuatro categorías: estaciones de monitoreo terrestre (20 estudios, 74%), observaciones satelitales (18 estudios, 67%), datos de reanálisis (13 estudios, 48%) y productos de modelos numéricos (5 estudios, 18.5%). La mayoría de los estudios (59%) emplean combinaciones de múltiples fuentes para aprovechar las fortalezas complementarias de cada tipo de dato.

Las redes de estaciones de monitoreo constituyen la fuente más frecuente, proporcionando datos de alta precisión temporal. Las estaciones de calidad del aire aparecen en trece estudios, variando desde configuraciones densas como 6,880 estaciones hasta redes focalizadas de 24-72 estaciones. Las estaciones meteorológicas están presentes en doce estudios, proporcionando variables como temperatura, precipitación, humedad, velocidad del viento y presión atmosférica con frecuencias desde 30 minutos hasta horarias. Estaciones especializadas incluyen sensores de suelo a múltiples profundidades, estaciones de velocidad del viento clasificadas según rugosidad del terreno, y sensores móviles de tráfico integrados con calidad del aire.

Los sensores satelitales proporcionan cobertura espacial amplia con resoluciones desde 500m hasta varios kilómetros. El satélite Sentinel-5 Precursor es el más utilizado (tres estudios) para predicción de O<sub>3</sub> y NO<sub>2</sub> con resoluciones de 1-1.4 km<sup>2</sup>. Los sensores MODIS aparecen en cinco estudios con productos de AOD en resoluciones de 500m a 1 km, frecuentemente validados con datos AERONET.

Los productos ERA5 del ECMWF constituyen la fuente de reanálisis predominante, apareciendo en diez estudios con resoluciones desde 0.25°×0.25° hasta 0.5°×0.625° y frecuencia horaria. Los datos incluyen variables meteorológicas en superficie y múltiples niveles verticales (500/850/900/1000 hPa) como temperatura, geopotencial, humedad específica y componentes U y V del viento.

### **3.6.5 Respuesta a pregunta: ¿Los modelos espacio temporales emplean de alguna forma modelos numéricos? ¿Cómo fueron utilizados en el estudio?**

De los estudios revisados, únicamente seis (22%) reportaron el uso de modelos numéricos de predicción meteorológica o química atmosférica. La gran mayoría de estudios (78%) no

emplearon modelos numéricos, basándose exclusivamente en observaciones directas (estaciones terrestres, sensores satelitales) y datos de reanálisis.

Los seis estudios que emplearon modelos numéricos lo hicieron principalmente en tres modalidades: como fuente de datos de entrada complementaria (dos estudios, 7.4%), como línea base para comparación y validación (cuatro estudios, 15%), y como herramienta para generación de datasets de reanálisis de alta resolución (un estudio, 3.7%).

Entre algunos trabajos notables, K. Zhang et al. (2023) utilizaron indirectamente el modelo WRF-Chem como fuente de datos de entrada para su modelo STA-ResCNN para predicción de PM<sub>2.5</sub> y PM<sub>10</sub> en tres ciudades urbanas de China con horizontes de 1 a 4 horas; mientras que Resifi et al. (2025) representa un caso único donde el modelo fue utilizado para generar un dataset de reanálisis de alta resolución de 42 años (1980-2023) para la Península Arábiga, que posteriormente sirvió como base de entrenamiento para modelos de deep learning. El sistema empleó una configuración asimilativa del modelo WRF con dominios anidados de 15 km y 5 km.

### **3.7 Conclusiones**

Según los estudios encontrados en esta revisión sistemática, se puede concluir que los modelos más utilizados con baja complejidad computacional y que obtienen muy buenos resultados son aquellos que utilizan la arquitectura LSTM gracias a la ventaja que pueden aprender patrones y características temporales a corto y largo plazo (Muthukumar et al., 2022); sin embargo, la principal limitante de utilizar esta arquitectura es la necesidad de data con continuidad temporal para garantizar la obtención de buenos resultados (Miao et al., 2022).

En cuanto a las fuentes de datos empleadas y su relación con los modelos numéricos, se observa que la mayoría de los estudios utilizan variables meteorológicas registradas por sensores y estaciones de calidad del aire, así como imágenes satelitales. Además, en seis investigaciones

se emplearon salidas de modelos numéricos para comparar su desempeño frente a los modelos de aprendizaje profundo, evidenciándose que estos últimos generan predicciones significativamente más rápidas, aunque con menor precisión (Choi et al., 2023; Faraji et al., 2022; Mu et al., 2023; Yu et al., 2023; B. Zhang et al., 2022; K. Zhang et al., 2023). Un caso particular es el estudio de Choi et al. (2023), que utiliza las predicciones del modelo LDAPS como datos adicionales para el entrenamiento, demostrando que el modelo propuesto puede replicar las funciones del modelo numérico con mayor eficiencia y resultados satisfactorios.

## Capítulo 4. Presentación de los resultados esperados

### R1. Implementar el modelo WRF para la predicción de las variables de estudio

El primer resultado esperado del presente trabajo consiste en implementar el modelo WRF en un servidor dedicado provisto por el grupo de Inteligencia Artificial de la Pontificia Universidad Católica del Perú (IA PUCP), que será posteriormente utilizado para la generación del conjunto de datos compuesto de predicciones de T, RH, U y V en la región de estudio, para el entrenamiento del modelo basado en aprendizaje profundo.

Para ello, se procedió a utilizar un archivo bash para la instalación de librerías necesarias y archivos de código fuente del modelo WRF V3. El contenido del archivo se detalla en la Figura 2.

```
source ~/.bashrc
# Actualizar paquetes
sudo apt update
sudo apt upgrade
# Instalar compiladores
sudo apt-get install build-essential csh gfortran m4 perl
# Crear directorios
mkdir $HOME/Build_WRF
mkdir $HOME/Build_WRF/LIBRARIES
# Descargar librerías
```

```

cd $HOME/Build_WRF/LIBRARIES
wget http://www2.mmm.ucar.edu/wrf/OnLineTutorial/compile_tutorial/tar_files/netcdf-
4.1.3.tar.gz
wget http://www2.mmm.ucar.edu/wrf/OnLineTutorial/compile_tutorial/tar_files/mpich-
3.0.4.tar.gz
wget http://www2.mmm.ucar.edu/wrf/OnLineTutorial/compile_tutorial/tar_files/zlib-1.2.7.tar.gz
wget http://www2.mmm.ucar.edu/wrf/OnLineTutorial/compile_tutorial/tar_files/jasper-
1.900.1.tar.gz
wget http://www2.mmm.ucar.edu/wrf/OnLineTutorial/compile_tutorial/tar_files/libpng-
1.2.50.tar.gz
# netcdf
cd $HOME/Build_WRF/LIBRARIES
tar -zxvf netcdf-4.1.3.tar.gz
cd netcdf-4.1.3/
./configure --prefix=$DIR/netcdf --disable-dap --disable-netcdf-4 --disable-shared
make
make install
# mpich
cd $HOME/Build_WRF/LIBRARIES
tar -zxvf mpich-3.0.4.tar.gz
cd mpich-3.0.4/
./configure --prefix=$DIR/mpich
make
make install
# zlib
cd $HOME/Build_WRF/LIBRARIES
tar -zxvf zlib-1.2.7.tar.gz
cd zlib-1.2.7/
./configure --prefix=$DIR/grib2
make
make install
# libpng
cd $HOME/Build_WRF/LIBRARIES
tar -zxvf libpng-1.2.50.tar.gz
cd libpng-1.2.50/
./configure --prefix=$DIR/grib2
make
make install
# Jasper
cd $HOME/Build_WRF/LIBRARIES

```

```

tar -zxvf jasper-1.900.1.tar.gz
cd jasper-1.900.1/
./configure --prefix=$DIR/grib2
make
make install
# GeoData
mkdir $HOME/Build_WRF/GEOG_DATA
cd $HOME/Build_WRF/GEOG_DATA
wget http://www2.mmm.ucar.edu/wrf/src/wps_files/geog_complete.tar.gz
tar -zxvf geog_complete.tar.gz
cp -r $HOME/Build_WRF/GEOG_DATA/geog/modis_landuse_20class_30s_with_lakes
$HOME/Build_WRF/GEOG_DATA
#-----Install WRF 3.9.1.1 -----#
cd $HOME/Build_WRF/
wget http://www2.mmm.ucar.edu/wrf/src/WRFV3.9.1.1.TAR.gz
wget http://www2.mmm.ucar.edu/wrf/src/WRFV3-Chem-3.9.1.TAR.gz
wget http://www2.mmm.ucar.edu/wrf/src/WPSV3.9.1.TAR.gz
wget http://www2.mmm.ucar.edu/wrf/src/WPSV3.8.TAR.gz
wget http://www2.mmm.ucar.edu/wrf/src/WRFV3.8.TAR.gz
wget http://www2.mmm.ucar.edu/wrf/src/WRFV3-Chem-3.8.TAR.gz
tar -zxvf WRFV3.9.1.1.TAR.gz
tar -zxvf WRFV3-Chem-3.9.1.TAR.gz
cp -a $HOME/Build_WRF/chem $HOME/Build_WRF/WRFV3
tar -zxvf WPSV3.9.1.TAR.gz

```

*Figura 2* Código de implementación de la instalación y configuración inicial del modelo numérico WRF en formato bash

Una vez ejecutado el archivo, se procedió a compilar el software WRF y su preprocesador WPS para generar los ejecutables necesarios.

Para el caso del software WRF V3, se utilizó el comando configure y se introdujo la opción 32 seguida de la 1, los cuales corresponden a en serie con compilador gfortran, debido a que WRF solo permite realizar simulaciones con estos parámetros y luego se ejecutó el comando compile con el parámetro em\_real que se utiliza para realizar simulaciones de casos reales que son el objetivo en el presente trabajo. Para el caso del preprocesador WPS, se utilizó el comando configure y se introdujo la primera opción que corresponde a gfortran en serie con archivos grib2 y luego se utilizó el comando compile sin especificar ningún parámetro.

Antes de configurar los parámetros sobre el área del estudio y las condiciones ambientales, es necesario comprender los parámetros más importantes durante la configuración, los cuales son la meteorología, dinámica atmosférica y las grillas (National Center for Atmospheric Research (NCAR), n.d.).

- Meteorología y Dinámica Atmosférica: El sistema meteorológico de WRF divide el dominio espacial en una serie de celdas llamadas "grillas" (grid cells). Para cada celda, se resuelven las ecuaciones de la atmósfera, considerando factores como la temperatura, la presión, la velocidad del viento y la humedad. Estos datos meteorológicos proporcionan la base para comprender cómo se mueven y dispersan los contaminantes en la atmósfera.
- Grillas: Se utilizan diferentes "grids" o grillas para representar escalas espaciales y temporales. Estos son los tres tipos principales de grillas:
  - Domino Principal (Coarse Grid): Es la grilla más grande que abarca toda la región de interés, como un país o una parte de un continente. Proporciona una vista general de las condiciones meteorológicas y químicas a gran escala.
  - Grilla Anidada (Nesting): Dentro del dominio principal, se crean grillas más pequeñas y detalladas que se superponen. Estas grillas anidadas tienen una resolución más alta y se enfocan en áreas específicas para proporcionar detalles más finos.
  - Microescala (Microscale): Estas grillas tienen una resolución aún mayor y se utilizan para simular procesos a nivel de ciudad o localidades específicas. Son útiles para comprender el impacto de los contaminantes a nivel local.

La configuración de parámetros de entrada al modelo para realizar las simulaciones se especifica para una fecha determinada y se encuentran en el archivo namelist.input, cuyo contenido se presenta en la Figura 3.

```

&time_control
2 run_days          = 0,
3 run_hours         = 24,
4 run_minutes       = 0,
5 run_seconds       = 0,
6 start_year        = 2022, 2022,
7 start_month       = 09, 09,
8 start_day         = 14, 14,
9 start_hour        = 00, 00,
10 end_year         = 2022, 2022,
11 end_month        = 09, 09,
12 end_day          = 14, 14,
13 end_hour         = 23, 23,
14 interval_seconds = 3600
15 input_from_file  = .true.,.true.,
16 history_interval = 60, 60,
17 frames_per_outfile = 1, 1,
18 restart          = .false.,
19 restart_interval = 240,
20 io_form_history  = 2
21 io_form_restart  = 2
22 io_form_input    = 2
23 io_form_boundary = 2
24 /
25
26 &domains
27 time_step        = 60,
28 time_step_fract_num = 0,
29 time_step_fract_den = 1,
30 max_dom          = 2,
31 e_we             = 43, 19,
32 e_sn             = 113, 37,
33 e_vert           = 45, 45,
34 dzstretch_s      = 1.1
35 p_top_requested  = 5000,
36 num_metgrid_levels = 34,
37 num_metgrid_soil_levels = 4,
38 dx               = 18000, 6000
39 dy               = 18000, 6000
40 grid_id          = 1, 2,

```

41	parent_id	= 0, 1,
42	i_parent_start	= 1, 19,
43	j_parent_start	= 1, 51,
44	parent_grid_ratio	= 1, 3,
45	parent_time_step_ratio	= 1, 3,
46	feedback	= 1,
47	smooth_option	= 0
48	/	
49		
50	&physics	
51	physics_suite	= 'tropical'
52	mp_physics	= -1, -1,
53	cu_physics	= 0, 0,
54	ra_lw_physics	= -1, -1,
55	ra_sw_physics	= -1, -1,
56	bl_pbl_physics	= -1, -1,
57	sf_sfclay_physics	= -1, -1,
58	sf_surface_physics	= -1, -1,
59	radt	= 15, 15,
60	bldt	= 0, 0,
61	cutd	= 0, 0,
62	icloud	= 1,
63	num_land_cat	= 21,
64	sf_urban_physics	= 0, 0,
65	fractional_seaice	= 1,
66	/	
67		
68	&fdda	
69	/	
70		
	&dynamics	
72	hybrid_opt	= 2,
73	w_damping	= 0,
74	diff_opt	= 2, 2,
75	km_opt	= 4, 4,
76	diff_6th_opt	= 0, 0,
77	diff_6th_factor	= 0.12, 0.12,
78	base_temp	= 290.
79	damp_opt	= 3,
80	zdamp	= 5000., 5000.,

```

81 dampcoef          = 0.2, 0.2,
82 khdif             = 0, 0,
83 kvdif             = 0, 0,
84 non_hydrostatic   = .true., .true.,
85 moist_adv_opt     = 1, 1,
86 scalar_adv_opt    = 1, 1,
87 gwd_opt           = 1, 0,
88 /
89
90 &bdy_control
91 spec_bdy_width    = 5,
92 specified         = .true.
93 /
94
95 &grib2
96 /
97
98 &namelist_quilt
99 nio_tasks_per_group = 0,
100 nio_groups = 1,
101 /

```

*Figura 3* Ejemplo de parámetros de configuración del modelo numérico WRF en el archivo namelist.input para ejecutar 24 horas de simulación para la fecha 14/09/2022

Como se especifica en el archivo de configuración en la Figura 3, se realizó la generación de datos con periodo horario de manera diaria desde la fecha 08/11/2023 hasta 06/08/2024, la cual abarca un periodo de 11 meses.

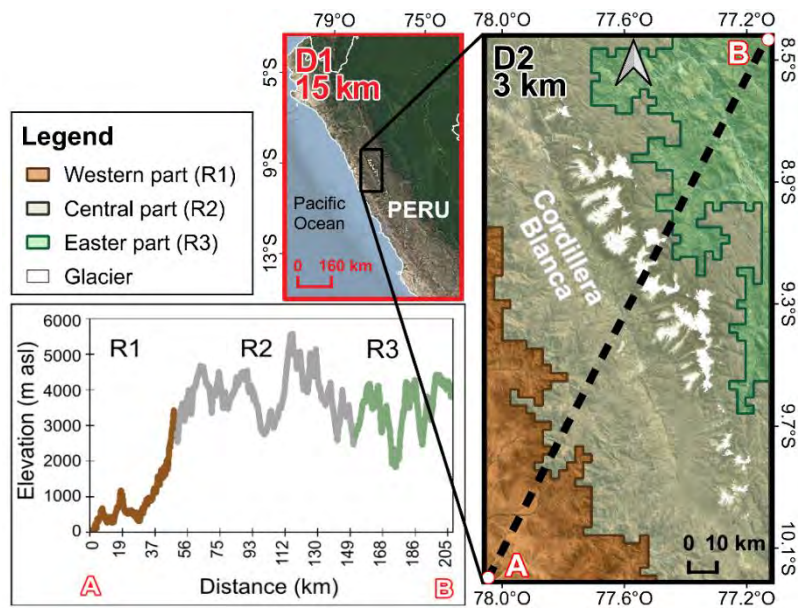


Figura 4 Dominios de modelamiento WRF del área de estudio

En la Figura 4 se ilustra el área de estudio. El primer dominio, D1 (perímetro rojo), representa el área reducida mediante WRF a una resolución espacial de 15 km, mientras que el segundo dominio, D2 (perímetro negro), corresponde a datos climáticos WRF con resolución de 3 km. D2 representa el área de estudio, dividida en tres subregiones: occidental (R1, marrón), central (R2, gris) y oriental (R3, verde).

Además, se mantienen las variables físicas y dinámicas con sus valores por defecto. De esta manera, el modelo numérico WRF ya está listo para ser ejecutado y obtener los resultados de simulación para las variables de interés.

## R2. Conjunto de datos completo y estructurado

El conjunto de datos está conformado por tres fuentes de datos diferentes. Todas las fuentes son utilizadas en diferentes fases de construcción de los modelos basados en aprendizaje profundo propuestos.

La primera fuente de datos corresponde a data meteorológica que sirve de entradas para el modelo WRF y para el entrenamiento del modelo IA. El Global Forecast System (GFS), desarrollado por el National Centers for Environmental Prediction (NCEP), es una fuente

fundamental de datos para modelos de mesoescala como el modelo WRF. El GFS proporciona las condiciones iniciales y de frontera necesarias para el WRF, lo que permite generar pronósticos regionales de alta resolución, esenciales para diversas aplicaciones como pronósticos regionales en operaciones civiles y militares, planificación, investigaciones sobre distribución y transporte de aerosoles, entre otros usos en sectores gubernamentales, agrícolas e industriales. A pesar de su uso extendido, el acceso a datos recientes del GFS puede ser limitado en zonas remotas o durante emergencias, lo que obliga en muchos casos al uso de datos anteriores para mantener la precisión de los pronósticos (Bernard & others, 2016).

Los datos de entrada obtenidos del modelo GFS contienen información a escala global, por lo que es necesario filtrar únicamente aquella correspondiente a la región de estudio. No obstante, con el fin de preservar la continuidad espacial y reducir el riesgo de sobreajuste debido al tamaño limitado del dominio D2, se incluyó un área adicional de aproximadamente 1 a 2 grados alrededor de la región objetivo. En términos espaciales, esto equivale a una extensión aproximada de  $\pm 1.1^\circ$  (alrededor de 125 km) tanto en dirección norte-sur como este-oeste respecto al área de salida. Esta expansión permite incorporar información contextual proveniente de las zonas adyacentes, la cual puede influir significativamente en la evolución de las condiciones atmosféricas locales. Pruebas preliminares realizadas con modelos base mostraron mejoras en el desempeño predictivo al incluir esta zona extendida, lo cual respalda la hipótesis de que las regiones circundantes ejercen una influencia relevante sobre los patrones meteorológicos del área de interés.

Los datos de GFS son originalmente proporcionados en forma de coordenadas geográficas (latitud y longitud). Para preservar la estructura espacial de los registros y facilitar su procesamiento en modelos de aprendizaje profundo, fue necesario transformar esta información a un formato matricial regular. Esta conversión se realizó mediante una reorganización de los registros basada en sus coordenadas geográficas, asegurando así que las

relaciones espaciales entre los puntos se mantuvieran intactas. Los datos de entrada del GFS se representan como una malla de 17 filas (latitud) por 13 columnas (longitud), con una resolución espacial de  $0.25^\circ \times 0.25^\circ$  por celda (aproximadamente  $27 \text{ km} \times 27 \text{ km}$ ), para cada variable meteorológica en una hora determinada.

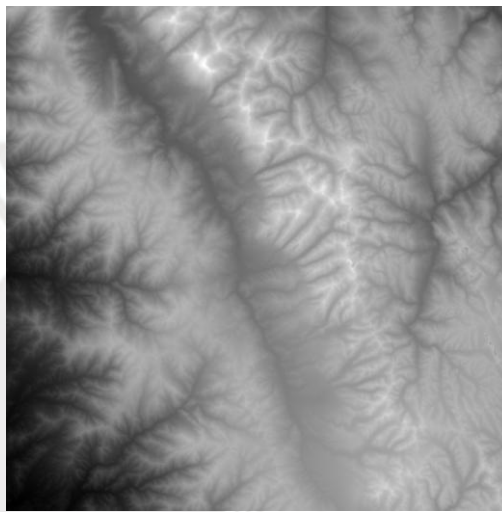
La segunda fuente de datos corresponde a las salidas generadas a partir de la ejecución del modelo numérico WRF durante un periodo de once meses. Estas salidas poseen una resolución espacial de  $3 \text{ km} \times 3 \text{ km}$  por celda y una frecuencia temporal horaria, lo que permite capturar con alta precisión la evolución temporal y espacial de las variables simuladas en la región de estudio. En el presente trabajo, dichas salidas fueron utilizadas para representar la distribución de material particulado, simulada en el dominio de alta resolución (D2).

Cada hora simulada por WRF genera un archivo en formato netCDF, el cual contiene información espacialmente distribuida de las variables de interés en matrices, donde cada celda representa el valor simulado de una variable para una ubicación geográfica específica dentro del dominio. En particular, cada matriz tiene una dimensión de 36 filas por 18 columnas, correspondiendo a una malla espacial que cubre exactamente la región de estudio, a diferencia de GFS que abarca un área mayor.

La tercera fuente de datos corresponde a la información de elevación del terreno, utilizada para incorporar características topográficas en el modelo. En este estudio se emplearon datos de la Misión Radar Topográfica Shuttle (SRTM, por sus siglas en inglés), desarrollada conjuntamente por la NASA y la Agencia Nacional de Inteligencia Geoespacial (NGA) de los Estados Unidos. Esta misión, llevada a cabo en el año 2000 a bordo del transbordador espacial Endeavour, generó un modelo digital de elevación casi global a partir de datos obtenidos mediante interferometría de radar. Los datos de elevación empleados están disponibles de forma pública y presentan una resolución espacial de 30 metros, lo que representa un nivel de detalle considerablemente alto para aplicaciones regionales (NASA Earthdata, 2000). Si bien

el modelo GFS incluye datos de elevación entre sus variables, una evaluación comparativa utilizando un modelo base mostró que la información topográfica proveniente de SRTM ofrece un desempeño significativamente superior, por lo que se optó por emplear esta última fuente en el presente estudio.

La Figura 5 muestra la imagen de elevación utilizada, procesada y adaptada a la región de estudio.



*Figura 5 Gráfico de elevación de terreno del área de estudio*

Como se observa en la Figura 5, el área de elevación considerada abarca la Cordillera Blanca y zonas adyacentes a la región de estudio, lo que permite al modelo basado en aprendizaje profundo incorporar información detallada sobre la forma y altitud del terreno. Esta representación topográfica contribuye a mejorar la calidad de las predicciones al capturar mejor los efectos orográficos que influyen en las variables atmosféricas.

La Figura 6 muestra la estructura y organización de carpetas y archivos del conjunto de datos utilizado para el entrenamiento del modelo.

```
-20220901
-2022090100_GFS.npy
-2022090100_WRF.npy
.
```

```

.
-2022090123_GFS.npy
-2022090123_WRF.npy
.
.
.
-20220914
.
-2022091400_GFS.npy
-2022091400_WRF.npy
.
.
.
-2022091423_GFS.npy
-2022091423_WRF.npy

```

Figura 6 Estructura de archivos del conjunto de datos para el entrenamiento de los modelos basados en aprendizaje profundo

Para cada día de simulación generado por el modelo WRF, se organiza una carpeta con el nombre en formato “YYYYMMDD”. Dentro de cada carpeta se encuentran 24 pares de archivos, uno por cada hora del día. Cada par contiene dos archivos con extensión .npy, que almacenan datos correspondientes a variables atmosféricas y salidas del modelo numérico para esa hora específica. Los archivos siguen el formato de nombre “YYYYMMDDHH\_tipo.npy”, donde “HH” representa la hora del día en el rango [00, 23], y “tipo” indica la fuente de datos utilizada, ya sea WRF o GFS.

La fuente GFS proporciona un total de 15 variables meteorológicas, mientras que el modelo WRF considera únicamente las cuatro variables de interés (T, RH, U, V). La Tabla 6 resume las variables relevantes disponibles en los productos del modelo GFS.

Tabla 6 Variables atmosféricas relevantes presentes en los datos del modelo GFS

Variable	Descripción
T	Temperatura a 2 metros sobre la superficie
RH	Humedad relativa a 2 metros sobre la superficie
U	Componente del viento hacia el este a 10 metros
V	Componente del viento hacia el norte a 10 metros
sp	Presión en superficie

vis	Visibilidad
gust	Ráfagas de viento
fsr	Rugosidad superficial pronosticada
lsm	Máscara tierra-mar
prate	Tasa de precipitación
csnow	Nieve categórica
crain	Lluvia categórica
cape	Energía potencial convectiva disponible
cin	Inhibición convectiva
SUNSD	Duración del brillo solar

Como se muestra en la Tabla 6, las variables proporcionadas por GFS abarcan desde condiciones superficiales (como temperatura, humedad y viento) hasta parámetros relacionados con la precipitación y la convección.

Del total de 154 días de datos disponibles, se utilizaron los últimos 5 días como conjunto de prueba, los 5 días previos como conjunto de validación, y los días restantes conformaron el conjunto de entrenamiento.

### **R3. Revisión sistemática sobre arquitecturas de modelos a implementar**

Como se detalla en las respuestas a las preguntas de revisión sistemática, las arquitecturas más empleadas en estudios recientes orientados a la predicción de variables meteorológicas espaciotemporales con aprendizaje profundo son alguna variación de red recurrente que emplea LSTM o GRU. Este cambio en el enfoque arquitectónico responde a la necesidad de modelar de manera conjunta tanto la evolución temporal como las relaciones espaciales presentes en los campos meteorológicos provenientes de fuentes como el GFS, con el fin de emular las salidas generadas por modelos numéricos como WRF.

La arquitectura LSTM es una variante de las redes neuronales recurrentes diseñada para abordar el problema de las dependencias a largo plazo en secuencias de datos (Hochreiter & Schmidhuber, 1997). Las LSTM contienen unidades de memoria llamadas "celdas", que pueden retener información durante largos periodos de tiempo y decidir cuándo y cómo actualizarla, pues están equipadas con puertas de activación, como la puerta de olvido, la puerta de entrada y la puerta de salida, que controlan el flujo de información en la red (Hochreiter & Schmidhuber, 1997). Las principales ventajas de LSTM son la captura de patrones a largo plazo y el manejo de secuencias de longitud variable, lo que es fundamental en el modelamiento de datos espaciotemporales, donde la cantidad de observaciones en cada serie puede variar (Hochreiter & Schmidhuber, 1997).

La arquitectura GRU es otra variante de las redes neuronales recurrentes que también aborda el problema de las dependencias a largo plazo en secuencias de datos y que, a diferencia de las LSTM, las GRU cuentan con menos compuertas internas y una estructura más simplificada, lo que puede llevar a una implementación más eficiente y un menor costo computacional (Chung et al., 2014). Las principales ventajas de las GRU son la eficiencia temporal y la captura de patrones temporales, lo que las hace adecuadas para problemas de predicción en el tiempo, por ejemplo, la dispersión espaciotemporal de contaminantes (Chung et al., 2014).

Las ventajas ofrecidas por las redes GRU y LSTM, así como la naturaleza de la data disponible, el tipo de modelamiento espaciotemporal y la disponibilidad de implementaciones en código públicas posicionan a ambas arquitecturas como las elegidas para la construcción de los dos modelos propuestos en el presente estudio.

Las arquitecturas seleccionadas corresponden, por un lado, a un modelo tipo encoder-decoder basado en capas LSTM tanto en el codificador como en el decodificador, y por otro, a una red compuesta por bloques ConvLSTM. Ambas arquitecturas están diseñadas para procesar secuencias de entrada formadas por  $n$  pasos temporales (time steps), lo que permite capturar la

evolución temporal de las variables meteorológicas y modelar sus dependencias a lo largo del tiempo.

Estas entradas tienen una estructura de matriz de cuatro dimensiones, donde la primera dimensión corresponde a los  $n$  pasos temporales y las siguientes tres dimensiones representan la componente espacial (filas y columnas) y las features (variables atmosféricas y características relevantes) asociadas a cada ubicación espacial. La salida de la red se genera en  $m$  pasos temporales, pronosticando la evolución futura de las variables durante esos  $m$  pasos. Además, generalmente se reduce el número de features en la salida y se utiliza una mayor resolución espacial, lo que implica un tamaño de grilla mayor (es decir, más filas y columnas en el lado espacial). La Figura 7 ilustra gráficamente la organización de un elemento de entrada para una de estas arquitecturas.

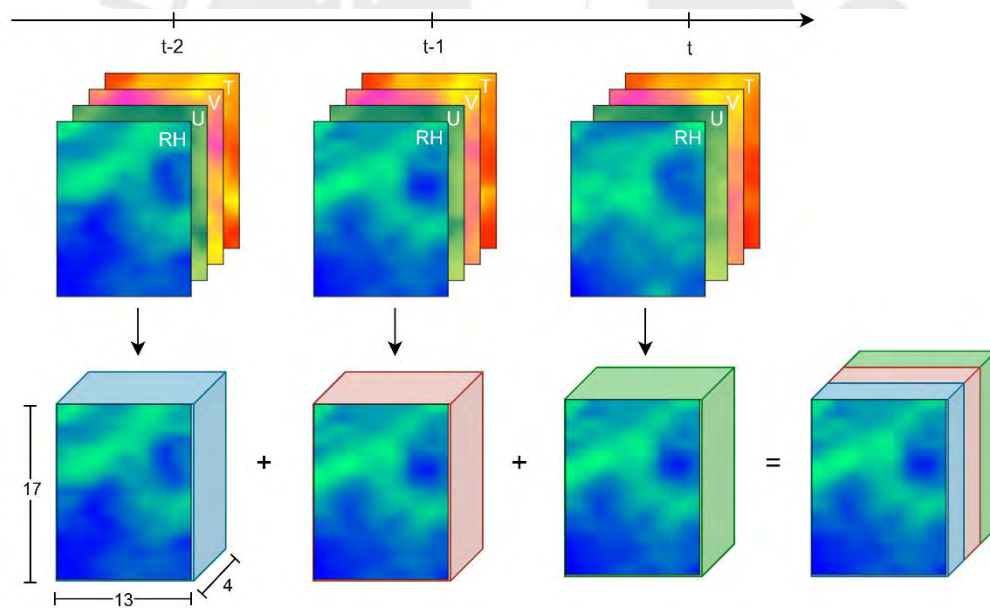
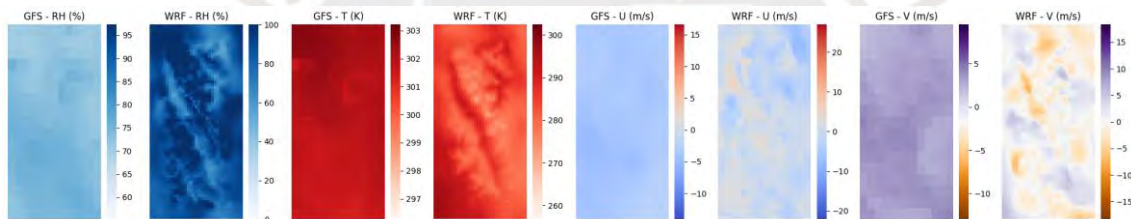


Figura 7 Estructura de un elemento de entrada al modelo IA [elaboración propia]

Como se puede observar en la Figura 7, cada paso temporal se representa mediante la combinación de cuatro matrices bidimensionales, que corresponden a las distintas características espaciales de las variables meteorológicas. Estas matrices bidimensionales se agrupan para formar una matriz tridimensional por cada paso temporal, y a su vez, todas las

matrices tridimensionales se unen para conformar una estructura de cuatro dimensiones que servirá como entrada para los modelos propuestos.

En el caso de las salidas, se genera una matriz de mayor resolución espacial, con dimensiones de 36 filas por 18 columnas, lo que permite capturar con mayor precisión las variaciones locales en comparación con la entrada proveniente del GFS. La Figura 8 muestra, para un paso temporal específico, la comparación entre los campos de las variables meteorológicas simuladas por el modelo GFS filtrado en D2 y las generadas por el modelo WRF. Se aprecia que, debido a la baja resolución de GFS, muchos patrones o gradientes locales tienden a suavizarse o perderse, mientras que en WRF, al operar a una escala más fina, permite evidenciar con mayor claridad las variaciones espaciales que ocurren en regiones específicas.



*Figura 8 Comparación entre salidas del modelo GFS y WRF para el dominio D2 y un mismo paso temporal.*

La estrategia de selección de muestras a partir del dataset emplea una ventana de una unidad temporal, en este caso, de una hora. Esta decisión se tomó debido a que, al aumentar esta ventana, la cantidad de muestras disminuye, lo cual perjudicaría considerablemente al aprendizaje de la red para el caso del presente estudio, en el que se cuenta con menos de un año de datos. La Figura 9 muestra un gráfico que explica la ventana temporal y las entradas y salidas de la red expresadas como muestras.

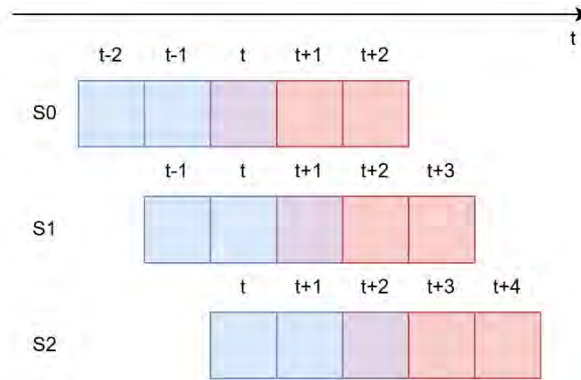


Figura 9 Estrategia de generación de muestras de entrenamiento con ventana deslizante con 3 puntos temporales de entrada (azul) y tres puntos de salida (rojo). [elaboración propia]

Como se observa en la Figura 9, cuyo eje x corresponde al tiempo, se tiene que para cada muestra, por ejemplo la muestra S0, se utilizan 5 puntos temporales para armar las entradas y salidas de una muestra, donde las celdas sombreadas de azul y morado son empleadas como entradas en una muestra, pues se definió que se utilizarán tres pasos temporales como entradas, y las celdas morado y rojo son empleadas como salidas, pues se busca predecir las variables de estudio a una mayor resolución para el tiempo actual y dos pasos en el futuro (t, t+1, t+2). Además, para la siguiente muestra S1, el tiempo actual se desplaza una unidad temporal adelante y se utilizan nuevamente los dos tiempos anteriores al actual y el tiempo actual, donde los elementos de los pasos temporales t-1, t y t+1 son empleados para las entradas al modelo y los pasos temporales t, t+1 y t+2 son empleados como salidas del modelo.

Definida la estructura de cada elemento temporal y seleccionadas las muestras representativas, se proponen dos arquitecturas para la predicción de las variables de interés en el área de estudio. La primera está basada en componentes ConvLSTM, diseñados para capturar patrones espaciotemporales, mientras que la segunda sigue un esquema encoder-decoder con LSTM, enfocado en modelar secuencias temporales de manera más abstracta.

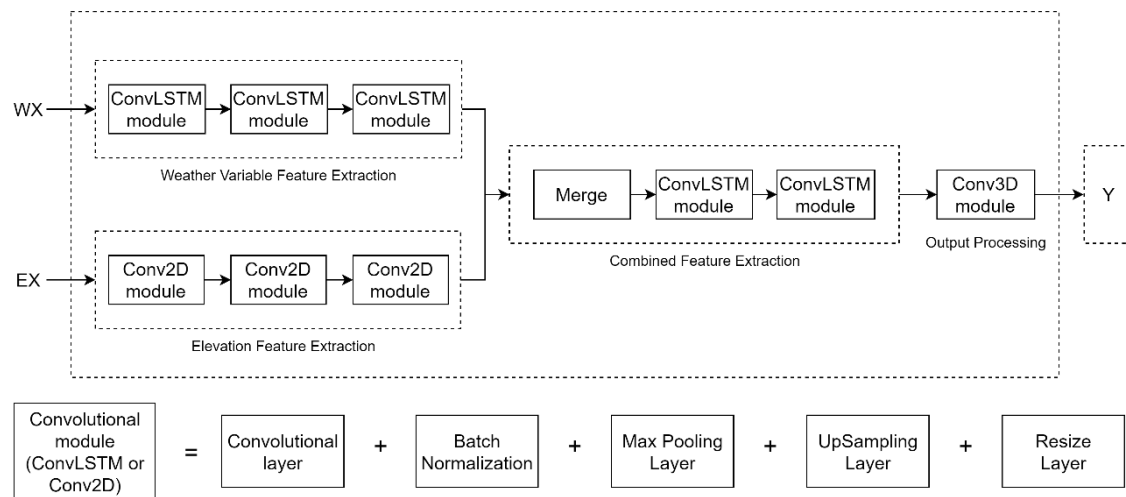


Figura 10 Diagrama de estructura de capas convLSTM [elaboración propia]

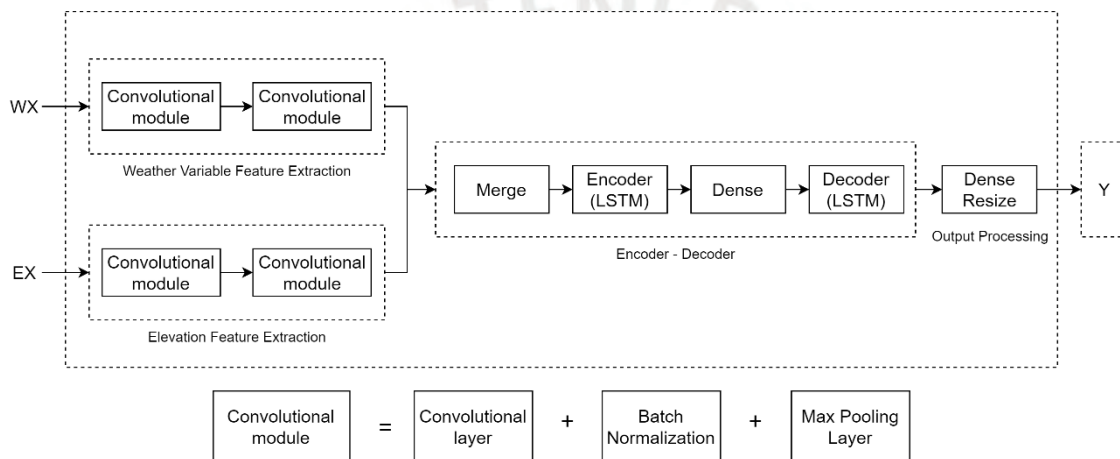


Figura 11 Diagrama de estructura de capas encoder-decoder [elaboración propia]

La Figura 10 presenta la arquitectura propuesta basada en módulos ConvLSTM, mientras que la Figura 11 muestra la arquitectura encoder-decoder diseñada para la misma tarea. Ambas comparten una etapa inicial de extracción de características a partir de entradas climáticas (WX) y de elevación (EX), utilizando módulos convolucionales que combinan capas de convolución, normalización y reducción espacial. Posteriormente, las representaciones generadas son fusionadas para el procesamiento conjunto. La diferencia principal entre ambas arquitecturas se encuentra en el tratamiento de la dimensión temporal y la generación de la salida: la arquitectura ConvLSTM emplea módulos recurrentes convolucionales seguidos de una capa Conv3D para producir la predicción final, mientras que la arquitectura encoder-decoder utiliza una secuencia de codificación y decodificación con LSTM y capas densas

intermedias, finalizando con una proyección mediante una capa densa que ajusta la resolución de salida.

#### **R4. Código fuente de implementación del flujo de entrenamiento de dos modelos algorítmicos.**

En el contexto de la predicción meteorológica, las metodologías se clasifican generalmente en tres categorías: pronósticos basados en juicio, métodos univariantes y métodos multivariantes. Los enfoques basados en juicio se fundamentan en la experiencia y la intuición del pronosticador, a menudo combinados con técnicas estadísticas simples. Los métodos univariantes analizan una sola serie temporal histórica, mientras que los métodos multivariantes consideran múltiples variables interdependientes, proporcionando una representación más realista de los sistemas atmosféricos (Al-Matarneh et al., 2014; Petropoulos et al., 2022).

Los modelos de pronóstico multivariado consideran tanto variables independientes (como condiciones atmosféricas utilizadas para hacer predicciones) como variables dependientes (como temperatura o humedad, que son el resultado esperado). Esta relación permite modelar los mecanismos de causa y efecto presentes en la atmósfera (Sharma et al., 2021; Johnstone y Sulungu, 2021). Al incorporar múltiples variables como temperatura, humedad y dirección del viento, estos enfoques capturan la complejidad inherente de los sistemas meteorológicos, manteniendo las estructuras de dependencia entre variables mediante el uso de matrices multidimensionales.

El éxito del pronóstico multivariado depende del uso efectivo de series temporales, ya sea bajo relaciones lineales o dinámicas no lineales más complejas. En situaciones donde las relaciones funcionales entre variables no son explícitas, se recurre a modelos más avanzados como las

redes neuronales artificiales, capaces de modelar dinámicas complejas y no lineales con alta precisión (Petropoulos et al., 2022).

Para evitar problemas como el sobreajuste y garantizar la capacidad de generalización del modelo, se implementó una estrategia sistemática de selección de características. Esta se basó en principios meteorológicos y prácticas comunes en el pronóstico del tiempo. En primer lugar, se identificaron las 15 variables meteorológicas disponibles (listadas en la Tabla 6). A partir de ellas, se definieron siete grupos de variables (Tabla 7), los cuales fueron evaluados de manera comparativa para determinar cuál ofrecía el mejor rendimiento al generar predicciones del modelo WRF.

Los grupos definidos abarcan distintos niveles de complejidad. El Grupo 1 (G1) incluye todas las variables meteorológicas disponibles, mientras que el Grupo 2 (G2) agrupa aquellas que forman parte de los elementos básicos más comunes en pronósticos tradicionales. Los Grupos 3 a 6 (G3 - G6) evalúan el aporte de subconjuntos más especializados, como estado atmosférico central, interacción viento-superficie, precipitación y convección, y ciclo diurno y radiación. Finalmente, el Grupo 7 (G7) representa un conjunto equilibrado de variables que cubren diversos aspectos de la dinámica atmosférica. Esta metodología permitió una evaluación detallada del impacto de cada conjunto de variables en la precisión del pronóstico, ayudando a identificar configuraciones óptimas para el modelado.

*Tabla 7 Grupos de variables atmosféricas utilizados para evaluar la relevancia en la predicción*

<b>Código</b>	<b>Grupo</b>	<b>VARIABLES</b>
G1	Todos los elementos meteorológicos relevantes	T, RH, U, V, sp, vis, gust, fsr, lsm, prate, csnow, crain, cape, cin, SUNSD
G2	Elementos meteorológicos base del estudio	T, RH, U, V
G3	Estado atmosférico central	T, RH, U, V, sp, vis
G4	Interacción viento-superficie	T, RH, U, V, gust, fsr, lsm

G5	Precipitación y convección	T, RH, U, V, prate, csnow, crain, cape, cin
G6	Radiación y ciclo diurno	T, RH, U, V, SUNSD, sp, lsm
G7	Clima integral	T, RH, U, V, sp, vis, prate, cape, cin, SUNSD, lsm

Para evaluar el impacto de cada grupo de variables en la Tabla 7 sobre el rendimiento de los modelos de inteligencia artificial, se diseñó un proceso sistemático de experimentación utilizando una arquitectura base de tipo LSTM-convolucional. El experimento se llevó a cabo entrenando el modelo durante 100 épocas con un tamaño de batch de 32, utilizando 6 días de datos para entrenamiento y un día para validación. Con el fin de prevenir el sobreajuste, se aplicó una estrategia de early stopping con una paciencia de 20 épocas. Además, para obtener una evaluación robusta y minimizar la influencia de la aleatoriedad en la inicialización de pesos, se realizaron 50 ejecuciones independientes por cada grupo de variables.

El desempeño de cada grupo se evaluó mediante el cálculo del error cuadrático medio (RMSE) sobre el conjunto de validación. La Figura 12 muestra los diagramas de caja con la distribución de los errores obtenidos para cada grupo, permitiendo visualizar la variabilidad y estabilidad de cada conjunto de características. Los resultados indicaron que el grupo G2, compuesto por las variables meteorológicas fundamentales (temperatura, humedad relativa, componente zonal del viento y componente meridional del viento), obtuvo el menor RMSE promedio, así como el menor rango intercuartílico entre las 50 ejecuciones.

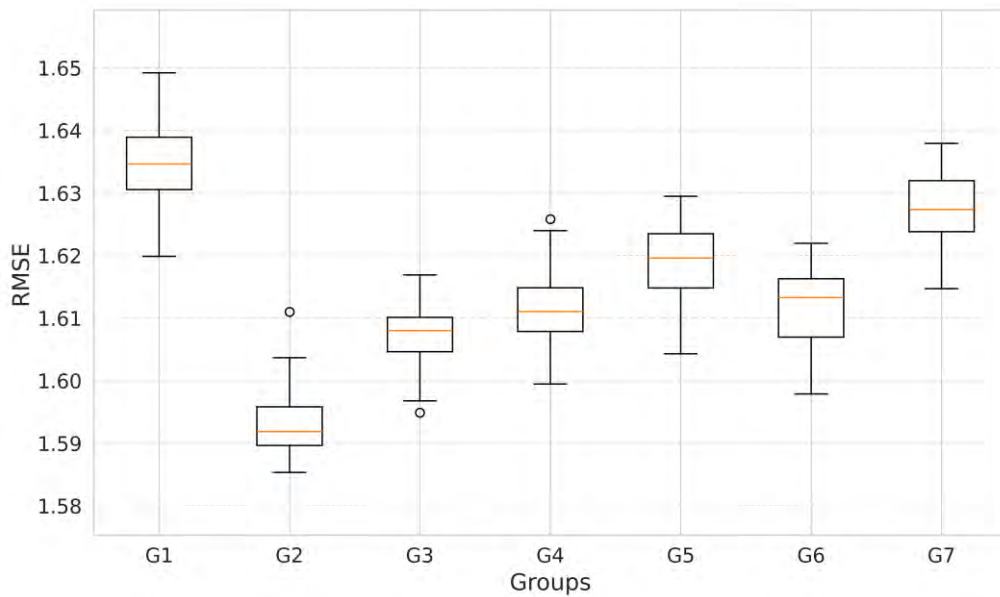


Figura 12 Diagramas de caja del RMSE promedio para siete grupos de variables (G1–G7) evaluados con un modelo base Convolutivo Recurrente.

Este hallazgo resalta la relevancia de un conjunto compacto pero representativo de variables, en línea con el principio de parsimonia en la selección de modelos. A pesar de su simplicidad, G2 superó a grupos más amplios que incluían variables adicionales, lo cual sugiere que estas cuatro variables contienen la mayor parte de la información necesaria para la predicción meteorológica de las variables de interés, evitando posibles redundancias o ruido. Por tanto, se optó por utilizar el conjunto G2 como entrada principal para los modelos IA propuestos, ya que ofrece un equilibrio entre rendimiento predictivo, eficiencia computacional e interpretabilidad. Se define el código base de ambas arquitecturas propuestas, considerando una configuración inicial de capas que no representa su versión final. Estas versiones servirán como punto de partida (baseline) y serán evaluadas bajo las mismas condiciones experimentales. Una vez identificada la arquitectura con mejor desempeño, se procederá con un proceso de ajuste de hiperparámetros para optimizar su rendimiento.

```
def create_convlstm_model():
    # Main Input
    main_input = Input(shape=(3, 17, 13, 4), name='main_input')
```

```

# ConvLSTM2D Layers for Main Input
x = ConvLSTM2D(filters=8, kernel_size=(3, 3), padding='same', return_sequences=True,
activation='relu')(main_input)
x = BatchNormalization()(x)
x = TimeDistributed(UpSampling2D(size=(2, 2)))(x)

x = ConvLSTM2D(filters=8, kernel_size=(3, 3), padding='same', return_sequences=True,
activation='relu')(x)
x = BatchNormalization()(x)
x = TimeDistributed(UpSampling2D(size=(2, 2)))(x)

# Adjust spatial dimensions to (70, 35)
x = TimeDistributed(Conv2D(8, (3, 3), padding='same', activation='relu'))(x)
x = TimeDistributed(UpSampling2D(size=(1, 1)))(x)
x = TimeDistributed(Cropping2D(cropping=((0, 0), (0, 17))))(x)
x = TimeDistributed(Resizing(70, 35))(x)

# Elevation Input
elevation_input = Input(shape=(100, 100, 1), name='elevation_input')

# Processing Elevation Input
elev = Conv2D(8, (3, 3), activation='relu', padding='same')(elevation_input)
elev = MaxPooling2D(pool_size=(2, 2))(elev)
elev = Conv2D(8, (3, 3), activation='relu', padding='same')(elev)
elev = MaxPooling2D(pool_size=(2, 2))(elev)
elev = Conv2D(64, (3, 3), activation='relu', padding='same')(elev)
elev = UpSampling2D(size=(2, 2))(elev)
elev = UpSampling2D(size=(2, 2))(elev)
elev = Conv2D(8, (3, 3), activation='relu', padding='same')(elev)
elev = Cropping2D(cropping=((15, 15), (15, 15)))(elev)
elev = Resizing(70, 35)(elev)
elev = ExpandDimsLayer()(elev)
elev = TileLayer()(elev)

# Combine Main and Elevation Inputs
combined = Concatenate(axis=-1)([x, elev])

# Further ConvLSTM2D Layers

```

```

x = ConvLSTM2D(filters=8, kernel_size=(3, 3), padding='same', return_sequences=True,
activation='relu')(combined)
x = BatchNormalization()(x)
x = ConvLSTM2D(filters=8, kernel_size=(3, 3), padding='same', return_sequences=True,
activation='relu')(x)
x = BatchNormalization()(x)

# Output Layer
outputs = Conv3D(filters=4, kernel_size=(3, 3, 3), activation='linear', padding='same')(x)

model = Model(inputs=[main_input, elevation_input], outputs=outputs)
return model

```

Figura 13 Código fuente de la definición de la estructura del modelo inicial de prueba de la arquitectura ConvLSTM en Python

```

def create_encoder_decoder_model():
    def create_model():
        # Main Input
        input_layer = Input(shape=(3, 17, 13, 4), name='main_input')

        # TimeDistributed Conv2D + BatchNorm blocks
        x = TimeDistributed(Conv2D(16, (2, 2), activation='selu', kernel_regularizer=l2(),
bias_regularizer=l2()))(input_layer)
        x = TimeDistributed(BatchNormalization()(x))

        x = TimeDistributed(Conv2D(16, (2, 2), activation='selu', kernel_regularizer=l2(),
bias_regularizer=l2()))(x)
        x = TimeDistributed(BatchNormalization()(x))

        x = TimeDistributed(Conv2D(16, (2, 2), activation='selu', kernel_regularizer=l2(),
bias_regularizer=l2()))(x)
        x = TimeDistributed(BatchNormalization()(x))

        x = TimeDistributed(Flatten()(x))

        # Elevation Input
        elevation_input = Input(shape=(100, 100, 1), name='elevation_input')

        # Processing Elevation Input
        elev = Conv2D(16, (3, 3), activation='relu', padding='same')(elevation_input)

```

```

elev = MaxPooling2D(pool_size=(2, 2))(elev)
elev = Conv2D(16, (3, 3), activation='relu', padding='same')(elev)
elev = MaxPooling2D(pool_size=(2, 2))(elev)
elev = Conv2D(16, (3, 3), activation='relu', padding='same')(elev)
elev = Flatten()(elev)

# Encoder
x = LSTM(32, activation='tanh', recurrent_activation='tanh',
        kernel_regularizer=l2(), bias_regularizer=l2(),
        dropout=0.02, recurrent_dropout=0.3)(x)

x = Dense(units=256, activation="tanh")(x)

# Combine with elevation features
combined = Concatenate()([x, elev])

# Repeat for decoder
x = RepeatVector(3)(combined)

# Decoder
x = LSTM(32, activation='tanh', recurrent_activation='tanh',
        kernel_regularizer=l2(), bias_regularizer=l2(),
        dropout=0.02, recurrent_dropout=0.3, return_sequences=True)(x)

# Output
x = TimeDistributed(Dense(units=9800, activation="linear"))(x)
output_layer = TimeDistributed(Reshape((70, 35, 4)))(x)

return Model(inputs=[input_layer, elevation_input], outputs=output_layer)

```

*Figura 14* Código fuente de la definición de la estructura del modelo inicial de prueba de la arquitectura encoder-decoder en Python

La arquitectura ConvLSTM realiza una extracción de características de manera espaciotemporal, preservando la estructura espacial en cada paso temporal mediante capas convolucionales y de memoria. Esto permite capturar dinámicamente las relaciones espaciotemporales desde la entrada hasta la reconstrucción de la salida.

Por otro lado, la arquitectura encoder-decoder con TimeDistributed realiza primero una extracción espacial por cada time step de forma independiente, utilizando capas convolucionales aplicadas secuencialmente en el tiempo. Posteriormente, las representaciones extraídas se aplanan y se ingresan al encoder (LSTM), donde se modela la dependencia temporal. Finalmente, el decoder genera la salida a través de una capa densa y se le da forma mediante una capa de reshape para recuperar la estructura espacial original.

## **R5. Dos modelos algorítmicos entrenados**

Para analizar la variabilidad espacial en el desempeño de los modelos de IA, el área de estudio se dividió en tres regiones: occidental, central y oriental. La región occidental (R1) abarca altitudes desde 48 hasta 3798 m, incluyendo zonas costeras y de sierra baja. La región central (R2) comprende altitudes entre 891 y 6688 m, e incluye los picos más altos de la cordillera. Finalmente, la región oriental (R3) se extiende hasta el límite con la Amazonía, con altitudes entre 1644 y 4652 m. Esta división permite evaluar el rendimiento del modelo en distintos rangos altitudinales y condiciones climáticas.

La diversidad climática y altitudinal del área de estudio proporciona un entorno ideal para poner a prueba la capacidad de los modelos climáticos en contextos de alta complejidad atmosférica. Esta característica fue clave para seleccionar la región, ya que permite desarrollar modelos de IA adaptados a escenarios exigentes, fortaleciendo su capacidad predictiva y su aplicabilidad en zonas montañosas a nivel global.

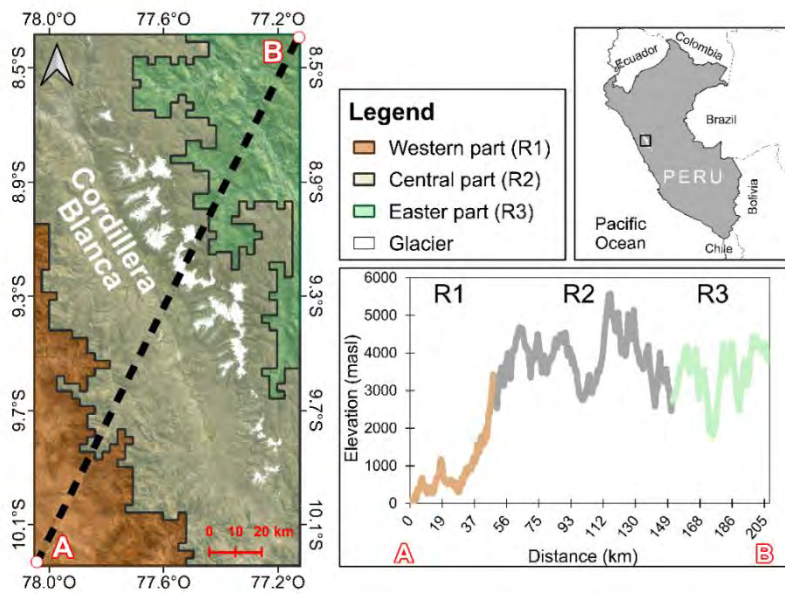


Figura 15 División regional del área de estudio en la Cordillera Blanca (R1–R3) según gradiente altitudinal y ubicación geográfica [elaboración propia]

La métrica empleada para la evaluación de los modelos es el RMSE o error cuadrático medio y se obtiene para cada time step de salida ( $t$ ,  $t+1$ ,  $t+2$ ), región y conjunto de datos (entrenamiento, validación, prueba).

Ambos modelos se entrenaron bajo los mismos parámetros para garantizar condiciones equivalentes de evaluación, tal como se resume en la Tabla 8. Se implementaron estrategias de control como early stopping y guardado automático del mejor modelo para evitar el sobreajuste y asegurar un entrenamiento robusto.

Tabla 8 Parámetros de entrenamiento de arquitectura de modelos propuestos

Parámetro	Valor	Descripción
Función de pérdida	Mean Square Error	La función de error cuadrático medio será empleada para el entrenamiento
Épocas máximas	2000	Número máximo de iteraciones del entrenamiento
Tamaño de batch	64	Cantidad de muestras procesadas por lote
Embarajamiento	False	Desactiva el reordenamiento aleatorio de los datos debido a que se trabaja con data temporal
Early Stopping	patience=200	Detiene el entrenamiento si no hay mejora en validación en 200 épocas

Restauración de pesos	<code>restore_best_weights=True</code>	Restaura los pesos del modelo con mejor desempeño en validación
Guardado del mejor modelo	<code>ModelCheckpoint</code>	Almacena el modelo con menor pérdida en validación

Los resultados del entrenamiento de ambos modelos base, Encoder-Decoder y ConvLSTM, se presentan en la Tabla 9 y Tabla 10. En estas se muestran los valores de error cuadrático medio (RMSE) para cada una de las variables meteorológicas (RH, T, U, V) en los pasos temporales  $t$ ,  $t+1$  y  $t+2$ , tanto en los conjuntos de entrenamiento como de validación, desagregados por región (R1, R2 y R3). En general, el modelo Encoder-Decoder muestra un mejor desempeño numérico en el conjunto de entrenamiento, con errores más bajos en casi todas las variables y regiones. Sin embargo, en la validación, el modelo ConvLSTM presenta una mayor capacidad de generalización, especialmente para variables como temperatura (T) y viento (U y V), donde los errores son notablemente menores en comparación al Encoder-Decoder.

Además de estos resultados cuantitativos, el análisis cualitativo mostrado en la Figura 16, un conjunto de mapas de calor que compara las predicciones de cada variable meteorológica para ambos modelos refuerza las diferencias observadas entre las arquitecturas. En dicho gráfico se aprecia que el modelo ConvLSTM logra capturar con mayor fidelidad la forma del terreno en las predicciones, especialmente en variables como la temperatura (T) y la humedad relativa (RH). En contraste, las predicciones generadas por el modelo Encoder-Decoder presentan una representación más difusa del relieve, con menor detalle espacial, lo que indica que este modelo no logra incorporar de manera efectiva las características de elevación presentes en los datos. Esta capacidad del modelo ConvLSTM para reflejar mejor la topografía lo hace visualmente más coherente con la salida del modelo WRF original, lo que sugiere que, pese a errores ligeramente mayores en entrenamiento, su estructura de memoria espacial y temporal contribuye a una generalización más fiel a la realidad física del entorno simulado.

Tabla 9 Resultados de entrenamiento del modelo con arquitectura Encoder-Decoder para los conjuntos de entrenamiento y validación

RMSE		Variable atmosférica											
		RH			T			U			V		
Set	Región	t	t+1	t+2	t	t+1	t+2	t	t+1	t+2	t	t+1	t+2
Train	R1	29.79	28.06	26.89	0.25	0.66	0.44	0.54	1.15	1.06	1.36	1.08	1.43
	R2	23.70	22.02	21.20	0.54	1.30	0.29	0.37	0.87	0.70	1.92	2.15	1.64
	R3	28.83	27.12	26.00	0.32	0.81	0.42	0.52	1.11	1.00	1.47	1.32	1.46
Val	R1	34.76	32.51	30.84	2.92	2.28	1.26	2.90	2.55	2.03	3.08	3.59	3.08
	R2	32.46	25.41	20.58	3.91	3.07	1.63	2.96	3.38	3.49	3.16	4.68	4.55
	R3	34.37	31.40	29.33	3.11	2.43	1.33	2.91	2.71	2.35	3.10	3.80	3.38

Tabla 10 Resultados de entrenamiento del modelo con arquitectura ConvLSTM para los conjuntos de entrenamiento y validación

RMSE		Variable atmosférica											
		RH			T			U			V		
Set	Región	t	t+1	t+2	t	t+1	t+2	t	t+1	t+2	t	t+1	t+2
Train	R1	35.48	38.25	31.85	0.30	1.19	0.58	0.89	2.77	2.37	2.02	1.22	1.10
	R2	34.14	34.54	29.14	0.65	2.20	1.00	0.58	1.84	1.58	2.27	2.59	2.26
	R3	35.25	37.64	31.40	0.39	1.42	0.67	0.84	2.63	2.26	2.07	1.55	1.37
Val	R1	26.78	21.94	25.30	2.76	1.84	1.53	3.91	3.41	2.59	3.41	3.69	3.56
	R2	21.26	14.14	17.21	3.92	3.27	2.23	3.03	2.85	2.98	3.28	4.34	4.64
	R3	25.91	20.81	24.10	2.99	2.15	1.67	3.77	3.32	2.66	3.39	3.81	3.77

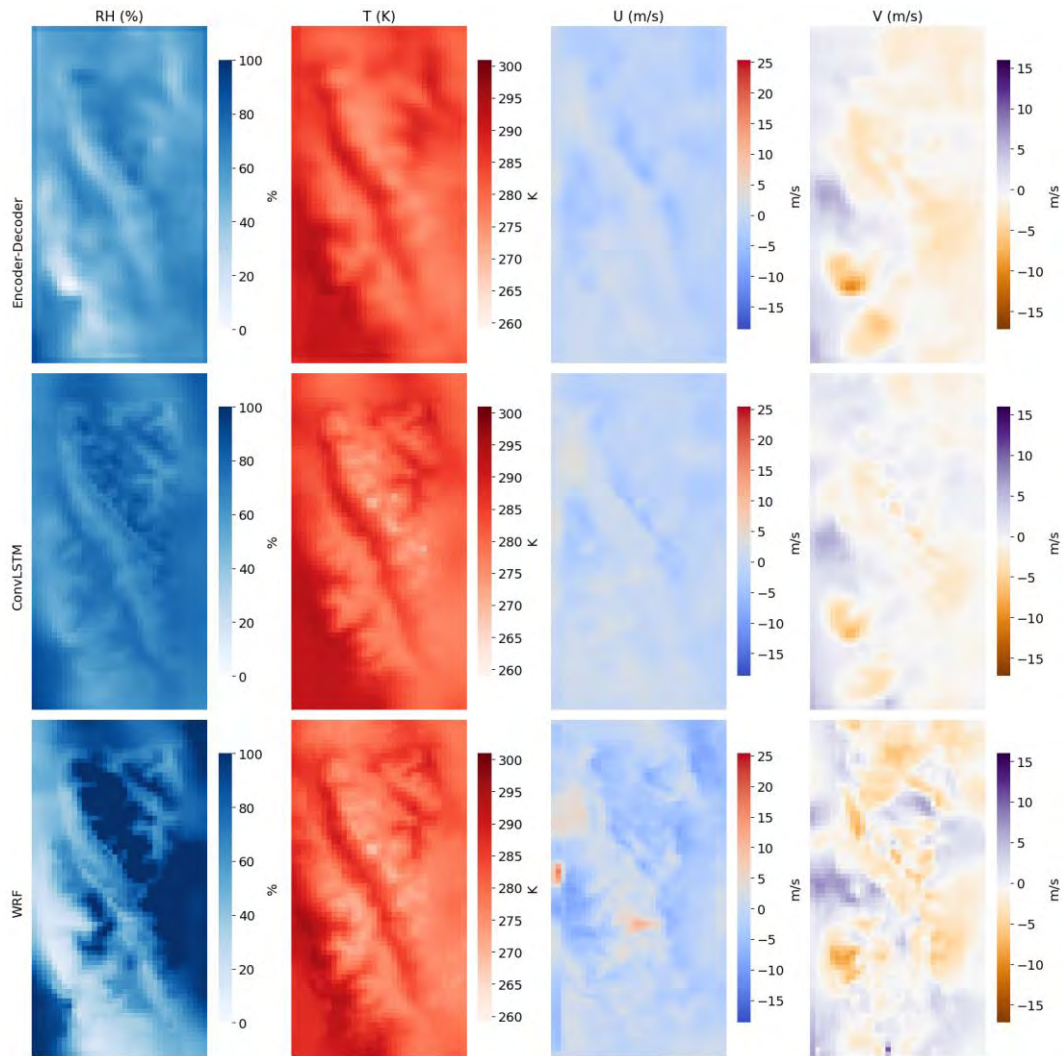


Figura 16 Mapas de calor para un tiempo determinado para las predicciones de las arquitecturas Encoder-Decoder, ConvLSTM comparadas con WRF

## R6. Estudio comparativo de evaluación de desempeño entre modelos algorítmicos

Para comparar las dos arquitecturas de modelos de aprendizaje profundo, se emplea un diseño de experimentos basados en diferentes entrenamientos y utilizando pruebas estadísticas para agregar rigor científico, controlar variables, evaluar la estabilidad de resultados y reducir sesgos en la comparación. Esto permite identificar diferencias significativas y proporcionar información cuantitativa confiable, lo que asegura una elección de arquitectura basada en resultados sólidos y reproducibles.

Se plantea un estudio comparativo para seleccionar el mejor modelo a partir de un conjunto de veinte entrenamientos por cada arquitectura elegida, bajo las mismas condiciones de

entrenamiento, de los cuales se obtiene el mejor modelo con la menor pérdida RMSE total de las 100 épocas de entrenamiento. Además, se registra la métrica RMSE total y por time step de salida por cada entrenamiento, los cuales son empleados posteriormente para las pruebas de hipótesis con el objetivo de encontrar aquella arquitectura cuyos resultados generalmente son mejores a la otra.

Luego de ejecutar el código de entrenamiento y obtención de resultados, se obtuvieron los valores de RMSE que se muestran en la Tabla 11, así como los valores RMSE para la salida total en la Figura 17.

*Tabla 11 Valores de RMSE del mejor modelo por corrida, arquitectura y salida*

Modelo/ Corrida	ConvLSTM	Encoder- Decoder
1	0.98077051	0.97068326
2	0.97941326	0.96687339
3	0.95315349	0.97158911
4	0.94436646	0.97684116
5	0.94714696	0.96629806
6	0.94435685	0.96629815
7	0.96644271	0.97717825
8	0.94965073	0.97230758
9	0.94025281	0.96488613
10	0.95717511	0.97095834
11	0.93878461	0.96492247
12	0.95165198	0.96490861
13	0.93213518	0.96915475
14	0.93781853	0.95622338
15	0.95154396	0.95735347
16	0.95641841	0.96432925

17	0.95131452	0.96162599
18	0.94873137	0.96958846
19	0.94706227	0.96225483
20	0.93646551	0.95922915
21	0.94329361	0.97649687
22	0.94562646	0.96634832
23	0.95928631	0.96810815
24	0.95286477	0.95915449
25	0.93390485	0.96443668

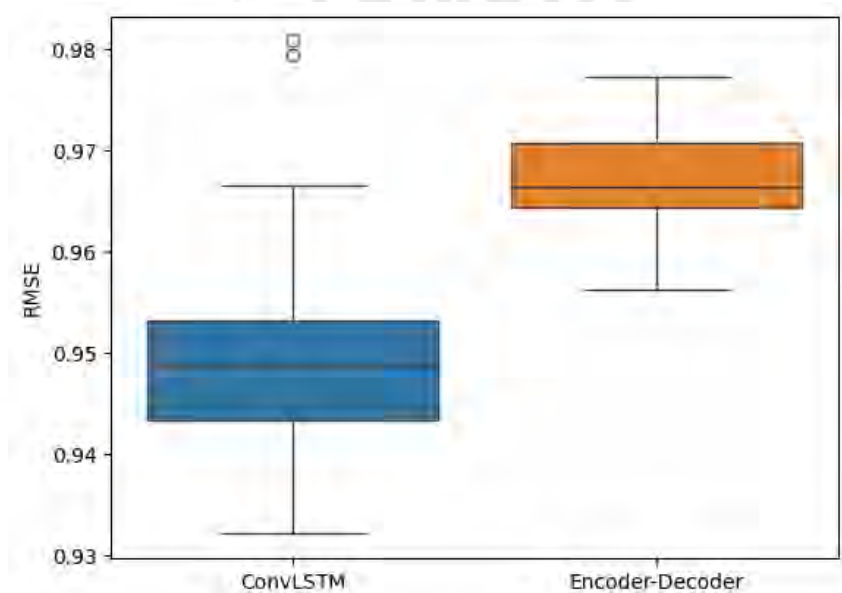


Figura 17 Frecuencias de RMSE del mejor modelo por corrida, arquitectura y salida total

Dado que no se conocen las varianzas poblacionales de los errores RMSE, y únicamente se dispone de las varianzas muestrales obtenidas a partir de 25 ejecuciones independientes por cada arquitectura (ConvLSTM y Encoder-Decoder), es necesario verificar si existe igualdad de varianzas entre ambos conjuntos de resultados. Para ello, se plantea la realización de una prueba de hipótesis que permita contrastar estadísticamente dicha igualdad. Las variables muestrales a considerar se definen de la siguiente manera:

$$X = \text{RMSE de arquitectura ConvLSTM}$$

$Y = RMSE \text{ de arquitectura Encoder - Decoder}$

Se plantean las hipótesis empleando las variables muestrales:

$$H_0: \frac{\sigma^2 x}{\sigma^2 y} = 1$$

$$H_1: \frac{\sigma^2 x}{\sigma^2 y} \neq 1$$

Luego, se calcula el estadístico de prueba:

$$F_0 = \frac{S^2 x}{S^2 y} = \frac{14.84576 - 07}{5.99735e - 07} = 4.50694$$

Se define un valor de significancia de 0.05 y se hallan las dos colas en la distribución de Fisher.

$$F(0.025, 24, 24) = 0.44066$$

$$F(0.975, 24, 24) = 2.26927$$

Dado que el valor del estadístico de prueba no se encuentra dentro del intervalo de aceptación, se rechaza la hipótesis nula, concluyéndose que las varianzas entre ambos conjuntos de resultados no son iguales.

Para evaluar si los errores RMSE generados por las arquitecturas ConvLSTM y Encoder-Decoder siguen una distribución normal, se aplicó la prueba de Shapiro-Wilk, la cual es apropiada para tamaños de muestra pequeños ( $n = 25$  en ambos casos). Se formularon las siguientes hipótesis para cada arquitectura:

$H_{0ConvLSTM}$ : Los resultados siguen una distribución normal

$H_{1ConvLSTM}$ : Los resultados no siguen una distribución normal

$H_{0Encoder-Decoder}$ : Los resultados siguen una distribución normal

$H_{1Encoder-Decoder}$ : Los resultados no siguen una distribución normal

Se utilizó un nivel de significancia de  $\alpha = 0.05$ . Los valores p obtenidos fueron los siguientes de 0.0378 para ConvLSTM y 0.6353 para Encoder-Decoder.

Dado que el valor  $p$  para ConvLSTM es menor al nivel de significancia, se rechaza la hipótesis nula, concluyéndose que los errores de esta arquitectura no siguen una distribución normal. Por otro lado, para Encoder-Decoder, el valor  $p$  es considerablemente mayor que 0.05, por lo que no se rechaza la hipótesis nula, y se asume que los errores sí siguen una distribución normal.

Ambos modelos fueron entrenados de forma separada utilizando los mismos datos de entrenamiento y validación, así como los mismos hiperparámetros, pero en ejecuciones diferentes, por lo que se concluye que ambos son independientes.

Se cumplen los requisitos necesarios para aplicar la prueba no paramétrica de Mann–Whitney U, dado que las muestras a comparar son independientes, no presentan igualdad de varianzas y al menos una de ellas no sigue una distribución normal. Bajo este contexto, se plantea la siguiente hipótesis nula: ambas arquitecturas presentan errores RMSE provenientes de la misma distribución. Como hipótesis alternativa, se considera que la arquitectura Encoder-Decoder presenta errores significativamente mayores que la arquitectura ConvLSTM.

Con base en los resultados obtenidos de la prueba de Mann–Whitney U, con un estadístico U de 70.0 y un valor  $p$  de  $1.33 \times 10^{-6}$ , se rechaza la hipótesis nula  $H_0$ , que establecía que no existía diferencia significativa en los errores RMSE entre ConvLSTM y Encoder-Decoder. Dado que el valor  $p$  es considerablemente menor que el umbral de significancia de 0.05, se concluye que ConvLSTM presenta errores RMSE significativamente menores que Encoder-Decoder. Este resultado sugiere que la arquitectura ConvLSTM tiene un mejor desempeño en términos de precisión en la predicción, lo que la convierte en una opción más adecuada para la tarea de predicción en este contexto.

## **R7. Estrategia de evaluación numérica para comparar el modelo seleccionado en O2 y WRF**

Para realizar una evaluación del modelo de aprendizaje profundo seleccionado (ConvLSTM) en comparación con WRF, se establece una metodología de evaluación que abarca tanto la eficiencia computacional como la calidad predictiva de ambos enfoques. Esta estrategia de evaluación se estructura en tres componentes principales que permiten una comparación objetiva entre el modelo de inteligencia artificial y el modelo numérico tradicional.

En primer lugar, se procederá con la optimización de hiperparámetros del modelo ConvLSTM seleccionado. Para este propósito, se empleó una estrategia de búsqueda aleatoria utilizando Keras Tuner para optimizar los hiperparámetros clave del modelo. Este enfoque ha demostrado ser más eficiente que la búsqueda en grilla para espacios de alta dimensionalidad (Bergstra & Bengio, 2012) y ha sido aplicado exitosamente en diversos estudios de modelado meteorológico y climático. Rasp et al. (2018) utilizaron búsqueda aleatoria para ajustar una red neuronal profunda destinada a la parametrización de física subgrilla en modelos climáticos, mientras que Kratzert et al. (2019) la aplicaron para optimizar arquitecturas LSTM en el modelado de escorrentía pluvial. En ambos casos, la búsqueda aleatoria permitió identificar configuraciones óptimas del modelo que mejoraron significativamente el rendimiento predictivo en comparación con métodos tradicionales.

El espacio de búsqueda de hiperparámetros se muestra en la Tabla 12.

*Tabla 12 Espacio de búsqueda de hiperparámetros del modelo ConvLSTM*

Hiperparámetro	Rango
Número de filtros convolucionales para entrada principal	[8, 16, 32, 64]
Número de filtros convolucionales para entrada de elevación	[8, 16, 32, 64]
Número de filtros convolucionales para procesamiento combinado	[8, 16, 32, 64]
Tasa de dropout para capas LSTM	[0, 0.5]

Tasa de dropout recurrente para capas LSTM	[0, 0.5]
Tasa de aprendizaje para optimizador Adam	[1e-4, 1e-2]

La búsqueda se realizó considerando seis parámetros clave de la arquitectura del modelo y del proceso de entrenamiento. El número de filtros convolucionales para la rama de entrada principal, la rama de entrada de elevación y las capas de procesamiento combinado se exploró dentro del conjunto discreto [8, 16, 32, 64]. Para la regularización, tanto las tasas de dropout estándar como recurrente en las capas LSTM se variaron entre 0 y 0.5. La tasa de aprendizaje para el optimizador Adam también fue objeto de optimización sistemática.

Posteriormente, se llevará a cabo una evaluación de la eficiencia computacional, comparando los tiempos de entrenamiento y predicción requeridos por el modelo ConvLSTM optimizado con respecto a WRF. Esta evaluación considerará los recursos computacionales demandados por cada modelo, así como su escalabilidad temporal, proporcionando una perspectiva sobre las ventajas operacionales de cada enfoque.

Finalmente, se realizará una evaluación detallada de la calidad predictiva mediante la aplicación de métricas de error. Esta evaluación analizará la capacidad del modelo para reproducir con fidelidad los patrones espaciotemporales de las variables meteorológicas de interés obtenidas de WRF, considerando los diferentes horizontes de predicción establecidos ( $t$ ,  $t+1$ ,  $t+2$ ).

### **R8. Estudio comparativo del desempeño predictivo y tiempo de procesamiento entre los modelos algorítmicos y WRF**

Los resultados de la optimización de hiperparámetros aplicada a la arquitectura ConvLSTM seleccionada revelan patrones sobre la configuración óptima del modelo. La Tabla 13 presenta las cinco mejores configuraciones de hiperparámetros identificadas durante el proceso de búsqueda aleatoria, junto con sus correspondientes valores de pérdida de validación, mientras

que la Figura 18 proporciona una representación gráfica de estos resultados, facilitando la visualización de las tendencias y patrones identificados en el espacio de hiperparámetros explorado.

Tabla 13 Espacio de búsqueda de hiperparámetros del modelo ConvLSTM

Hiperparámetro	Modelo 1	Modelo 2	Modelo 3	Modelo 4	Modelo 5
num_conv_filters_main	8	16	32	16	8
num_conv_filters_elev	8	32	32	16	64
num_conv_filters_combined	16	8	8	8	16
lstm_dropout_rate	0.0	0.0	0.0	0.0	0.0
lstm_recurrent_dropout_rate	0.0	0.0	0.0	0.0	0.0
learning_rate	0.010154	0.009086	0.008078	0.036760	0.031226
batch_size	64	96	96	64	64
val_loss	1.613361	1.888414	2.767747	3.612598	7.916195

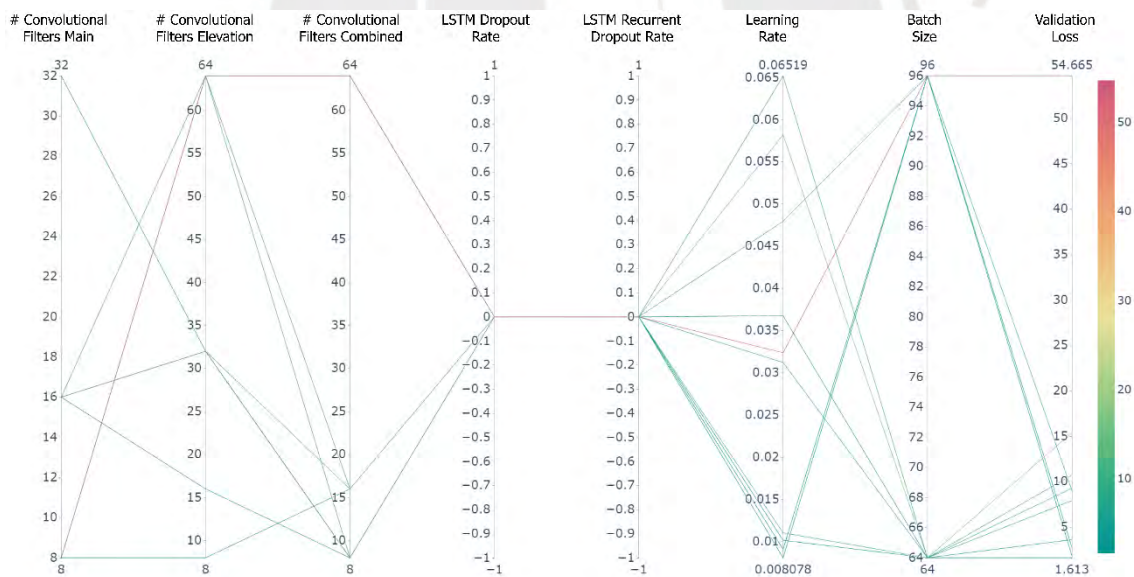


Figura 18 Mejores configuraciones de hiperparámetros del modelo CNN-LSTM obtenidas mediante la búsqueda aleatoria

El experimento reveló que el número de filtros convolucionales impacta significativamente el rendimiento del modelo, obteniéndose la mejor pérdida de validación (1.613) mediante una arquitectura relativamente simple de 8 filtros tanto en las ramas principal como de elevación, y 16 filtros en las capas combinadas. Esto sugiere que una arquitectura más ligera supera a

configuraciones más complejas, ya que los modelos con mayor cantidad de filtros (32-64) mostraron un rendimiento considerablemente inferior con pérdidas superiores a 2.7.

Los parámetros de entrenamiento también demostraron ser cruciales, con la configuración óptima utilizando una tasa de aprendizaje de 0.010154 y un tamaño de batch de 64. Todas las configuraciones exitosas mantuvieron tasas de dropout nulas, indicando que la regularización adicional fue innecesaria para esta arquitectura. Estos resultados demuestran que el modelo se beneficia más de la simplicidad arquitectural y parámetros de aprendizaje apropiados que de configuraciones complejas o técnicas de regularización.

El modelo ConvLSTM optimizado fue entrenado bajo las mismas condiciones experimentales especificadas en la Tabla 8, empleando la configuración de hiperparámetros óptima identificada en la fase de optimización. Para evaluar el desempeño predictivo del modelo propuesto, se realizó una comparación con las predicciones generadas por WRF sobre el conjunto de datos de prueba, analizando tanto la precisión puntual como los patrones espaciotemporales de las variables meteorológicas de interés.

La evaluación cuantitativa del desempeño se presenta en la Figura 19, la cual muestra gráficos de dispersión para cada horizonte temporal de predicción ( $t$ ,  $t+1$ ,  $t+2$ ) y variable de estudio.

Complementariamente, la Figura 20 presenta una evaluación espacial detallada mediante mapas de calor (heatmaps) correspondientes a una fecha y hora específica del conjunto de prueba. Esta visualización ilustra el desempeño predictivo espacial del modelo entrenado en comparación con WRF, así como la distribución espacial del error absoluto entre ambas aproximaciones para cada horizonte temporal y variable meteorológica analizada.

Finalmente, la Tabla 14 muestra los valores de RMSE resultantes del proceso de entrenamiento, desagregados por conjunto de datos (entrenamiento, validación y prueba), variable de estudio (RH, T, U, V), región (R1, R2, R3) y paso temporal ( $t$ ,  $t+1$ ,  $t+2$ ).

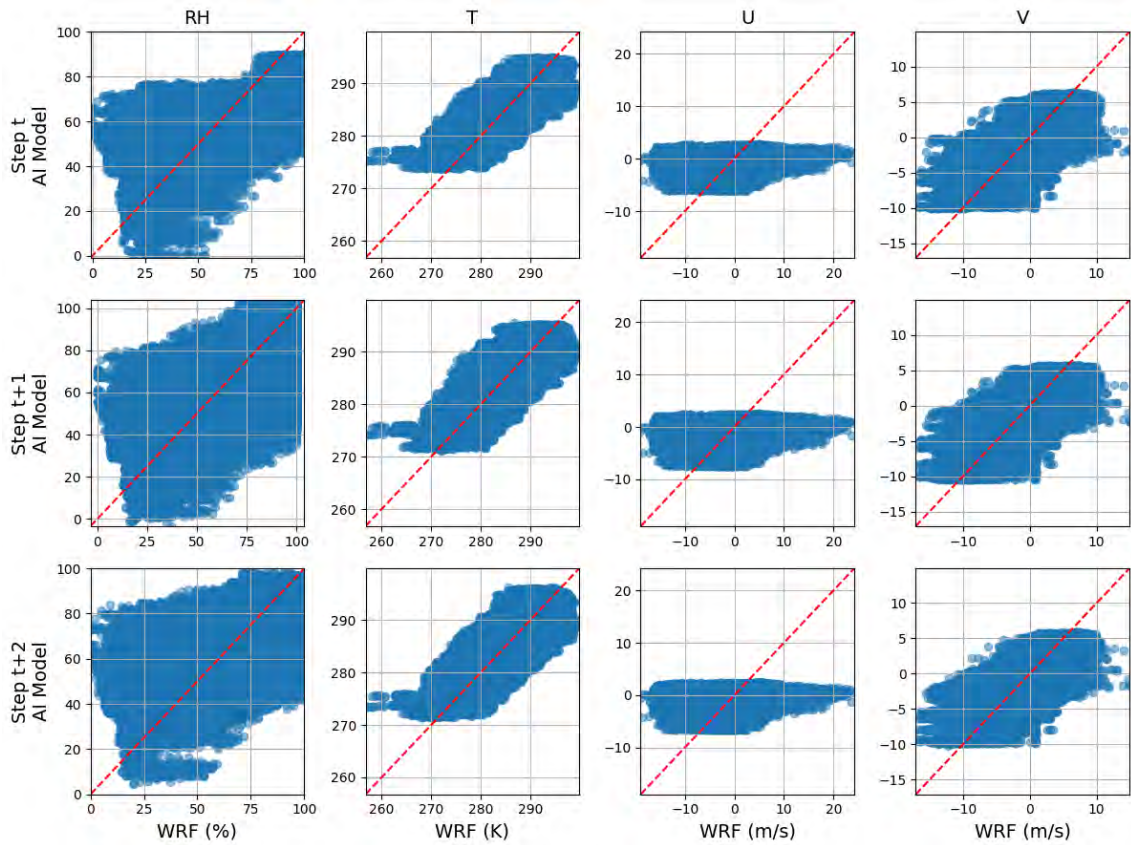


Figura 19 Gráficos de dispersión del modelo ConvLSTM para el conjunto de prueba por variable y paso temporal.

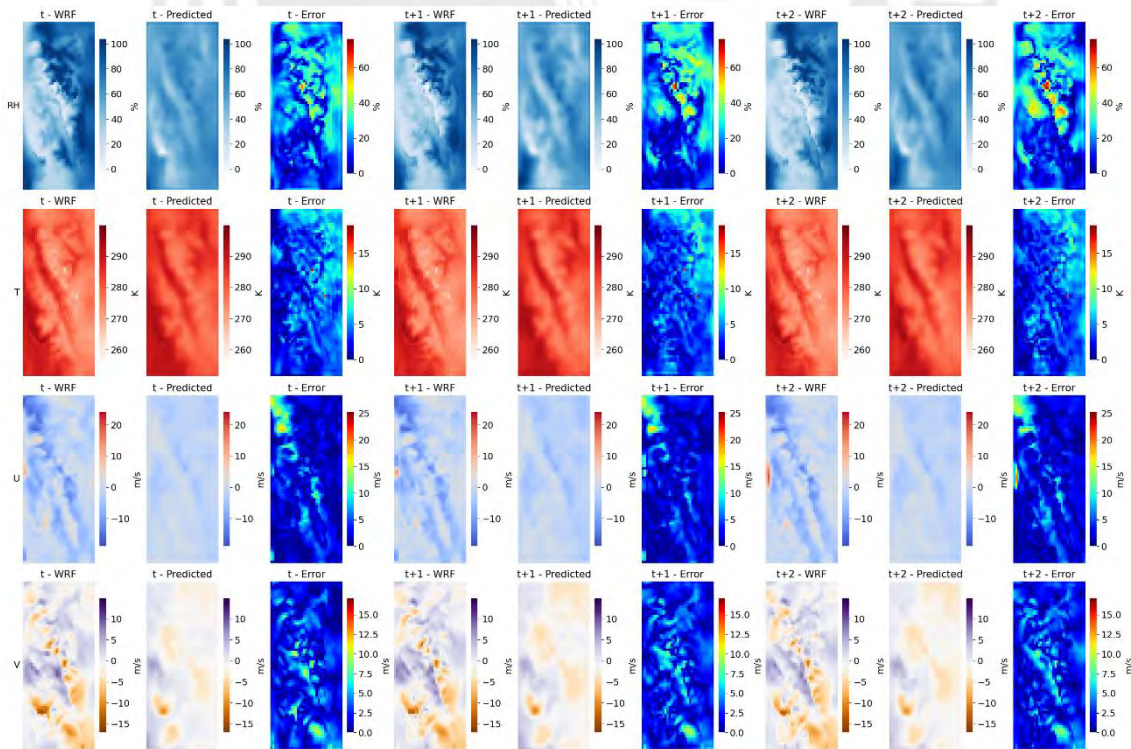


Figura 20 Mapas de calor de salidas y error absoluto del modelo ConvLSTM y WRF por variable de estudio y paso temporal.

Tabla 14 Resultados de entrenamiento del modelo con arquitectura ConvLSTM optimizado para los conjuntos de entrenamiento, validación y prueba

RMSE		Variable atmosférica											
		RH			T			U			V		
Set	Región	t	t+1	t+2	t	t+1	t+2	t	t+1	t+2	t	t+1	t+2
Train	R1	13.28	12.25	11.13	0.29	0.52	0.52	1.51	0.27	0.18	0.72	0.94	1.90
	R2	9.18	14.52	14.24	0.86	1.19	0.51	1.27	0.33	0.18	1.03	0.59	2.26
	R3	12.56	12.74	11.82	0.47	0.71	0.52	1.46	0.28	0.18	0.80	0.88	1.98
Val	R1	4.49	16.02	17.88	1.47	0.22	0.50	0.04	1.15	0.79	0.27	1.77	0.21
	R2	7.11	17.08	12.41	1.71	0.58	1.51	0.10	1.22	1.75	0.20	1.00	0.12
	R3	5.13	16.24	16.92	1.52	0.32	0.81	0.06	1.17	1.06	0.25	1.64	0.19
Test	R1	18.09	16.71	16.42	0.35	0.46	0.85	1.40	1.37	1.36	0.52	3.31	0.54
	R2	15.78	19.97	13.35	0.68	0.41	0.89	2.61	2.86	1.41	1.58	1.97	0.30
	R3	17.65	17.42	15.85	0.44	0.45	0.85	1.72	1.77	1.37	0.85	3.09	0.50

El análisis comparativo regional, basado en la Tabla 14 y la Figura 20, revela una notable variabilidad espacial en el desempeño del modelo. Las predicciones en la región R1 (regiones costeras y de baja altitud) fueron las más precisas, con valores bajos de RMSE incluso en horizontes de pronóstico más largos. Esta precisión se atribuye a dinámicas atmosféricas relativamente estables en un ambiente costero semiárido a árido, con variabilidad topográfica mínima y la fuerte influencia de las corrientes oceánicas frías causadas por el afloramiento a lo largo de la costa del Pacífico. Este afloramiento es impulsado por vientos alisios del sureste asociados con el anticiclón del Pacífico sureste (SEPA, por sus siglas en inglés), y mantiene bajas temperaturas superficiales del mar (SST), las cuales, combinadas con una inversión térmica persistente de bajo nivel, estabilizan la atmósfera inferior y reducen la variabilidad en variables meteorológicas clave como temperatura y velocidad del viento, resultando en mayor predictibilidad (Aliaga-Nestares et al., 2023).

La región R2 (región montañosa central), caracterizada por gradientes pronunciados y cambios de elevación, exhibe valores más altos de RMSE, particularmente para componentes del viento, reflejando las dificultades en el modelamiento en áreas con microclimas pronunciados y condiciones meteorológicas diversas. Los vientos de niveles medios y superiores del este que se originan del Alto de Bolivia interactúan con los Andes, creando microclimas y mejorando la variabilidad espacial. La precipitación, por ejemplo, puede variar dramáticamente en distancias cortas, lo que complica el desarrollo de modelos precisos (Aliaga-Nestares et al., 2023; Martínez et al., 2024). Los Andes orientales, particularmente en la transición hacia la cuenca del Amazonas, presentan dinámicas hidro climáticas únicas influenciadas por la interacción entre la circulación atmosférica regional, la topografía y el intercambio de humedad con el Amazonas (Martínez et al., 2024).

La región R3 (región oriental), donde los datos de elevación parecen mejorar la sensibilidad del modelo a las variaciones del terreno, presenta un RMSE bajo. Este menor RMSE puede atribuirse a la alta coherencia espacial del transporte de humedad a gran escala y la fuerte influencia de la cuenca del Amazonas, que proporciona una fuente relativamente consistente de vapor de agua. Este vapor de agua, transportado hacia el oeste por los vientos alisios, es elevado por las pronunciadas pendientes andinas, generando precipitación orográfica que domina los patrones hidro climáticos de la región (Aliaga-Nestares et al., 2023; Martínez et al., 2024). Al incorporar datos detallados de cobertura del suelo y observaciones de alta resolución, el desempeño del modelo puede mejorarse aún más, capturando las dinámicas de meso escala de la convección y la distribución espacial de la precipitación.

*Tabla 15 Entorno computacional empleado para la corrida de experimentos*

Detalle de recurso	Especificación
GPU	NVIDIA RTX A5500

CPU	AMD Ryzen Threadripper PRO 5955WX 16-Core
BackEnd	Tensorflow 2.13.1
API de Deep Learning	Keras 2.3.1
Lenguaje de programación	Python 3.10.12
IDE	Jupyter Lab 4.2.1

Tabla 16 Comparación de desempeño computacional entre el modelo IA optimizado y WRF

Métrica	Modelo IA optimizado	Modelo WRF
<b>Tamaño del programa</b>		
Almacenamiento	13 MB	2.306 GB
Requerido	-	29 GB
<b>Entrenamiento</b>		
Tiempo	179.15 s	N/A
FLOPs	125 400 576	N/A
Uso de memoria pico	434.45 MB	N/A
Parámetros	1 121 752	N/A
Velocidad	31.86 batches/segundo	N/A
<b>Métricas de predicción (1 día de simulación)</b>		
Tiempo de inferencia	0.08 s	386 s
Uso de memoria pico (RAM)	0.25 MB	3 MB
Velocidad de inferencia	300 predicciones/segundo	0.06 predicciones/segundo

La Tabla 15 muestra los recursos computacionales en los que se corren las simulaciones de WRF y el modelo IA propuesto, mientras que la Tabla 16 proporciona un resumen de la comparación de desempeño entre el modelo ConvLSTM y la configuración del modelo WRF. Las métricas mostradas en esta tabla se derivan de 100 ejecuciones de entrenamiento, con los valores medios registrados para cada métrica. Las métricas generales destacan los requisitos de almacenamiento, mientras que las métricas de entrenamiento se enfocan en el consumo de recursos y la eficiencia del modelo de IA durante el entrenamiento.

La Tabla 8 demuestra la eficiencia del modelo de IA en comparación con el modelo WRF. El modelo de IA requiere solo 13 MB de almacenamiento, significativamente menos que los 31.306 GB del modelo WRF, y alcanza velocidades de predicción de 300 predicciones por segundo versus 0.06 para WRF. Con un tiempo de predicción diaria de 0.08 s y un uso mínimo de memoria pico (0.25 MB), el modelo de IA es altamente eficiente para aplicaciones en tiempo real.

## **Capítulo 5. Conclusiones y trabajos futuros**

### **5.1 Conclusiones**

El presente trabajo desarrolló y evaluó dos modelos de aprendizaje profundo para la generación de datos meteorológicos de alta resolución: un modelo basado en arquitectura Encoder–Decoder y un modelo ConvLSTM optimizado con doble entrada. Ambos modelos fueron entrenados utilizando como entrada los campos atmosféricos de baja resolución de GFS y como salida las simulaciones de alta resolución del WRF. El objetivo fue evaluar la capacidad de ambas arquitecturas para aprender las relaciones espaciotemporales entre las variables atmosféricas de entrada y las simuladas por WRF, y comparar su rendimiento y eficiencia computacional respecto a dicho modelo numérico.

Los resultados demostraron que ambos enfoques de inteligencia artificial pueden reproducir con buena fidelidad los patrones espaciotemporales de las variables temperatura (T), humedad relativa (RH) y componentes del viento (U y V). Sin embargo, el modelo ConvLSTM optimizado superó consistentemente al modelo Encoder–Decoder en precisión, estabilidad del entrenamiento y capacidad de generalización. Su arquitectura, que integra capas ConvLSTM2D para la extracción de dependencias temporales y convoluciones espaciales, demostró ser más adecuada para capturar las variaciones dinámicas de las variables meteorológicas en el dominio andino.

El uso de datos de elevación como entrada estática secundaria en el modelo permitió mejorar la representación de la influencia topográfica en los patrones atmosféricos, especialmente en las regiones costera (R1) y amazónica (R3), donde el modelo mostró los menores errores medios (RMSE). No obstante, en la región montañosa (R2), caracterizada por fuertes gradientes altitudinales y microclimas heterogéneos, la precisión disminuyó, evidenciando la necesidad de técnicas que mejoren la modelación de interacciones complejas en terrenos abruptos.

El análisis comparativo de desempeño entre ambos modelos indicó que el ConvLSTM alcanzó menores errores en los tres horizontes de predicción ( $t$ ,  $t+1$  y  $t+2$ ), manteniendo una coherencia espacial más alta en las variables estimadas. Además, la evolución del RMSE mostró una tendencia creciente con el horizonte temporal, comportamiento esperado en modelos recurrentes que dependen de la memoria temporal.

Desde el punto de vista computacional, el modelo ConvLSTM también demostró ventajas significativas en eficiencia. Requirió solo 13 MB de almacenamiento, con un tiempo de inferencia de 0.08 segundos por día simulado, frente a los 2.3 GB de almacenamiento y 386 segundos del modelo WRF para un dominio equivalente. Esto representa un incremento aproximado de 5000 veces en velocidad de predicción, junto con una drástica reducción en uso de memoria y recursos de cómputo. Por su parte, el modelo Encoder–Decoder, aunque más liviano, no alcanzó la misma precisión ni estabilidad que la arquitectura ConvLSTM, mostrando una mayor sensibilidad a la configuración de hiperparámetros y a la variabilidad regional.

En conjunto, los resultados confirman que el modelo ConvLSTM de doble entrada constituye la mejor alternativa entre las arquitecturas evaluadas, al ofrecer un equilibrio óptimo entre precisión, coherencia espacial y eficiencia computacional. Su desempeño respalda su potencial

aplicación como un modelo liviano de downscaling climático, capaz de reproducir las simulaciones de WRF con un costo computacional varias órdenes de magnitud menor.

## **5.2 Trabajos futuros**

El presente estudio sienta las bases para futuras investigaciones orientadas a mejorar los modelos de predicción atmosférica mediante técnicas de aprendizaje profundo. La combinación de la eficiencia computacional de los modelos basados en IA con la precisión física de los modelos numéricos tradicionales, como el WRF, representa una línea de investigación prometedora para el desarrollo de sistemas híbridos de pronóstico climático regional. Dichos sistemas podrían aprovechar las ventajas de ambos enfoques, manteniendo la coherencia física de las simulaciones numéricas y la rapidez de los modelos neuronales en la generación de resultados.

La investigación se centró en la predicción de cuatro variables principales (T, RH, U y V) a partir de campos atmosféricos de baja resolución. Sin embargo, el marco metodológico desarrollado puede extenderse a un conjunto más amplio de variables meteorológicas, tal como lo hace el modelo WRF, que simula decenas de campos físicos simultáneamente. Para alcanzar este objetivo, se podría incrementar el número de filtros de salida y diseñar una arquitectura multivariable capaz de representar interdependencias entre variables, dando lugar a un modelo IA regional integral.

Respecto al tratamiento espacial, los resultados mostraron que el modelo ConvLSTM optimizado logra una buena representación de los patrones regionales, aunque presenta limitaciones en zonas de topografía compleja. En este sentido, futuras investigaciones podrían incorporar funciones de pérdida ponderadas o enfoques multi-escala que otorguen mayor importancia a regiones montañosas (como la zona R2), mejorando la sensibilidad del modelo ante gradientes altitudinales y microclimas locales. Paralelamente, la integración de

observaciones reales de estaciones meteorológicas o sensores satelitales permitiría aplicar corrección de sesgo (bias correction) y fortalecer la validez del modelo en condiciones reales de campo.

Otra línea de investigación consiste en ampliar el periodo de entrenamiento y validación para incluir datos multianuales o de diferentes estaciones climáticas. Esto permitiría evaluar la capacidad del modelo para generalizar en distintos regímenes meteorológicos y detectar su desempeño ante eventos extremos, como precipitaciones intensas o anomalías asociadas a El Niño y La Niña. Además, el entrenamiento con resoluciones temporales menores (sub-horarias) podría mejorar la capacidad del modelo para representar procesos atmosféricos rápidos, relevantes para sistemas de alerta temprana.

Finalmente, considerando los bajos requerimientos de almacenamiento e inferencia del modelo ConvLSTM, se abre la posibilidad de escalar las predicciones a nivel nacional o implementar el modelo en dispositivos de bajo consumo, tales como microservidores o estaciones locales de monitoreo. Esta ampliación permitiría el desarrollo de un sistema de pronóstico regional operativo, capaz de generar proyecciones meteorológicas de corto plazo con alta resolución espacial y temporal. La consolidación de este tipo de sistemas híbridos de IA y modelado físico podría transformar la forma en que se producen y distribuyen los pronósticos atmosféricos en regiones de alta complejidad topográfica como los Andes peruanos.

## Referencias

- Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., & Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11), e00938. <https://doi.org/10.1016/j.heliyon.2018.e00938>
- Aliaga-Nestares, V., De La Cruz, G., & Takahashi, K. (2023). Comparison between the Operational and Statistical Daily Maximum and Minimum Temperature Forecasts on the Central Coast of Peru. *Weather and Forecasting*, 38(4), 555–570. <https://doi.org/10.1175/WAF-D-21-0094.1>
- Alizadeh, N., Mahdianpari, M., Hemmati, E., & Marjani, M. (2025). FusionFireNet: A CNN-LSTM model for short-term wildfire hotspot prediction utilizing spatio-temporal datasets. *Remote Sensing Applications: Society and Environment*, 37, 101436. <https://doi.org/10.1016/j.rsase.2024.101436>
- An, N., Xie, E., Yu, Y., Yang, Y., Lv, Q., Zhang, S., Zhan, W., & Wu, Y. (2025). Spatio-temporal prediction of soil hydro-thermal response in embankment to the varying climatic conditions using a Two-Step LSTM-ML approach. *Transportation Geotechnics*, 55, 101648. <https://doi.org/10.1016/j.trgeo.2025.101648>
- Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13, 281–305.
- Bernard, D. E., & others. (2016). *Operational Use of NWP Models at AFWA* (Issue AD1016607). <https://apps.dtic.mil/sti/pdfs/AD1016607.pdf>
- Chen, Z., Zhang, B., Du, C., Meng, W., & Meng, A. (2024). A novel dynamic spatio-temporal graph convolutional network for wind speed interval prediction. *Energy*, 294, 130930. <https://doi.org/10.1016/j.energy.2024.130930>
- Chimborazo, O., Minder, J. R., & Vuille, M. (2022). Observations and Simulated Mechanisms of Elevation-Dependent Warming over the Tropical Andes. *Journal of Climate*, 35(3), 1021–1044. <https://doi.org/10.1175/JCLI-D-21-0379.1>
- Chimborazo, O., & Vuille, M. (2021). Present-day climate and projected future temperature and precipitation changes in Ecuador. *Theoretical and Applied Climatology*, 143(3–4), 1581–1597. <https://doi.org/10.1007/s00704-020-03483-y>

- Choi, H., Park, S., Kang, Y., Im, J., & Song, S. (2023). Retrieval of hourly PM<sub>2.5</sub> using top-of-atmosphere reflectance from geostationary ocean color imagers I and II. *Environmental Pollution*, 323, 121169. <https://doi.org/10.1016/j.envpol.2023.121169>
- Chollet, F. (2017). *Deep Learning with Python*.
- Chu, X., Bai, W., Sun, Y., Li, W., Liu, C., & Song, H. (2022). A Machine Learning-Based Method for Wind Fields Forecasting Utilizing GNSS Radio Occultation Data. *IEEE Access*, 10, 30258–30273. <https://doi.org/10.1109/ACCESS.2022.3159231>
- Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*.
- Cifuentes, F., González, C. M., & Aristizábal, B. H. (2021). Insights to WRF-Chem sensitivity in a zone of complex terrain in the tropical Andes: Effect of boundary conditions, chemical mechanisms, nesting, and domain configuration. *Atmospheric Pollution Research*, 12(6), 101093. <https://doi.org/10.1016/j.apr.2021.101093>
- Doblas-Reyes, F., Sörensson, A., Almazroui, M., Dosio, A., Gutowski, W., Haarsma, R., Hamdi, R., Hewitson, B., Kwon, W.-T., Lamptey, B., Maraun, D., Stephenson, T., Takayabu, I., Terray, L., Turner, A., & Zuo, Z. (2023). Linking Global to Regional Climate Change. In *Climate Change 2021 – The Physical Science Basis* (pp. 1363–1512). Cambridge University Press. <https://doi.org/10.1017/9781009157896.012>
- Fang, G., Zhu, Y., & Zhang, J. (2024). Spatiotemporal Evolution Analysis of PM<sub>2.5</sub> Concentrations in Central China Using the Random Forest Algorithm. *Sustainability*, 16(19), 8613. <https://doi.org/10.3390/su16198613>
- Faraji, M., Nadi, S., Ghaffarpasand, O., Homayoni, S., & Downey, K. (2022). An integrated 3D CNN-GRU deep learning method for short-term prediction of PM<sub>2.5</sub> concentration in urban environment. *Science of The Total Environment*, 834, 155324. <https://doi.org/10.1016/j.scitotenv.2022.155324>
- Farhangmehr, V., Imanian, H., Mohammadian, A., Cobo, J. H., Shirkhani, H., & Payeur, P. (2025). A spatiotemporal CNN-LSTM deep learning model for predicting soil temperature in diverse large-scale regional climates. *Science of The Total Environment*, 968, 178901. <https://doi.org/10.1016/j.scitotenv.2025.178901>
- Feurer, M., & Hutter, F. (2019). Hyperparameter Optimization. In *Automated Machine Learning* (pp. 3–33). Springer, Cham.

- Gallo, C., Eden, J. M., Dieppois, B., Drobyshev, I., Fulé, P. Z., San-Miguel-Ayanz, J., & Blackett, M. (2023). Evaluation of CMIP6 model performances in simulating fire weather spatiotemporal variability on global and regional scales. *Geoscientific Model Development*, *16*(10), 3103–3122. <https://doi.org/10.5194/gmd-16-3103-2023>
- Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media, Inc.
- GitHub Inc. (n.d.). *GitHub*. Retrieved June 2, 2023, from <https://github.com/about>
- Gokul, P. R., Mathew, A., Bhosale, A., & Nair, A. T. (2023). Spatio-temporal air quality analysis and PM2.5 prediction over Hyderabad City, India using artificial intelligence techniques. *Ecological Informatics*, *76*, 102067. <https://doi.org/10.1016/j.ecoinf.2023.102067>
- Gong, B., Langguth, M., Ji, Y., Mozaffari, A., Stadtler, S., Mache, K., & Schultz, M. G. (2022). Temperature forecasting by deep learning methods. *Geoscientific Model Development*, *15*(23), 8931–8956. <https://doi.org/10.5194/gmd-15-8931-2022>
- Gupta, P., Verma, S., Mukhopadhyay, P., Bhatla, R., & Payra, S. (2024). Fidelity of WRF model in simulating heat wave events over India. *Scientific Reports*, *14*(1), 2693. <https://doi.org/10.1038/s41598-024-52541-2>
- Gutierrez, R. A., Junquas, C., Armijos, E., Sörensson, A. A., & Espinoza, J. (2024). Performance of Regional Climate Model Precipitation Simulations Over the Terrain-Complex Andes-Amazon Transition Region. *Journal of Geophysical Research: Atmospheres*, *129*(1). <https://doi.org/10.1029/2023JD038618>
- He, H., Shi, B., Hao, Y., Feng, L., Lyu, X., & Ling, Z. (2024). Forecasting sea surface temperature during typhoon events in the Bohai Sea using spatiotemporal neural networks. *Atmospheric Research*, *309*, 107578. <https://doi.org/10.1016/j.atmosres.2024.107578>
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, *9*(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>

- Iskandaryan, D., Ramos, F., & Trilles, S. (2023). Graph Neural Network for Air Quality Prediction: A Case Study in Madrid. *IEEE Access*, *11*, 2729–2742.  
<https://doi.org/10.1109/ACCESS.2023.3234214>
- Johnstone, C., & Sulungu, E. D. (2021). Application of neural network in prediction of temperature: a review. *Neural Computing and Applications*, *33*(18), 11487–11498.  
<https://doi.org/10.1007/s00521-020-05582-3>
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., & Herrnegger, M. (2019). Rainfall–runoff modelling using long short-term memory (LSTM) networks. *Hydrology and Earth System Sciences*, *23*, 1–23.
- Li, S., Chen, M., Yi, L., Lu, Q., & Yang, H. (2025). MIESTC: A Multivariable Spatio-Temporal Model for Accurate Short-Term Wind Speed Forecasting. *Atmosphere*, *16*(1), 67. <https://doi.org/10.3390/atmos16010067>
- López-Noreña, A. I., Berná, L., Tames, M. F., Millán, E. N., Puliafito, S. E., & Fernandez, R. P. (2022). Influence of emission inventory resolution on the modeled spatio-temporal distribution of air pollutants in Buenos Aires, Argentina, using WRF-Chem. *Atmospheric Environment*, *269*, 118839.  
<https://doi.org/10.1016/j.atmosenv.2021.118839>
- Lovell-Smith, J. W., Feistel, R., Harvey, A. H., Hellmuth, O., Bell, S. A., Heinonen, M., & Cooper, J. R. (2016). Metrological challenges for measurements of key climatological observables. Part 4: atmospheric relative humidity. *Metrologia*, *53*(1), R40–R59.  
<https://doi.org/10.1088/0026-1394/53/1/R40>
- Mamani, R., & Hendrick, P. (2021). WRF Model Parameterization Around the Highland Titicaca Lake. *Earth and Space Science*, *8*(12). <https://doi.org/10.1029/2021EA001649>
- Martinez, J. A., Arias, P. A., Junquas, C., Espinoza, J. C., Condom, T., Dominguez, F., & Morales, J. S. (2022). The Orinoco Low-Level Jet and the Cross-Equatorial Moisture Transport Over Tropical South America: Lessons From Seasonal WRF Simulations. *Journal of Geophysical Research: Atmospheres*, *127*(3).  
<https://doi.org/10.1029/2021JD035603>
- Martinez, J. A., Junquas, C., Bozkurt, D., Viale, M., Fita, L., Trachte, K., Campozano, L., Arias, P. A., Boisier, J. P., Condom, T., Goubanova, K., Pabón-Caicedo, J. D., Poveda, G., Solman, S. A., Sörensson, A. A., & Espinoza, J. C. (2024). Recent progress in

- atmospheric modeling over the Andes – part I: review of atmospheric processes. *Frontiers in Earth Science*, 12. <https://doi.org/10.3389/feart.2024.1427783>
- McKinney, W. (2017). *Python for Data Analysis* (2nd ed.). O'Reilly Media, Inc.
- Miao, L., Tang, S., Ren, Y., Kwan, M.-P., & Zhang, K. (2022). Estimation of daily ground-level PM<sub>2.5</sub> concentrations over the Pearl River Delta using 1 km resolution MODIS AOD based on multi-feature BiLSTM. *Atmospheric Environment*, 290, 119362. <https://doi.org/10.1016/j.atmosenv.2022.119362>
- Mourre, L., Condom, T., Junquas, C., Lebel, T., E. Sicart, J., Figueroa, R., & Cochachin, A. (2016). Spatio-temporal assessment of WRF, TRMM and in situ precipitation data in a tropical mountain environment (Cordillera Blanca, Peru). *Hydrology and Earth System Sciences*, 20(1), 125–141. <https://doi.org/10.5194/hess-20-125-2016>
- Mu, X., Wang, S., Jiang, P., & Wu, Y. (2023). Estimation of surface ozone concentration over Jiangsu province using a high-performance deep learning model. *Journal of Environmental Sciences*, 132, 122–133. <https://doi.org/10.1016/j.jes.2022.09.032>
- Muthukumar, P., Cocom, E., Nagrecha, K., Comer, D., Burga, I., Taub, J., Calvert, C. F., Holm, J., & Pourhomayoun, M. (2022). Predicting PM<sub>2.5</sub> atmospheric air pollution using deep learning with meteorological data and ground-based observations and remote-sensing satellite big data. *Air Quality, Atmosphere & Health*, 15(7), 1221–1234. <https://doi.org/10.1007/s11869-021-01126-3>
- NASA Earthdata. (2000). *Shuttle Radar Topography Mission (SRTM)*.
- National Center for Atmospheric Research (NCAR). (n.d.). *Weather Research and Forecasting (WRF) Model*. Retrieved August 19, 2023, from <https://www2.mmm.ucar.edu/wrf/>
- Neha, G. (2013). Network and Complex Systems. *Network and Complex Systems*, 3(1). [https://d1wqtxts1xzle7.cloudfront.net/31377814/Artificial\\_Neural\\_Network-libre.pdf?1392407140=&response-content-disposition=inline%3B+filename%3DIISTE\\_May\\_30th\\_Edition\\_Peer\\_reviewed\\_art.pdf&Expires=1683503661&Signature=DvZcW0YpJyuWuDFHUmhmhfAJjPsVHDxZoADIwy1BZkhr2eIdzr7Fe3doirv9QJ2slhviHqcFho1tTlorepTDjaLDJx-MHDAe0nnchZnvjN2snaY-UTT8H~qAe2naKqHYtJWWV-2hWnJDQujV3aUb-w6p2725oICuJ679n8-tA5LcaQ01ZcgY2cObcNM3jacJs-](https://d1wqtxts1xzle7.cloudfront.net/31377814/Artificial_Neural_Network-libre.pdf?1392407140=&response-content-disposition=inline%3B+filename%3DIISTE_May_30th_Edition_Peer_reviewed_art.pdf&Expires=1683503661&Signature=DvZcW0YpJyuWuDFHUmhmhfAJjPsVHDxZoADIwy1BZkhr2eIdzr7Fe3doirv9QJ2slhviHqcFho1tTlorepTDjaLDJx-MHDAe0nnchZnvjN2snaY-UTT8H~qAe2naKqHYtJWWV-2hWnJDQujV3aUb-w6p2725oICuJ679n8-tA5LcaQ01ZcgY2cObcNM3jacJs-)

OTygGf~XqmnwsSQ~zL5xrWuw5~LdUI3Jh57loe0VaK4jloFB6Ui7CfMhZHiM6SCO  
ZJPL4GlcRpRlFfMjb0mjukxo8jU11WFVqi4HKP0t4WbZ-  
LxcVDun5GuJ23cvpQsqDgCc~JBmIQf-T~auj9ZA\_\_&Key-Pair-  
Id=APKAJLOHF5GGSLRBV4ZA

- Ochoa, A., Pineda, L., Crespo, P., & Willems, P. (2014). Evaluation of TRMM 3B42 precipitation estimates and WRF retrospective precipitation simulation over the Pacific–Andean region of Ecuador and Peru. *Hydrology and Earth System Sciences*, 18(8), 3179–3193. <https://doi.org/10.5194/hess-18-3179-2014>
- Ouyang, X., Yang, Y., Zhang, Y., Zhou, W., & Guo, D. (2023). Dual-channel spatial–temporal difference graph neural network for PM<sub>2.5</sub> forecasting. *Neural Computing and Applications*, 35(10), 7475–7494. <https://doi.org/10.1007/s00521-022-08036-0>
- Potter, E. R., Fyffe, C. L., Orr, A., Quincey, D. J., Ross, A. N., Rangecroft, S., Medina, K., Burns, H., Llacza, A., Jacome, G., Hellström, R. Å., Castro, J., Cochachin, A., Montoya, N., Loarte, E., & Pellicciotti, F. (2023). A future of extreme precipitation and droughts in the Peruvian Andes. *Npj Climate and Atmospheric Science*, 6(1), 96. <https://doi.org/10.1038/s41612-023-00409-z>
- Powers, J. G., Klemp, J. B., Skamarock, W. C., Davis, C. A., Dudhia, J., Gill, D. O., Coen, J. L., Gochis, D. J., Ahmadov, R., Peckham, S. E., Grell, G. A., Michalakes, J., Trahan, S., Benjamin, S. G., Alexander, C. R., Dimego, G. J., Wang, W., Schwartz, C. S., Romine, G. S., ... Duda, M. G. (2017). The Weather Research and Forecasting Model: Overview, System Efforts, and Future Directions. *Bulletin of the American Meteorological Society*, 98(8), 1717–1737. <https://doi.org/10.1175/BAMS-D-15-00308.1>
- Python Organization. (n.d.). *Python*. Retrieved June 2, 2023, from <https://www.python.org/>
- Rasp, S., Pritchard, M. S., & Gentine, P. (2018). Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences*, 115(39), 9684–9689.
- Resifi, S., Aawar, E. Al, Dasari, H. P., Jebari, H., & Hoteit, I. (2025). A novel deep learning approach for regional high-resolution spatio-temporal wind speed forecasting for energy applications. *Energy*, 328, 136356. <https://doi.org/10.1016/j.energy.2025.136356>

- Saavedra, M., Junquas, C., Espinoza, J.-C., & Silva, Y. (2020). Impacts of topography and land use changes on the air surface temperature and precipitation over the central Peruvian Andes. *Atmospheric Research*, 234, 104711.  
<https://doi.org/10.1016/j.atmosres.2019.104711>
- Schumacher, V., Fernández, A., Justino, F., & Comin, A. (2020). WRF High Resolution Dynamical Downscaling of Precipitation for the Central Andes of Chile and Argentina. *Frontiers in Earth Science*, 8. <https://doi.org/10.3389/feart.2020.00328>
- Seneviratne, S. I., Zhang, X., Adnan, M., Badi, W., Dereczynski, C., Di Luca, A., Ghosh, S., Iskandar, I., Kossin, J., Lewis, S., Otto, F., Pinto, I., Satoh, M., Vicente-Serrano, S. M., Wehner, M., & Zhou, B. (2023). Weather and Climate Extreme Events in a Changing Climate. In *Climate Change 2021 – The Physical Science Basis* (pp. 1513–1766). Cambridge University Press. <https://doi.org/10.1017/9781009157896.013>
- Sharma, N., Sharma, R., & Jindal, N. (2021). Machine Learning and Deep Learning Applications-A Vision. *Global Transitions Proceedings*, 2(1), 24–28.  
<https://doi.org/10.1016/j.gltp.2021.01.004>
- Shmueli, G. (2010). To Explain or to Predict? *Statistical Science*, 25(3).  
<https://doi.org/10.1214/10-STS330>
- Spiridonov, V., Jakimovski, B., Spiridonova, I., & Pereira, G. (2019). Development of air quality forecasting system in Macedonia, based on WRF-Chem model. *Air Quality, Atmosphere & Health*, 12(7), 825–836. <https://doi.org/10.1007/s11869-019-00698-5>
- Sri Preethaa, K. R., Muthuramalingam, A., Natarajan, Y., Wadhwa, G., & Ali, A. A. Y. (2023). A Comprehensive Review on Machine Learning Techniques for Forecasting Wind Flow Pattern. *Sustainability*, 15(17), 12914. <https://doi.org/10.3390/su151712914>
- Stasch, C., Scheider, S., Pebesma, E., & Kuhn, W. (2014). Meaningful spatial prediction and aggregation. *Environmental Modelling & Software*, 51, 149–165.  
<https://doi.org/10.1016/j.envsoft.2013.09.006>
- Tie, X., Brasseur, G., & Ying, Z. (2010). Impact of model resolution on chemical ozone formation in Mexico City: application of the WRF-Chem model. *Atmospheric Chemistry and Physics*, 10(18), 8983–8995. <https://doi.org/10.5194/acp-10-8983-2010>
- Tu, J., & Lu, E. (2020). Relative importance of water vapor and air temperature in the interannual variation of the seasonal precipitation: a comparison of the physical and

- statistical methods. *Climate Dynamics*, 54(7–8), 3655–3670.  
<https://doi.org/10.1007/s00382-020-05197-3>
- Tyagi, H., Suran, S., & Pattanaik, V. (2016). Weather - Temperature Pattern Prediction and Anomaly Identification using Artificial Neural Network. *International Journal of Computer Applications*, 140(3), 15–21. <https://doi.org/10.5120/ijca2016909252>
- Weights & Biases. (n.d.). *Weights & Biases*. Retrieved June 3, 2023, from <https://wandb.ai/site>
- Wu, C., He, H., Song, R., Zhu, X., Peng, Z., Fu, Q., & Pan, J. (2023). A hybrid deep learning model for regional O3 and NO2 concentrations prediction based on spatiotemporal dependencies in air quality monitoring network. *Environmental Pollution*, 320, 121075. <https://doi.org/10.1016/j.envpol.2023.121075>
- Wu, C., Yang, S., Jiao, D., Chen, Y., Yang, J., & Huang, B. (2024). Estimation of daily XCO2 at 1 km resolution in China using a spatiotemporal ResNet model. *Science of The Total Environment*, 954, 176171. <https://doi.org/10.1016/j.scitotenv.2024.176171>
- Yáñez-Morrón, G., Gironás, J., Caneo, M., Delgado, R., & Garreaud, R. (2018). Using the Weather Research and Forecasting (WRF) Model for Precipitation Forecasting in an Andean Region with Complex Topography. *Atmosphere*, 9(8), 304. <https://doi.org/10.3390/atmos9080304>
- Yarleque, C., Vuille, M., Hardy, D. R., Timm, O. E., De la Cruz, J., Ramos, H., & Rabatel, A. (2018). Projections of the future disappearance of the Quelccaya Ice Cap in the Central Andes. *Scientific Reports*, 8(1), 15564. <https://doi.org/10.1038/s41598-018-33698-z>
- Yu, M., Masrur, A., & Blaszcak-Boxe, C. (2023). Predicting hourly PM2.5 concentrations in wildfire-prone areas using a SpatioTemporal Transformer model. *Science of The Total Environment*, 860, 160446. <https://doi.org/10.1016/j.scitotenv.2022.160446>
- Zeng, Q., Li, M., Fan, M., Tao, J., Chen, L., Zhang, Y., Zhu, H., & Zhu, Y. (2024). Estimating 1-km PM2.5 concentrations based on a novel spatiotemporal parallel network STMSPNet in the Beijing-Tianjin-Hebei region. *Atmospheric Environment*, 338, 120796. <https://doi.org/10.1016/j.atmosenv.2024.120796>

- Zhang, A., Chen, S., Zhao, F., & Dai, X. (2022). Good environmental governance: Predicting PM2.5 by using Spatiotemporal Matrix Factorization generative adversarial network. *Frontiers in Environmental Science*, *10*. <https://doi.org/10.3389/fenvs.2022.981268>
- Zhang, B., Zou, G., Qin, D., Ni, Q., Mao, H., & Li, M. (2022). RCL-Learning: ResNet and convolutional long short-term memory-based spatiotemporal air pollutant concentration prediction model. *Expert Systems with Applications*, *207*, 118017. <https://doi.org/10.1016/j.eswa.2022.118017>
- Zhang, K., Yang, X., Cao, H., Thé, J., Tan, Z., & Yu, H. (2023). Multi-step forecast of PM2.5 and PM10 concentrations using convolutional neural network integrated with spatial-temporal attention and residual learning. *Environment International*, *171*, 107691. <https://doi.org/10.1016/j.envint.2022.107691>
- Zhang, T., Wang, D., Huang, L., Chen, Y., & Li, E. (2024). Residual Spatiotemporal Convolutional Neural Network Based on Multisource Fusion Data for Approaching Precipitation Forecasting. *Atmosphere*, *15*(6), 628. <https://doi.org/10.3390/atmos15060628>

## Anexos

Lima, 8 de noviembre del 2023

### Validación de “Desarrollo de un modelo de predicción espaciotemporal de variables atmosféricas en zonas de glaciares usando redes neuronales artificiales”

Por medio de la presente acta se hace constar que **Romel Erick Principe Aguirre** ha revisado el proyecto de tesis titulado “Desarrollo de un modelo de predicción espaciotemporal de variables atmosféricas en zonas de glaciares usando redes neuronales artificiales” del estudiante Christian Andre Carhuacho Rodríguez, alumno de la especialidad de Ingeniería Informática en la Pontificia Universidad Católica del Perú. Se realizó la validación y revisión del resultado esperado 1 “Implementar el modelo WRF para la predicción de las variables de estudio” con medio de verificación “Reporte sobre instalación y configuración”.

Atentamente,



**Romel Erick Principe Aguirre**

Lima, 02 de noviembre del 2023

**Validación de “Desarrollo de un modelo de predicción espaciotemporal de variables atmosféricas en zonas de glaciares usando redes neuronales artificiales”**

Por medio de la presente acta se hace constar que Yo, **Edwin Villanueva Talavera**, he revisado el proyecto de tesis titulado “Desarrollo de un modelo de predicción espaciotemporal de variables atmosféricas en zonas de glaciares usando redes neuronales artificiales” del estudiante Christian Andre Carhuancho Rodríguez, alumno de la especialidad de Ingeniería Informática en la Pontificia Universidad Católica del Perú. Se realizó la validación y revisión de los siguientes resultados esperados:

- “Conjunto de datos completo y estructurado” con medio de verificación “Reporte sobre obtención de datos y estadísticas descriptivas”
- “Revisión sistemática sobre arquitecturas de modelos a implementar” con medio de verificación “Reporte sobre elección de arquitecturas”
- “Código fuente de implementación del flujo de entrenamiento de dos modelos algorítmicos” con medio de verificación “Repositorio en GitHub”
- “Dos modelos algorítmicos entrenados” con medio de verificación “Métricas de evaluación de desempeño del entrenamiento y pruebas”
- “Estudio comparativo de evaluación de desempeño entre modelos algorítmicos” con medio de verificación “Reporte del estudio comparativo”
- “Estrategia de evaluación numérica para comparar el modelo seleccionado en O2 y WRF” con medio de verificación “Reporte de selección de estrategia de evaluación numérica”
- “Estudio comparativo del desempeño predictivo y tiempo de procesamiento entre los modelos algorítmicos y WRF” con medio de verificación “Reporte del estudio comparativo”.

Atentamente,



**Dr. Edwin Villanueva**

## **Anexo A: Plan de Proyecto**

- **Justificación**

El estudio de la distribución de variables T, RH, U, V en zonas de glaciares es relevante en el contexto actual, donde el cambio climático es tema de preocupación global. Este proyecto busca brindar herramientas efectivas para comprender y predecir la distribución espacial de las variables atmosféricas en estas áreas sensibles.

La investigación también beneficiará a científicos, expertos en medio ambiente, gestores de recursos naturales y personas responsables de la toma de decisiones, al proporcionarles información precisa sobre los patrones de evolución de las variables de estudio en zonas sensibles, como los glaciares. Esto permitirá la implementación de estrategias de gestión adecuadas, entre ellas medidas preventivas para proteger estos ecosistemas y la salud humana.

El proyecto aborda un problema real al modelar la distribución espacial de las variables de estudio en zonas frágiles. Estos lugares están expuestos a diversas fuentes de contaminación, como emisiones industriales y transporte, que pueden tener impactos negativos en la calidad del aire y acelerar, por ejemplo, procesos de deshielo de los glaciares. Los resultados obtenidos contribuirán a la identificación de medidas de mitigación y políticas ambientales efectivas.

El desarrollo de modelos de dispersión espaciotemporales de variables atmosféricas en zonas de glaciares llenará un vacío de conocimiento en la literatura científica. Existen pocos estudios específicos que aborden este tema, por lo que la investigación proporcionará una base teórica sólida y contribuirá al avance del conocimiento en el campo de ciencias ambientales y la protección de áreas sensibles como los glaciares.

La utilización de redes neuronales artificiales en la modelización de variables atmosféricas en zonas de glaciares representa una metodología innovadora y eficiente. La aplicación de estas técnicas permitirá capturar relaciones complejas entre variables meteorológicas, geográficas y

de emisiones, mejorando la precisión de los modelos y facilitando la toma de decisiones informada.

- **Viabilidad**

**Viabilidad técnica:**

La disponibilidad de datos es viable debido a que el proyecto utiliza simulaciones espaciotemporales de variables atmosféricas del modelo numérico WRF. Este es de código abierto, con disponibilidad permanente una vez instalado localmente en un computador.

Como se ha mostrado en la revisión sistemática, existe un catálogo extenso de estudios científicos que tratan experimentos similares sobre ciencias ambientales, pero con diferentes contextos aplicados. Se tiene acceso a todo el material bibliográfico, por lo cual se podrán consultar los recursos literarios necesarios para diseñar el experimento.

Se tiene acceso a las herramientas adecuadas para desarrollar los modelos, como los lenguajes de programación y librerías mencionadas en la sección de métodos y procedimientos. Se posee conocimiento del uso de estas herramientas debido a una previa especialización en los temas de aprendizaje profundo mediante el autoaprendizaje por parte del autor.

Se cuenta con los equipos tecnológicos necesarios para realizar el proyecto: La computadora personal es propiedad del alumno y el servidor de entrenamiento es proporcionado por el grupo de investigación IA-PUCP, donde el asesor es miembro.

Se cuenta también con la asesoría de un especialista en el área de ciencias ambientales para orientar en cualquier tema adicional que pueda surgir con respecto a la configuración del modelo WRF o del análisis de predicciones de las variables de estudio durante las etapas de entrenamiento y validación.

**Viabilidad temporal:**

Las actividades se podrán concluir antes de la fecha de entrega del proyecto, por lo cual se justifica la viabilidad temporal de este proyecto.

### **Viabilidad económica:**

Se considera que el proyecto es viable económicamente por las siguientes razones:

1. Se utiliza software de uso libre para el desarrollo de los modelos: Python y sus librerías, WRF y GitHub.
2. El servidor de entrenamiento es provisto por el asesor, por lo cual no se debe incurrir en gastos adicionales para la adquisición de este recurso.
3. Los costos asociados a las horas del tesista serán asumidos por el mismo, mientras que las horas de asesoría brindadas por los especialistas son asumidas por el servicio educativo de la universidad.

### **• Alcance**

El presente proyecto corresponde al área de ciencias de la computación, por lo que el énfasis será en los modelos algorítmicos de predicción de variables atmosféricas y no en las especificaciones técnicas de la aplicación ni en el análisis desde el punto de vista de las ciencias ambientales. En este sentido, el proyecto contendrá como parte de la obtención de modelos predictivos temporales y espaciales las fases de un estudio de aprendizaje profundo: Pre-procesamiento de datos, entrenamiento y validación.

El presente proyecto utiliza datos de simulaciones espaciotemporales de variables atmosféricas para zonas de alta montaña utilizando la herramienta WRF para la etapa de entrenamiento y validación de los modelos algorítmicos.

El enfoque de este proyecto será de encontrar el mejor modelo algorítmico basado en aprendizaje profundo para la predicción de variables atmosféricas como parte de estudios comparativos. Como se ha visto en la problemática, los modelos con mejores resultados son

numéricos y requieren de gran capacidad de cómputo, por lo cual se pretende resolver el problema mediante una alternativa con semejante calidad de predicción pero que requiera menor tiempo de procesamiento. Para ello se aplicará modelos basados en aprendizaje profundo, mediante la implementación y evaluación de varios modelos con el fin de encontrar cuál de ellos presenta mejores resultados de predicción y tiempo de procesamiento para el entorno escogido. Dado que se debe tener un porcentaje mínimo aceptable de predicción y un tiempo de procesamiento considerablemente menor al del modelo WRF, en concordancia con lo visto en el estado del arte y lo sugerido por el asesor de tesis, se considerará que los modelos de predicción espacial deberán tener una tasa media de exactitud de al menos 70% similar al modelo WRF en la misma área geográfica y un tiempo de procesamiento inferior a la mitad del tiempo empleado por WRF en el mismo nodo de procesamiento.

Para el estudio comparativo de la predicción espacial se considerarán modelos basados en redes neuronales por el conocimiento preliminar que se tiene en el ámbito de aprendizaje profundo y porque han sido altamente referenciados en la revisión sistemática realizada. Entre las arquitecturas de modelos neuronales a considerar están: LSTM, GRU y CNN-LSTM.

Finalmente, en las predicciones de variables atmosféricas se considerarán las variables temperatura, humedad relativa, componentes U y V del viento.

- **Limitaciones**

La cantidad de sensores en la zona a estudiar es limitada, lo cual puede afectar la precisión y representatividad de los modelos desarrollados en la calidad de predicción espacial realizada si es que se decide incluir esta información en la validación.

La intensidad de las variables seleccionadas en las zonas de glaciares está influenciada por una variedad de factores atmosféricos y topográficos complejos. Modelar con precisión estas interacciones puede resultar desafiante y requerir la consideración de múltiples variables y su

compleja relación. Esto conlleva a un mayor tiempo de disponibilidad del especialista en ciencias ambientales para la interpretación de conceptos ambientales.

El tamaño y representatividad de la muestra de datos utilizada para entrenar y validar los modelos pueden tener un impacto en su rendimiento. Una muestra pequeña o sesgada puede limitar la capacidad de generalización de los modelos y afectar su capacidad para capturar patrones de dispersión a gran escala. Esto requiere de un mayor tiempo de simulación para el armado del conjunto de datos debido al alto requerimiento computacional y de tiempo de procesamiento de WRF, lo que puede afectar la planificación de tareas si ocurren errores en las simulaciones.

El uso de redes neuronales artificiales y el procesamiento de grandes volúmenes de datos pueden requerir recursos computacionales significativos. Limitaciones en el hardware o tiempo de cálculo pueden afectar la capacidad de entrenar y ajustar los modelos de manera eficiente.

Por otro lado, la falta de comparación directa con mediciones reales debido a la disponibilidad de los datos en sensores puede limitar la capacidad de evaluar la precisión y rendimiento de los modelos propuestos.

● **Identificación de los riesgos del proyecto**

Descripción	Síntomas	Probabilidad	Impacto	Severidad	Mitigación	Contingencia
Cambios en los requerimientos del proyecto	Cambios frecuentes en los objetivos o alcance del proyecto, confusiones en las metas a alcanzar.	Moderada	Moderado	Moderada	Establecer una comunicación fluida con el asesor de tesis y otros stakeholders involucrados para asegurar una comprensión clara de los	Evaluar cuidadosamente los cambios propuestos y su impacto en el tiempo y los recursos disponibles. Realizar ajustes en el cronograma y

					requerimientos iniciales.	los recursos según sea necesario.
Falta de acceso a recursos financieros o de hardware	Limitaciones en el acceso a hardware o software especializado, bases de datos o equipos de monitoreo ambiental.	Moderada	Alto	Alta	Realizar una evaluación exhaustiva de los recursos necesarios para el proyecto y su disponibilidad. Buscar fuentes alternativas de financiamiento o colaboraciones con instituciones o empresas que puedan proporcionar los recursos necesarios.	Explorar alternativas de menor costo o recursos alternativos disponibles en caso de falta de acceso a recursos específicos. Considerar la posibilidad de utilizar recursos compartidos o disponibles en línea.
Falta de tiempo para el avance de entregables	Presión temporal, dificultades para cumplir con los plazos establecidos.	Alta	Moderado	Alta	Realizar una planificación detallada que incluya estimaciones realistas de tiempo para cada tarea. Establecer hitos y seguimiento regular del progreso del proyecto.	Revisar y ajustar el cronograma según sea necesario para evitar retrasos significativos. Asignar recursos adicionales si es posible para acelerar la ejecución del proyecto.
Baja precisión de predicciones debido a la	Dificultad para obtener datos representativos y precisos para	Alta	Medio	Alta	Buscar fuentes alternativas de datos, como estudios	Utilizar técnicas de interpolación o generación de

calidad de los datos	entrenar y validar los modelos.				científicos previos o datos obtenidos de organizaciones gubernamentales o ambientales.	datos sintéticos para compensar la falta de datos reales en áreas específicas.
Complejidad de la interacción atmosférica	Dificultad para modelar con precisión las interacciones atmosféricas en los modelos desarrollados.	Media	Alto	Alta	Consultar a expertos en el campo para obtener orientación adicional.	Utilizar enfoques simplificados o técnicas de validación cruzada para evaluar la precisión de los modelos en diferentes escenarios atmosféricos.
Capacidad de cómputo insuficiente	Limitaciones en el hardware o tiempo de cálculo disponibles, lo que puede retrasar el desarrollo y la experimentación.	Media	Alto	Alta	Optimizar el código y utilizar técnicas de paralelización o computación distribuida para acelerar los cálculos. Utilizar servicios en la nube o clusters de cómputo si es necesario.	Priorizar y ajustar los objetivos y las tareas del proyecto para adaptarse a las limitaciones computacionales disponibles.
Dificultad en la evaluación del modelo con datos reales	Limitada disponibilidad de datos de validación y desafíos para evaluar la precisión y el	Media	Alto	Media	Recopilar la mayor cantidad posible de datos de validación en zonas de glaciares.	Realizar un análisis cuidadoso de los resultados obtenidos y discutir las limitaciones y

	rendimiento del modelo.				Utilizar técnicas de comparación indirecta, como el uso de datos de estaciones cercanas o la comparación con modelos existentes en áreas similares.	las implicaciones en la interpretación de los resultados
Ausencia debido a que el tesista se enferma y no puede trabajar en el proyecto durante un período de tiempo	Ausencia del estudiante, falta de avance en las tareas y actividades del proyecto.	Baja	Moderado o	Media	Mantener un cronograma flexible	Ajustar el cronograma una vez que el tesista se recupere.
Mal funcionamiento o daño de los equipos	Incapacidad para obtener datos precisos, desarrollar el modelo y retrasos en la ejecución del proyecto.	Media	Alto	Alta	Realizar un mantenimiento regular de los equipos, tener repuestos disponibles y contar con un plan de respaldo en caso de falla de los equipos.	Reparar o reemplazar los equipos dañados lo más rápido posible y ajustar el cronograma si es necesario.
Pérdida de información	Incapacidad para respaldar los resultados, retrasos en el análisis y reporte de resultados.	Baja	Medio	Media	Implementar un sistema de respaldo y almacenamiento seguro de datos, realizar copias de seguridad regularmente y	Recuperar datos de respaldo o volver a realizar experimentos si es necesario.

					utilizar herramientas de control de versiones.	
Falta de disponibilidad de especialistas	Incapacidad para resolver problemas técnicos, falta de retroalimentación y orientación especializada.	Media	Moderado	Media	Establecer contactos con expertos en el campo, asistir a conferencias o eventos relacionados, y buscar apoyo en comunidades en línea.	Utilizar recursos en línea, tutoriales o foros de discusión para obtener asesoramiento técnico en caso de no tener acceso directo a especialistas.
Falta de conectividad a Internet	Incapacidad para descargar recursos, acceder a información en línea y colaborar con otros investigadores o expertos.	Baja	Bajo	Baja	Contar con una conexión de respaldo, trabajar en modo sin conexión cuando sea posible y anticipar períodos de falta de conectividad.	Utilizar recursos locales, bibliotecas o redes alternativas para acceder a la información o colaborar con otros.

- **Estructura de descomposición del trabajo (EDT)**

A continuación, se mostrará la estructura de desglose del trabajo de la siguiente manera: El objetivo general se ubicará en el nivel 0 y se describirá en forma de párrafo en este documento. El nivel 1 estará formado por los resultados esperados que conducen a la obtención de los objetivos específicos y, en última instancia, el objetivo general. Estos resultados esperados se presentarán en forma de lista utilizando la notación Rx, donde x representa el número correspondiente a cada resultado esperado. Por último, en el nivel 2 se detallarán los

entregables que conforman cada resultado esperado, los cuales se presentarán como una lista asociada a cada resultado esperado.

Objetivo general: Construir un modelo basado en aprendizaje profundo que permita predecir las variables atmosféricas de Temperatura (T), Humedad relativa (RH), componentes zonal (U) y meridional (V) del viento, con una precisión y desempeño comparable con los resultados obtenidos del modelo numérico WRF y con un menor costo computacional.

- R 1. Implementar el modelo WRF para la predicción de las variables de estudio.
  - Reporte sobre instalación y configuración del modelo WRF, aprobado por un especialista en el modelo.
- R 2. Conjunto de datos completo y estructurado.
  - Reporte sobre obtención de datos y estadísticas descriptivas, aprobado por el asesor.
- R 3. Revisión sistemática sobre arquitecturas de modelos a implementar.
  - Reporte sobre elección de arquitectura, aprobado por el experto en inteligencia artificial.
- R 4. Código fuente de implementación del flujo de entrenamiento de dos modelos algorítmicos.
  - Repositorio en GitHub con el código fuente de la implementación de los dos modelos algorítmicos.
- R 5. Dos modelos algorítmicos entrenados.
  - Métricas de evaluación del desempeño del entrenamiento y pruebas de los dos modelos, cuyos valores deben ser mejores al valor medio obtenido de WRF.
- R 6. Estudio comparativo de evaluación de desempeño entre modelos algorítmicos.
  - Reporte del estudio comparativo entre los dos modelos entrenados, aprobado por el experto en inteligencia artificial.

R 7. Estrategia de evaluación numérica para comparar el modelo seleccionado en O2 y WRF.

- Reporte de selección de estrategia de evaluación numérica, aprobado por el experto e inteligencia artificial.

R 8. Estudio comparativo del desempeño predictivo y tiempo de procesamiento entre el modelo algorítmico seleccionado y WRF.

- Reporte del estudio comparativo entre el modelo seleccionado y WRF, aprobado por el experto en inteligencia artificial.

● **Lista de tareas**

<b>Tarea</b>	<b>Duración estimada (en días)</b>
Investigar y comprender el modelo WRF	1
Obtener el software WRF y recursos necesarios	2
Configurar el dominio y la resolución espacial del modelo WRF	2
Configurar los parámetros del modelo WRF	2
Elaborar el reporte sobre instalación y configuración del modelo WRF	2
Reunión con el especialista en el modelo WRF para validación del reporte	1
Ejecutar las simulaciones de WRF en el área a estudiar	14
Obtener estadísticas descriptivas sobre el conjunto de datos armado a partir de las simulaciones del modelo WRF	1
Elaborar el reporte sobre obtención de datos y estadísticas descriptivas	2
Reunión con el asesor para validación del reporte	1
Seleccionar arquitecturas de redes neuronales candidatas de los trabajos revisados en estado del arte	4
Elaboración del reporte sobre elección de arquitectura del modelo a implementar	2
Reunión con el experto en inteligencia artificial para validación del reporte	1
Preparar el entorno de desarrollo	3
Realizar el preprocesamiento de datos de entrada del modelo	4
Implementar los dos modelos algorítmicos	5
Realizar el entrenamiento de cada uno de los dos modelos	7
Crear el repositorio público en GitHub con el código fuente de la implementación y entrenamiento de los dos modelos	1

Reunión con el asesor para validación del proceso de implementación, entrenamiento y retroalimentación sobre resultados obtenidos	1
Obtener las métricas de desempeño predictivo de cada uno de los dos modelos	4
Reunión con el asesor para validación del objetivo de desempeño predictivo de cada modelo y retroalimentación	1
Seleccionar la estrategia de evaluación de experimentos a emplear	4
Elaboración del reporte de estudio comparativo entre los dos modelos entrenados	3
Reunión con el experto en inteligencia artificial para la validación del reporte elaborado	1
Seleccionar estrategia de evaluación numérica de los trabajos revisados en estado del arte	3
Elaborar reporte de selección de estrategia de evaluación numérica	3
Reunión con el experto en inteligencia artificial para la validación del reporte elaborado	1
Obtener métricas del desempeño de procesamiento del modelo seleccionado y WRF	4
Elaborar reporte de estudio comparativo entre el modelo seleccionado y WRF	3
Reunión con el experto en inteligencia artificial para validación del reporte elaborado	1

- **Lista de recursos**

- **Personas involucradas y necesidades de capacitación**

- Christian Andre Carhuancho Rodríguez
- Edwin Rafael Villanueva Talavera
- Christian Pedro Yarlequé Gálvez

- **Materiales requeridos para el proyecto**

- No aplica

- **Estándares utilizados en el proyecto**

- Proceso de construcción de modelos de aprendizaje profundo empleando CRISP-DM.

- **Equipamiento requerido**

- Computadora personal

- Servidor con tarjetas GPU para entrenamiento
- **Herramientas requeridas**
  - WRF
  - Python 3
  - Jupyter notebook
  - Tensorflow / Keras
  - Weights & Biases

● **Costeo del Proyecto**

Ítem	Descripción			Unidad	Cantidad	Valor Unitario (S/.)	Monto Total (S/.)	Monto Acumulado (S/.)
<b>0.</b>	<b>Costo total del proyecto</b>			---	---	---	---	<b>9540</b>
<b>1.</b>	<b>Estudiantes o tesistas</b>			---	---	---	---	<b>5040</b>
1.1.	Christian Carhuacho			Días	84	60	5040	
<b>2.</b>	<b>Otros participantes (en caso aplique)</b>			---	---	---	---	<b>1950</b>
2.1.	Edwin Villanueva			Días	10	150	1500	
2.2.	Christian Yarlequé			Días	3	150	450	
<b>3.</b>	<b>Bienes y equipos</b>	<b>Unid 1</b>	<b>Cant 1</b>	<b>Unid 2</b>	<b>Cant 2</b>			<b>2550</b>
3.1.	Computadoras	Equipo	1	Días	80	10	800	
3.2.	Servidor con tarjeta GPU	Equipo	1	Días	25	70	1750	