

Universidad del Valle de Guatemala
Facultad de Ingeniería
Deep Learning y Sistemas Inteligentes
Departamento de Ciencias de la Computación
Ciclo II, 2024



Fecha de entrega: 11 nov 2024 23:59

Proyecto
**Implementación de LSTMs para la
Predicción de Presión en Ventiladores de Oxígeno**

Adrián Ricardo Flores Trujillo	Carné 21500
Daniel Armando Valdez Reyes	Carné 21240
Emilio José Solano Orozco	Carné 21212
Dáriel Eduardo Villatoro Rodas	Carné 20776
Andrea Ximena Ramírez Recinos	Carné 21874

Guatemala, 11 de noviembre de 2024

1. Introducción

Durante la pandemia de COVID-19 en 2020, la demanda de respiradores artificiales aumentó drásticamente debido a los problemas respiratorios asociados con el SARS-CoV-2. En ese período, surgió la necesidad urgente de diseñar y producir estos dispositivos en masa para que estuvieran disponibles lo antes posible. Las limitaciones principales para lograr este objetivo incluían la precisión y el costo de los materiales necesarios para controlar adecuadamente los respiradores convencionales, lo cual evidenció la necesidad de explorar métodos alternativos (Uddin et al., 2024) [15].

Usualmente, se emplean controladores PID para automatizar el control de flujo y presión en los respiradores y en el sistema respiratorio de los pacientes. Estos dispositivos requieren operadores especializados para su configuración y supervisión clínica, lo cual limita significativamente su accesibilidad. Además, el desarrollo de nuevos métodos para controlar los respiradores y sus controladores PID puede resultar prohibitivamente costoso o subóptimo, lo que motiva la búsqueda continua de soluciones para superar estas limitaciones.

Para abordar este problema, se propone la implementación de dos modelos de redes neuronales recurrentes, específicamente LSTM y BiLSTM, con el objetivo de reemplazar el controlador PID convencional en un respirador artificial. Estos modelos fueron entrenados con datos de flujo y presión generados a partir de una simulación de un respirador artificial que utilizaba un controlador PID convencional. El propósito de ello fue predecir la presión requerida en el siguiente paso de tiempo, permitiendo una toma de decisiones efectiva en el control del flujo de aire y la presión objetivo dentro del sistema.

Los resultados experimentales muestran que los modelos de aprendizaje profundo cumplen satisfactoriamente con la tarea de predecir la presión objetivo dentro del sistema, alcanzando un error medio absoluto de solo 0.24 en el conjunto de datos de validación en el mejor modelo. Estos resultados sugieren que el uso de redes neuronales profundas para el control de respiradores artificiales es una vía prometedora; sin embargo, es probable que se necesite más investigación y el desarrollo de arquitecturas de modelos más complejas antes de su aplicación en ensayos clínicos con pacientes reales.

2. Objetivos

2.1 Objetivo General

El objetivo principal de este proyecto consiste en desarrollar un modelo de red neuronal capaz de predecir y controlar con precisión la presión en ventiladores de oxígeno, contribuyendo a mejorar la eficacia en el soporte respiratorio.

2.2 Objetivos Específicos

- Desarrollar un modelo que logre una mejor capacidad de generalización en la adaptación a diferentes características pulmonares, superando las limitaciones de los controladores PID, los cuales son el estándar actual en la industria.
- Asistir en el control de ventiladores mecánicos mediante el uso de redes neuronales, promoviendo una respuesta más ajustada a las necesidades individuales de cada paciente.
- Seleccionar el modelo más adecuado para abordar la problemática planteada, basado en un análisis comparativo de su desempeño y efectividad.

3. Marco Teórico

3.1 Respiradores Artificiales

Un respirador artificial es un dispositivo mecánico de soporte vital que automatiza el proceso de respiración cuando una persona no puede realizarlo por sí misma. Estos dispositivos se utilizan principalmente durante cirugías o en pacientes con condiciones o lesiones graves que impiden el correcto funcionamiento de sus pulmones. Su función principal es enviar oxígeno directamente a los pulmones del paciente y mantener un ritmo adecuado para la extracción de dióxido de carbono (Cleveland Clinic, 2024). Además, un aspecto fundamental de estos dispositivos es la estabilización de la presión interna en las vías respiratorias y los pulmones. Si un individuo no puede regular estas funciones por sí mismo, éste puede experimentar el colapso de las vías respiratorias o de los alvéolos pulmonares, los cuales son responsables del intercambio de oxígeno y dióxido de carbono en la sangre (Seadler et al., 2023) [12]. El manejo adecuado y automatizado de estos dispositivos es esencial, especialmente en situaciones de alta demanda. Actualmente, se utilizan controladores PID para esta tarea; sin embargo, estos suelen ser inconsistentes y difíciles de mantener. Por ello, se busca una solución más confiable para realizar esta función.

Internamente, un respirador consta de varias partes, las cuales se dividen en dos segmentos: inhalación y expiración. Cada uno de estos segmentos está compuesto por sensores de flujo y presión, los cuales monitorean estas variables en la entrada y salida de aire del dispositivo. Los datos de estos sensores luego son utilizados para alimentar un dispositivo, generalmente un microcontrolador, para que tome decisiones óptimas sobre el control de estos factores, tomando en cuenta el estado y las necesidades del paciente. Luego de tomar una decisión, el microcontrolador es capaz de alterar los valores de entrada y salida al operar válvulas de aire en cada segmento del respirador (Agarwal et al., 2021) [1].

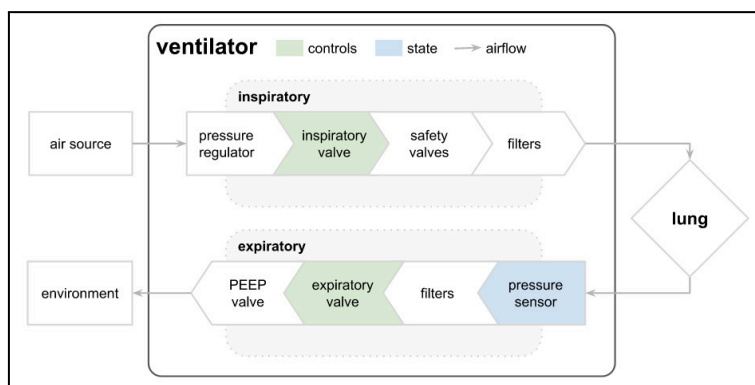


Figura 1 - Diagrama simplificado del flujo funcional de un respirador artificial, obtenida de Howard et al. (2021) [6].

Durante la pandemia causada por el virus SARS-CoV-2 en 2020, la demanda de respiradores aumentó significativamente debido a los problemas respiratorios provocados por la COVID-19. La precisión y la calidad de los materiales requeridos para fabricar un respirador convencional resaltaron la necesidad de desarrollar nuevos diseños más económicos y de producción rápida (Uddin et al., 2024) [15]. Entre estos, Caulfield (2020) [3] menciona un diseño creado por Bill Dally, enfocado en mantener el costo y el tamaño del dispositivo al mínimo. Por otro lado, el diseño de Uddin et al. (2024) [15] ejemplifica un enfoque moderno, basado en nuevas técnicas de transferencia de oxígeno.

Además de la necesidad médica de disponer de suficientes respiradores, estos dispositivos también se utilizan en entornos académicos, donde sirven para entrenar a operadores en el uso

y mantenimiento de estos equipos en condiciones de práctica. Para ello, se emplean habitualmente dispositivos de código abierto, que incluyen simuladores de pulmones artificiales para realizar mediciones y practicar en un entorno seguro. Esta práctica también permite la creación de conjuntos de datos que pueden analizar el rendimiento de los respiradores y, potencialmente, entrenar modelos de aprendizaje profundo para automatizar el ajuste de estos dispositivos (Howard et al., 2021) [6].

6.1 Controladores PID

Un controlador Proporcional-Integral-Derivativo (PID) es un dispositivo empleado en diversas aplicaciones que requieren el control de variables en sistemas industriales, como la regulación de temperatura, flujo, presión y velocidad. Estos controladores son ampliamente utilizados en el manejo de respiradores artificiales, donde se conectan a los sensores de presión y flujo en la entrada y salida de aire de los dispositivos. A partir de estas mediciones, los controladores PID aplican operaciones matemáticas predefinidas para determinar si es necesario ajustar algún parámetro del dispositivo.

El objetivo principal de estos controladores es automatizar la regulación de la salida de un proceso para mantenerla lo más cercana posible a un valor objetivo. Esto se logra a través de un sistema de lazo cerrado, en el cual la salida del sistema se retroalimenta como su propia entrada, sin necesidad de intervención externa (Omega Engineering, s.f.) [4].

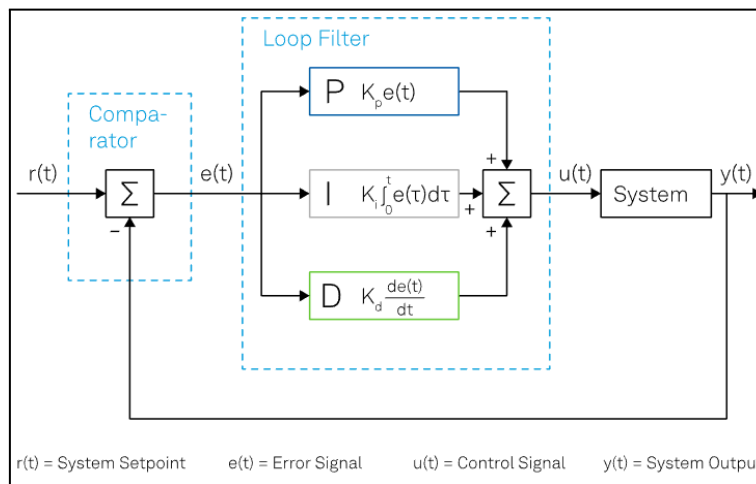


Figura 2 - Diagrama simplificado de los componentes internos de un controlador PID, obtenida de Molineux (2023) [2].

Estos dispositivos están diseñados para operar con un conocimiento mínimo del modelo o sistema al que se aplican. Para lograr esto, como se muestra en la Figura 2, se utilizan tres componentes clave: un componente proporcional, un componente integrador y un componente derivador. Estos componentes procesan la señal de entrada del sistema, y sus resultados se suman para formar la señal de salida del controlador, que luego se retroalimenta al sistema.

Además, el controlador incluye un comparador que recibe la señal de entrada y prepara los datos para que sean procesados por los otros componentes. Todo este proceso se describe mediante la siguiente ecuación:

$$u(t) = u_p(t) + u_i(t) + u_d(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t)$$

Donde K_p , K_i y K_d corresponden a los coeficientes operativos relacionados al componente de proporcionalidad, integrador y diferenciador respectivamente. Mediante la aplicación de esta operación en cada intervalo de tiempo, el sistema es capaz de minimizar el error con respecto al objetivo establecido y la entrada del sistema (Molineux, 2023) [2].

En el contexto de un respirador artificial, los controladores PID se conectan a los sensores de presión y flujo y ajustan automáticamente las válvulas de entrada y salida de aire. Esto garantiza que el paciente mantenga niveles seguros de presión en las vías respiratorias y en los pulmones. Para lograrlo, los operadores deben ajustar manualmente los coeficientes de los componentes del controlador y luego comprobar su correcto funcionamiento en un entorno de práctica antes de emplearlo en una situación real.

7.1 Uso de Modelos de Deep Learning en Respiradores Artificiales

Según lo expuesto anteriormente, se observa que la automatización de los procesos en un respirador artificial requiere operadores capacitados y una vigilancia constante del desempeño de los controladores PID. Además, la configuración y preparación de estos controladores puede ser innecesariamente compleja. Para abordar este problema, se puede replantear como un desafío de control en el que un modelo de aprendizaje profundo es entrenado con datos recolectados de respiradores reales y simulaciones, con el fin de automatizar el control del flujo en estos dispositivos (Diao, 2024).

Los estudios de Suo et al. (2021) muestran que los modelos de aprendizaje profundo especializados en diferentes entornos y dispositivos tienden a presentar un error menor en relación al objetivo, en comparación con los controladores PID. Esto sugiere que estas arquitecturas pueden ser una alternativa viable en situaciones reales, proporcionando resultados potencialmente más seguros y confiables. Los controladores PID, por otro lado, son más susceptibles a ajustes constantes y sobre ajustes respecto a su objetivo.

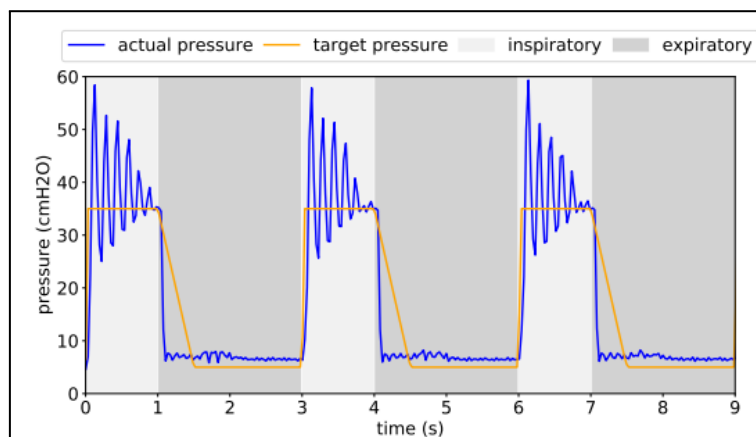


Figura 3 - Presión manejada por un PID en comparación con la presión objetivo dentro de un entorno de prueba, obtenida de Suo et al. (2021)

Este enfoque requiere un pre-procesamiento de datos específico, que puede presentar diversas fuentes de error debido a la naturaleza de los respiradores. Una de estas fuentes es la asunción de un comportamiento de fluidos ideales, en la que se descarta la influencia del flujo y la turbulencia al simplificar las dinámicas de los pulmones y los componentes del respirador. Además, se asumen cambios instantáneos en la presión del sistema; aunque estos cambios ocurren en un intervalo muy pequeño de tiempo, dicha suposición no es

completamente realista. Esto conlleva el riesgo de que el modelo pueda estar ligeramente desfasado respecto a la situación real (Suo et al., 2021).

Al igual que Suo et al. (2021) y el presente estudio, Diao et al. (2024) implementaron con éxito un modelo de red neuronal recurrente (RNN) para predecir la presión necesaria en un paso de tiempo específico en un respirador artificial. Este modelo recurrente, entrenado con datos simulados obtenidos de un controlador PID y un respirador propio conectado a un pulmón falso, logró un error medio absoluto de 0.1878 con respecto al objetivo en un paso de tiempo determinado.

4. Análisis Exploratorio de Datos de Ventiladores de Oxígeno: Patrones y Factores Influyentes

6.1 Caracterización del Conjunto de Datos

4.1.1 Origen y Recopilación de los Datos

Los datos empleados en este estudio se obtuvieron de la plataforma [Kaggle](#), mediante su [página oficial](#), y corresponden a registros experimentales de un ventilador mecánico de código abierto. El sistema experimental consiste en un ventilador modificado conectado a un pulmón artificial, simulado mediante un fuelle.

Este sistema cuenta con dos variables de control principales: la primera es una variable continua, que varía entre 0 y 100, representando el porcentaje de apertura de la válvula solenoide inspiratoria que regula la entrada de aire al pulmón. La segunda variable es binaria y controla el estado de la válvula respiratoria, indicando si está abierta (1) o cerrada (0) para permitir la salida de aire.

El conjunto de datos consiste en múltiples series temporales correspondientes a ciclos respiratorios, donde cada serie representa una respiración de aproximadamente 3 segundos. Cada fila dentro de estas series temporales representa un paso de tiempo en el ciclo respiratorio, proporcionando las dos señales de control, la presión resultante en las vías respiratorias y atributos relevantes del pulmón. Esta información se deriva de la documentación proporcionada por [Google Brain](#) sobre el conjunto de datos.

El conjunto de datos presenta una escala considerable, abarcando más de seis millones de registros de pacientes. Cada observación individual está compuesta por ocho variables, las cuales serán detalladas en la siguiente sección.

4.1.3 Análisis de Variables y Estructura del Conjunto de Datos

Como se mencionó previamente, cada observación en el conjunto de datos consta de ocho variables. La primera variable, denominada **id**, corresponde a un identificador único asignado a cada paso de tiempo en el conjunto de datos. También se incluye un identificador exclusivo para cada ciclo de respiración, **breath_id**, que permite diferenciar entre las distintas respiraciones.

Adicionalmente, dentro de las observaciones, se encuentra la variable **R**, que indica la resistencia al flujo de aire en las vías respiratorias (medida en cmH₂O por litro por segundo). Este valor es análogo a la resistencia experimentada al intentar inflar un globo a través de una pajilla: una pajilla más estrecha (alto valor de R) aumenta la dificultad de inflar el globo, mientras que una más ancha (bajo valor de R) facilita el flujo de aire. La variable **C** representa la *compliance pulmonar* (medida en mililitros por cmH₂O), e indica la flexibilidad del pulmón, expresando el cambio de volumen pulmonar en respuesta a cambios en la presión aplicada. Siguiendo el ejemplo del globo, un globo más delgado (alto valor de

C) se expande con mayor facilidad, mientras que uno más grueso (bajo valor de C) requiere mayor presión para inflarse.

En el conjunto de datos también se incluye la variable **time_step**, que representa el instante específico en el que se realiza cada medición dentro del ciclo respiratorio, así como las variables de control **u_in** y **u_out**. La variable **u_in** regula la válvula de entrada de aire (inspiración) al pulmón, y sus valores varían de 0 a 100, donde 0 indica que no se permite la entrada de aire y 100 representa una apertura máxima de la válvula. Por su parte la variable **u_out** controla la válvula de salida de aire (expiración), y puede tomar valores de 0 (sin salida de aire) o 1 (salida de aire permitida).

Finalmente, la variable de respuesta **pressure** mide la presión en las vías respiratorias dentro del circuito respiratorio, expresada en cmH₂O. Esta variable refleja la presión ejercida en el aire de las vías respiratorias durante cada ciclo respiratorio. Todos estos detalles de las variables utilizadas se resumen de forma concisa en el [Cuadro 1 - Descripción General de Variables del Conjunto de Datos Utilizado](#).

Nombre	Descripción	Tipo
id	Identificador único.	Cualitativa (Nominal)
breath_id	Identificador de ciclo de respiración.	Cualitativa (Nominal)
R	Resistencia de las vías respiratorias.	Cualitativa (Nominal)
C	Cumplimiento del pulmón.	Cualitativa (Nominal)
time_step	Marca de tiempo de cada medición.	Cuantitativa (Continua)
u_in	Entrada de aire en la válvula.	Cuantitativa (Continua)
u_out	Salida de aire en la válvula.	Cualitativa (Nominal)
pressure	Presión en las vías respiratorias.	Cuantitativa (Continua)

Cuadro 1 - Descripción General de Variables del Conjunto de Datos Utilizado

4.2 Limpieza y Preprocesamiento de Datos

Es relevante señalar que, en general, el conjunto de datos estaba en muy buenas condiciones, por lo que no se requirieron operaciones de limpieza como tal. No obstante, por razones de documentación y transparencia, se incluirán en este documento las evidencias que respalden dicha afirmación. Por otro lado, sí se consideró necesario realizar un preprocesamiento específico de algunas de las columnas. Los detalles de este proceso se explicarán más adelante en esta sección.

Primero, se llevó a cabo un breve análisis para tratar de identificar posibles datos faltantes. De manera general, se realizó un conteo de todas las observaciones que presentaban valores ausentes en alguna de las columnas o variables. A continuación, se presenta un cuadro que resume los resultados obtenidos.

Variable	Conteo de Valores Ausentes
id	0
breath_id	0
R	0
C	0
time_step	0
u_in	0
u_out	0
pressure	0

Cuadro 2 - Resultado de Análisis de Datos Faltantes.

Como se puede observar, dado que no se encontraron columnas con valores ausentes, no fue necesario aplicar ningún tipo de imputación en el conjunto de datos. Todas las variables están completamente pobladas, lo que aseguró la integridad del conjunto de datos para el análisis que se realizó posteriormente.

Otra posible preocupación que tiende a surgir es la presencia de valores duplicados, ya que pueden introducir sesgos tanto en el análisis exploratorio como en cualquier modelo predictivo que se pretenda desarrollar. Sin embargo, al igual que con los valores ausentes, no se encontraron datos duplicados en el conjunto de datos. Por ende, no se vio la necesidad de implementar estrategias de eliminación o consolidación de estos.

Por otro lado, la identificación y tratamiento de datos atípicos es una etapa fundamental en el pre-procesamiento de datos. Si bien estos valores pueden afectar negativamente la capacidad de generalización de los modelos, su eliminación no siempre es la solución más adecuada. En el contexto de este estudio, se ha optado por un análisis cuidadoso de los datos atípicos, considerando su potencial impacto y la información valiosa que podrían llegar a aportar.

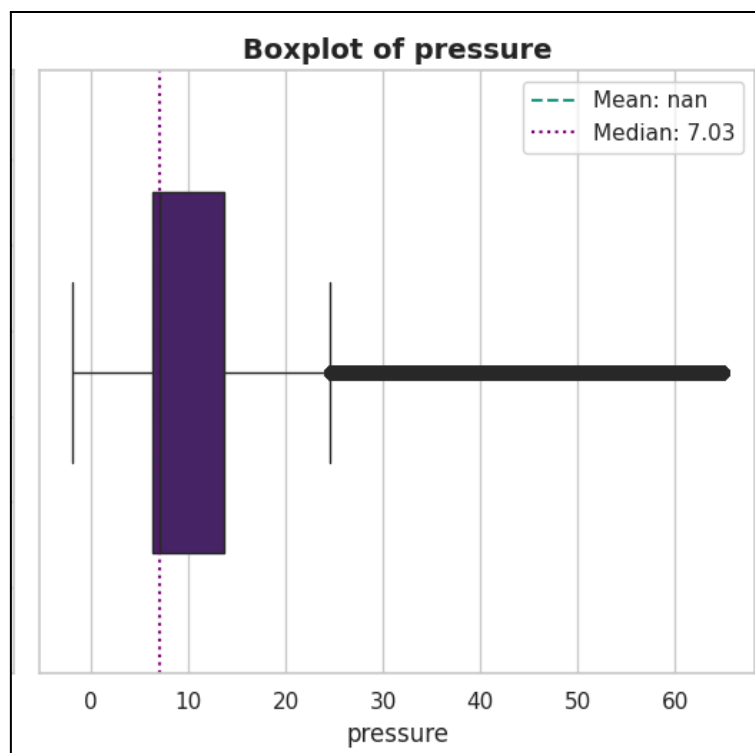


Figura 4 - Diagrama de Caja y Bigotes para Variable 'pressure'.

Solamente dos de las ocho variables en el conjunto de datos poseen datos atípicos, siendo estas `pressure` y `u_in`. Enfocándose estrictamente en la variable objetivo, como se puede observar en la [Figura 4 - Diagrama de Caja y Bigotes para Variable 'pressure'](#), la cantidad de datos atípicos es moderadamente alta. No obstante, se tomó la decisión de permanecer y trabajar con estos datos por las razones que se mencionan a continuación. Se cree que estos valores corresponden a eventos significativos en los patrones respiratorios, como respuestas de presión extrema bajo ciertas condiciones de resistencia o cumplimiento, por que el mejor enfoque es conservarlos. Además, se cree que estos datos podrían reflejar respuestas fisiológicas críticas que podrían mejorar la capacidad del modelo para comprender este tipo de escenario “inusual” pero de gran importancia.

Finalmente, se decidió eliminar la columna `id` del conjunto de datos, ya que esta variable únicamente proporciona un identificador único para cada registro de tiempo, sin aportar información significativa para la caracterización del ciclo respiratorio o para el modelado de la presión en las vías respiratorias.

4.3 Visualización de Datos y Análisis de Distribución

	<code>breath_id</code>	<code>time_step</code>	<code>u_in</code>	<code>pressure</code>
Conteo	6,036,000	6,036,000	6,036,000	6,036,000
Media	62,839	NaN	62.84	NaN
Desviación Estándar	36,335	0	0	0
Valor Mínimo	1	0	0	-1.90
Primer Cuartil (25%)	31,377	0.64	3.94	6.33
Segundo Cuartil (50%)	62,766	1.31	4.39	7.03
Tercer Cuartil (75%)	94,301	1.97	4.98	13.64
Valor Máximo	125,749	2.94	100.00	64.81

Cuadro 3 - Resultado de Estadísticas de Variables Cuantitativas.

El cuadro anteriormente colocado muestra estadísticas descriptivas importantes para cada una de las variables numéricas del conjunto de datos. En primer lugar y en relación con las señales de control, la variable `time_step` tiene un rango relativamente extenso desde 0 hasta 2.93, parece ser que la respiración más larga hallada en el conjunto de datos, y refleja la secuencia temporal de las mediciones en cada respiración. La señal de control `u_in` tiene un valor promedio de 3.94 en el cuartil 25%, lo que sugiere que la entrada de aire se encuentra generalmente en un rango medio. En el cuartil 75%, `u_in` aumenta, alcanzando 4.98 lo que indica una mayor apertura de las válvulas de entrada.

Finalmente, la variable `pressure` muestra una media de 7.03 y un valor máximo de 64.81, lo que refleja fluctuaciones en la presión dentro del sistema respiratorio, con valores tanto bajos como altos durante los ciclos de respiración. Se cree que es posible inferir que el sistema respiratorio presenta una considerable variabilidad en los parámetros de control y las condiciones respiratorias.

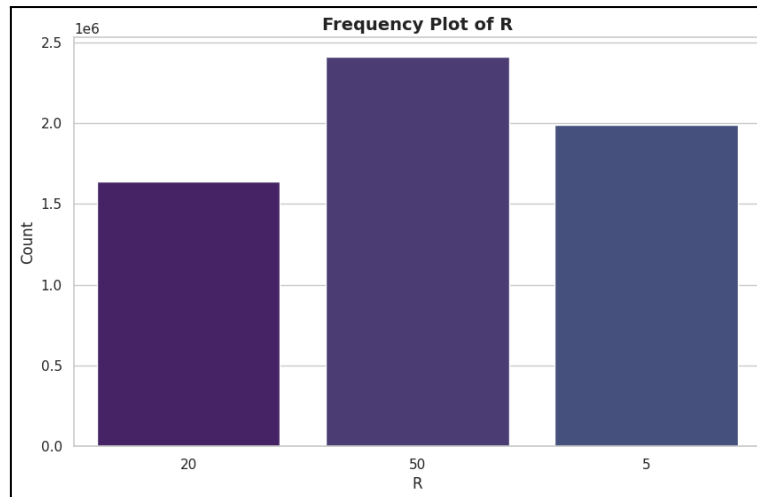


Figura 5 - Gráfico de Frecuencia para Variable 'R'.

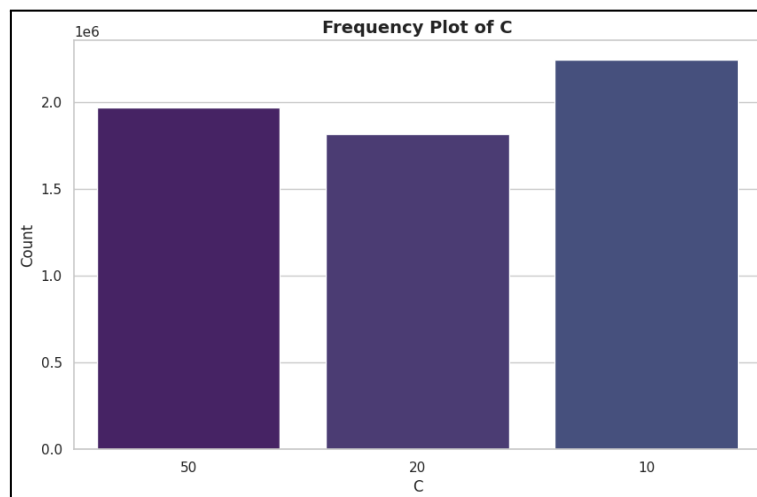


Figura 6 - Gráfico de Frecuencia para Variable 'C'.

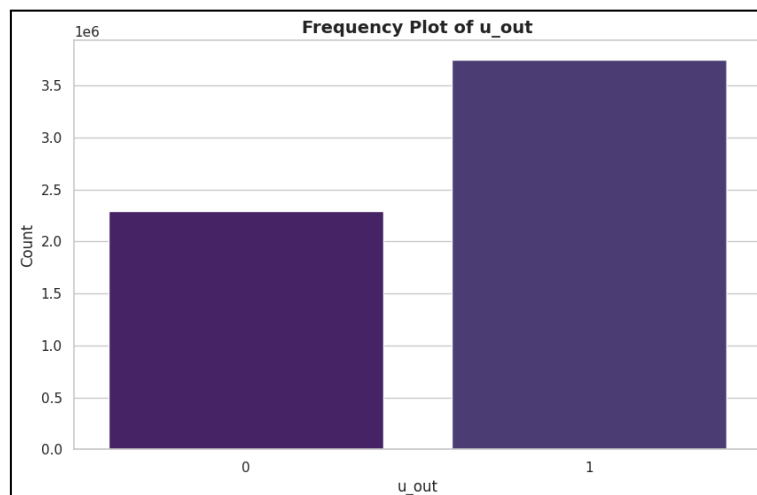


Figura 7 - Gráfico de Frecuencia para Variable 'u_out'.

En cuanto a las variables cualitativas, se observa un notable desbalance en la distribución de las categorías de la variable `u_out`, en la que predomina el valor "1", que indica "salida de aire permitida". Este desbalance sugiere que la salida de aire ocurre con mayor frecuencia

durante los ciclos respiratorios, lo que podría estar relacionado con un patrón general de espiración activa en el sistema.

Por otro lado, la variable C, que representa el compliance pulmonar, exhibe una distribución más equilibrada, lo que es favorable para el análisis, ya que permite contar con una muestra diversa de pulmones con características variadas de flexibilidad. En contraste, la variable R, que mide la resistencia al flujo de aire, presenta una distribución sesgada hacia muestras de pulmones con vías respiratorias más restrictivas.

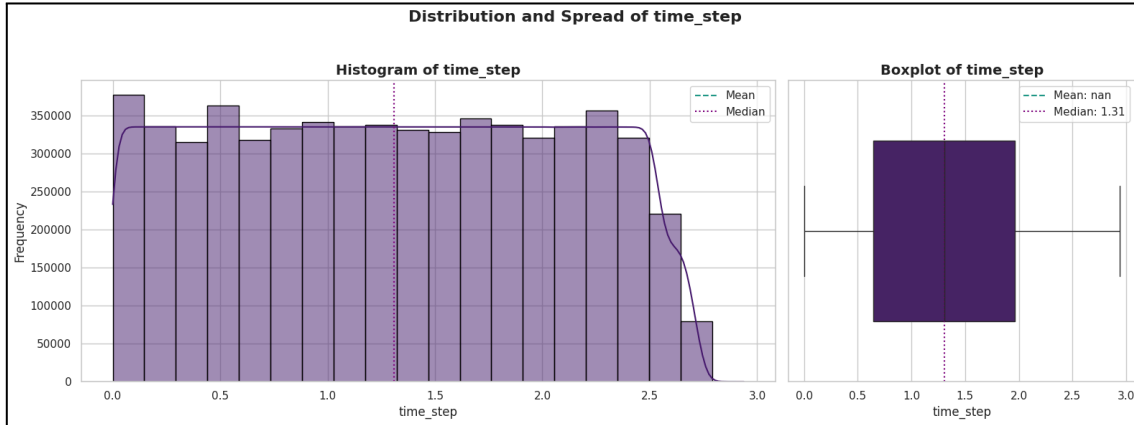


Figura 8 - Histograma y Diagrama de Caja y Bigotes para Variable 'time_step'.

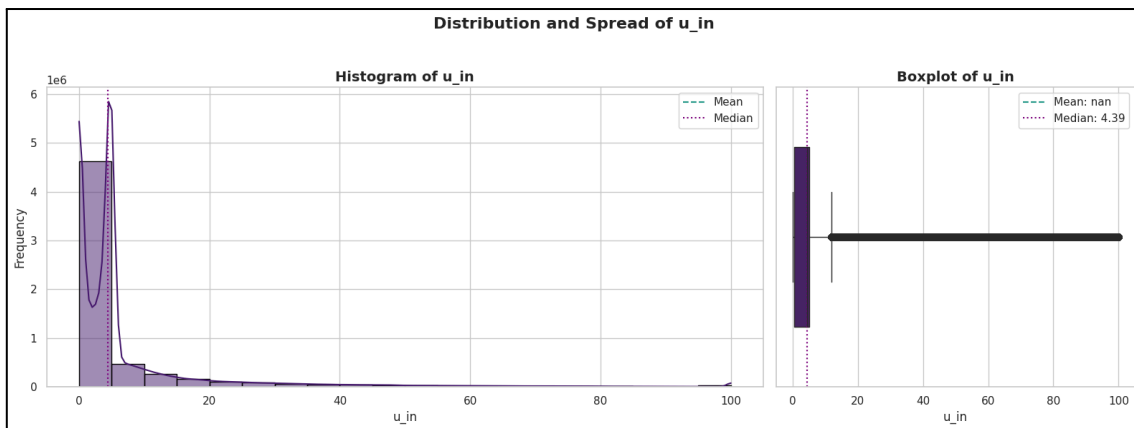


Figura 9 - Histograma y Diagrama de Caja y Bigotes para Variable 'u_in'.

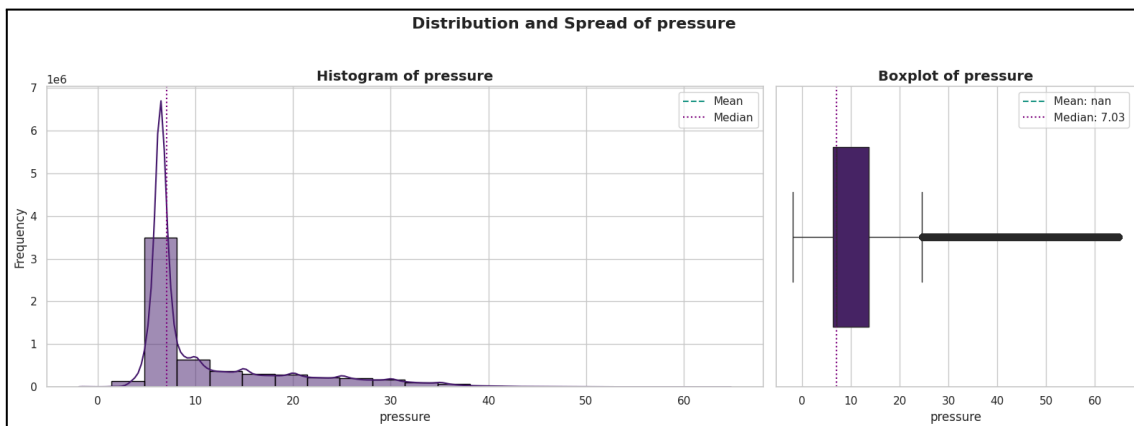


Figura 10 - Histograma y Diagrama de Caja y Bigotes para Variable 'pressure'.

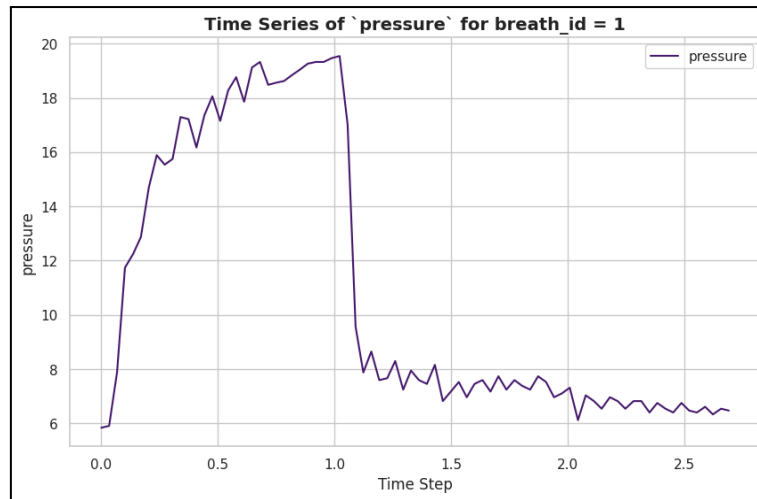


Figura 11 - Diagrama de Serie de Tiempo para Variable 'pressure' (breath_id = 1).



Figura 12 - Diagrama de Serie de Tiempo para Variable 'u_in' (breath_id = 1).

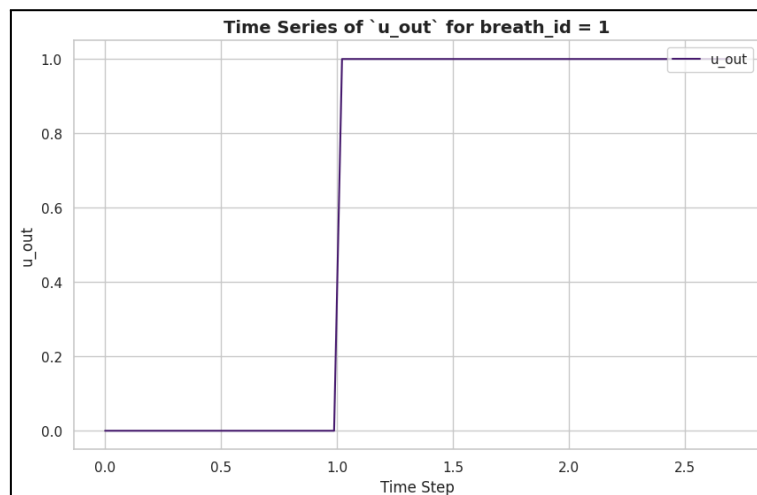


Figura 13 - Diagrama de Serie de Tiempo para Variable 'u_out' (breath_id = 1).

El análisis de las series de tiempo revela patrones bastante interesantes en lo que respecta a la dinámica respiratoria. Inicialmente, se observa un pico en la presión administrada, seguido de una disminución gradual. Paralelamente, la válvula inspiratoria presenta un flujo de aire

elevado al inicio del ciclo, reduciéndose progresivamente hasta su cierre. Este patrón sugiere una fase inspiratoria inicial, seguida de una fase espiratoria caracterizada por un flujo de aire decreciente y una presión descendente. Finalmente, otra observación curiosa es que la válvula parece ser activada casi siempre después de 1 segundo exacto, por lo general.

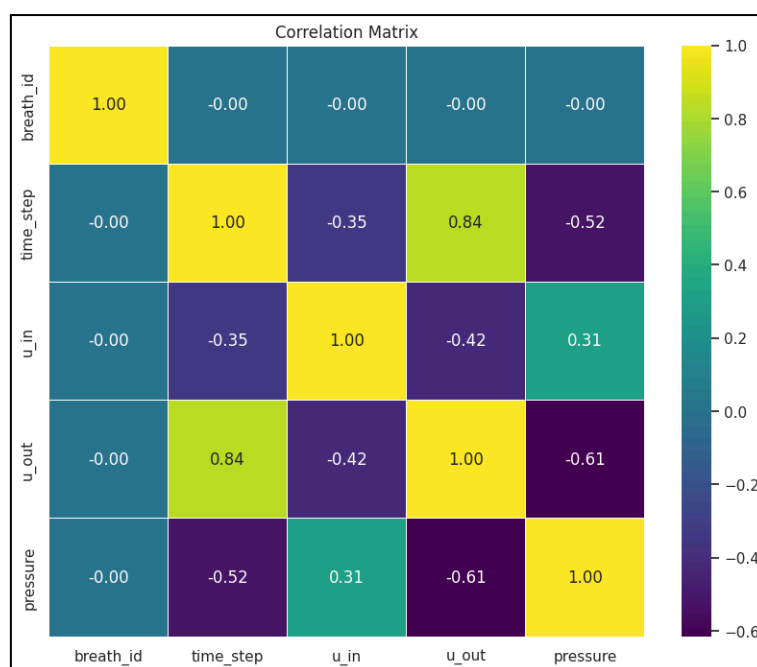


Figura 14 - Mapa de Correlación entre Variables Cuantitativas

Por su parte, el análisis de correlación reveló una relación significativa entre las variables independientes u_{out} , $time_step$ y la variable dependiente $pressure$. Se observó una correlación negativa moderada entre u_{out} y $pressure$ ($r < -0.50$), indicando que a medida que aumenta la señal de salida de la válvula, disminuye la presión intrapulmonar. Asimismo, se detectó una fuerte correlación positiva entre $time_step$ y u_{out} ($r > 0.80$), sugiriendo una relación directa entre el tiempo y la apertura de la válvula.

Finalmente, se encontró una consistencia en la recopilación de datos, con 80 pasos de tiempo por secuencia respiratoria, que indica una frecuencia de muestreo uniforme. Esta regularidad temporal permite aplicar técnicas de ingeniería de características basadas en **lags**, como diferenciaciones, para capturar patrones temporales y dependencias entre observaciones consecutivas. Además, la ventana fija de 80 pasos permite aplicar también estadísticas móviles, tales como medias móviles o desviaciones estándar.

5. Metodología

6.1 Aplicación de Redes Neuronales Recurrentes de Memoria a Largo Plazo (LSTM) para el Análisis de Ciclos de Respiración

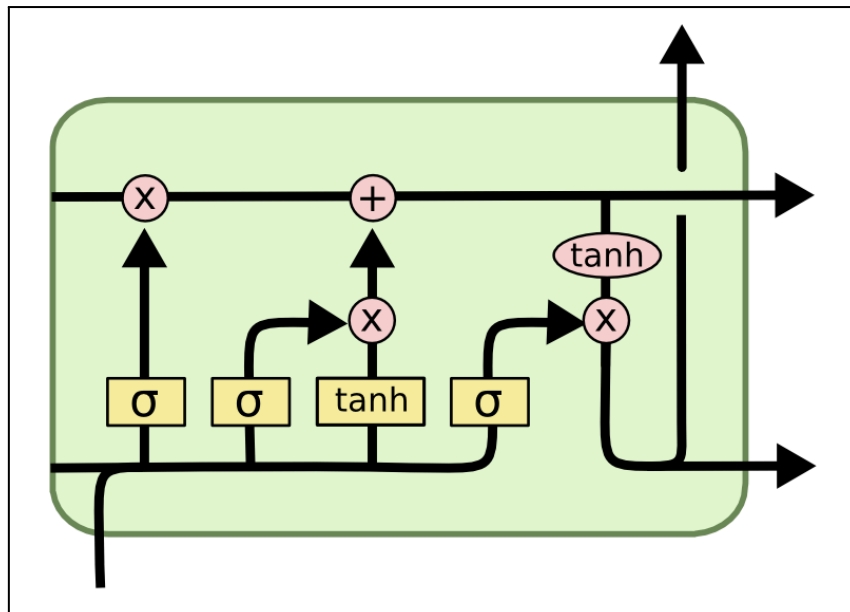


Figura 15 - Diagrama de Arquitetura de Modelo LSTM

El análisis exploratorio de los datos revela una evidente estructura temporal, lo que sugiere que el problema en cuestión se clasifica como una **serie de tiempo**. Dada la capacidad de las LSTM para modelar dependencias de largo plazo y capturar patrones complejos en secuencias, se propone su implementación como la estrategia más eficiente para abordar esta problemática. La elección de este modelo también se justifica en el uso de este mismo tipo de arquitectura en estudios con el mismo propósito, tales como el ya antes mencionado realizado por Suo et al. (2021) [12]. Sin embargo, se considera importante comprender el origen detrás de esta arquitectura, así como sus componentes para argumentar aún más su implementación en este trabajo.

Las Redes Neuronales Recurrentes, también conocidas como RNNs, fueron introducidas por Rumelhart et al. (1985). Estos modelos son principalmente utilizados para trabajar con datos secuenciales, como series de tiempo y el lenguaje natural. Esto se debe a que tienen la capacidad de procesar información a lo largo del tiempo, aprovechando tanto las entradas actuales como las pasadas para realizar sus respectivas predicciones, es por esto que se planea el uso de una arquitectura con la misma “base” que la de este tipo.

Sin embargo, las RNN tradicionales a menudo tienen dificultades para capturar dependencias a largo plazo debido al problema del gradiente desvaneciente. Para abordar esta limitación, se introdujeron las LSTMs. Estas emplean un mecanismo de celda de memoria, controlado por puertas de entrada, olvido y salida, que se explicarán más adelante, para almacenar y recuperar información de manera selectiva. De esta forma, se permite que las LSTMs aprendan y modelen patrones temporales más complejos que aquellos que puede modelar una simple RNN (Petneházi, G., 2019) [10].

En cuanto a su arquitectura, a muy grandes rasgos, poseen una puerta de entrada que regula el flujo de nueva información. Así mismo, tienen otro componente llamado la puerta de olvido, que determina qué información se descarta y qué información se queda. Finalmente, tienen una puerta de salida que controla la información que se emite (Schmidt, 2019) [9].

5.2 Aplicación de LSTMs Bidireccionales para el Análisis de Ciclos de Respiración

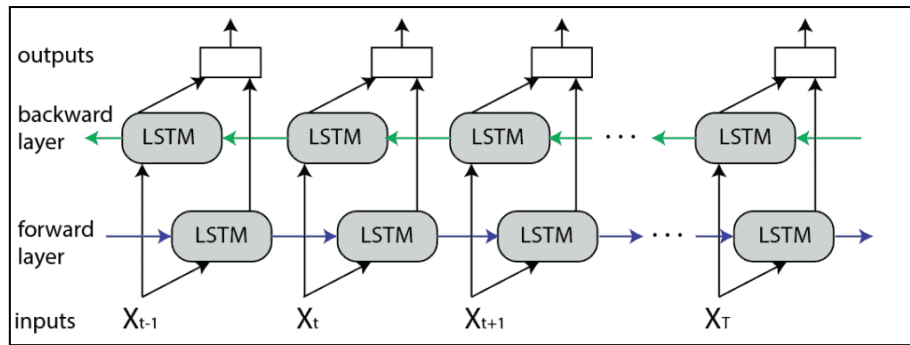


Figura 16 - Diagrama de Arquitectura de Modelo LSTM Bidireccional

Las LSTM Bidireccionales (BiLSTM) son una arquitectura de red neuronal recurrente capaz de procesar datos secuenciales tanto hacia adelante como hacia atrás. Esto permite a las aprender dependencias a largo plazo más complejas en los datos secuenciales que las LSTM tradicionales, que solo pueden procesar datos secuenciales en una dirección. Están compuestas por dos redes LSTM, una que procesa la secuencia de entrada en la dirección hacia adelante y otra que procesa la secuencia de entrada en la dirección hacia atrás. Las salidas de las dos redes LSTM se combinan para producir la salida final (Zvornicanin & Zvornicanin, 2024) [16].

5.3 Descripción de la Solución Propuesta

Dado que se ha definido el uso de una arquitectura LSTM para el presente estudio, se describe a continuación la solución que se pretende llevar a cabo a partir del uso de este tipo modelo. Para lograr predicciones precisas, se requiere un preprocesamiento de los datos, de modo que el modelo disponga de información significativa, y así aprovechar al máximo el uso de LSTMs. Debido a la naturaleza de la información disponible, se ha optado por aplicar técnicas que permitan el cálculo de métricas móviles y de retrasos (lags), lo cual facilita la captura de patrones temporales que son relevantes para la arquitectura seleccionada, que depende de la información histórica para realizar sus predicciones.

Entre las métricas adicionales propuestas se encuentran las características de suma acumulada, tales como **u_in_cumsum** y **area_cumsum**. El uso de este tipo de métrica en el análisis de series temporales es importante, ya que permite capturar el efecto acumulado de una variable a lo largo del tiempo. Por ejemplo, **u_in** representa la entrada de control para la válvula solenoide de inspiración y, a medida que se acumula, podría influir en el comportamiento general de las observaciones.

Por otro lado, también se propone la generación de retrasos. La inclusión de estos en las variables de entrada es una práctica común en el modelado de series temporales con LSTM. Permiten al modelo aprender las relaciones causales entre las observaciones pasadas y presentes, mejorando así la precisión de las predicciones. En este trabajo, se pretende generar cuatro retrasos para las variable **u_in**, **R** y **C**. Tras un análisis de otros trabajos similares y considerando las características del conjunto de datos, se ha determinado que cuatro retrasos son suficientes para capturar las dependencias temporales más relevantes, sin comprometer la eficiencia computacional del modelo. Es importante mencionar que para el modelo bidireccional, estos retrasos se tuvieron que realizar tanto hacia adelante como hacia atrás (lags negativos).

Finalmente, en lo que respecta a los aspectos más relevantes del pre procesamiento, con el objetivo de capturar las dinámicas no estacionarias y los cambios abruptos observados en la variable **u_in**, se ha decidido aplicar técnicas de diferenciación. En el EDA se observa cómo

se presentan comportamientos altamente variables, con picos y caídas repentinas de los valores (referirse a [Figura 12 - Diagrama de Serie de Tiempo para Variable 'u_in' \(breath_id = 1\)](#)). Aplicar diferenciación permite abordar esto, y le da la oportunidad al modelo de comprender este tipo de comportamiento de una forma más profunda.

Seguidamente, en la solución planteada, simplemente se aplican técnicas de normalización de datos, así como estandarizados para que el modelo sea menos susceptible a las variaciones extremas, como algunas que se vieron en la variable u_in, por ejemplo.

Model: "sequential"		
Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 80, 128)	86016
dropout (Dropout)	(None, 80, 128)	0
batch_normalization (Batch Normalization)	(None, 80, 128)	512
lstm_1 (LSTM)	(None, 80, 64)	49408
dropout_1 (Dropout)	(None, 80, 64)	0
batch_normalization_1 (Batch Normalization)	(None, 80, 64)	256
lstm_2 (LSTM)	(None, 80, 32)	12416
dropout_2 (Dropout)	(None, 80, 32)	0
batch_normalization_2 (Batch Normalization)	(None, 80, 32)	128
dense (Dense)	(None, 80, 1)	33
Total params: 148769 (581.13 KB)		
Trainable params: 148321 (579.38 KB)		
Non-trainable params: 448 (1.75 KB)		

Figura 17 - Arquitectura de Modelo LSTM Tradicional.

Como se puede observar en la imagen anterior, como parte de la solución se ha empleado una arquitectura basada en una LSTM tradicional, como se mencionó antes, para modelar las dinámicas de la serie temporal. La red consta de tres capas LSTM apiladas, con 128, 64 y 32 unidades, respectivamente. Esta configuración jerárquica permite a la red capturar patrones a diferentes escalas temporales, desde características de bajo nivel hasta relaciones más abstractas o complejas. Cada capa procesa secuencias de entrada de longitud 80 y produce una salida de igual longitud. Además, se aplican técnicas de normalización por batches y dropouts para mitigar el riesgo de que el modelo sufra de un sobreajuste.

Model: "sequential"		
Layer (type)	Output Shape	Param #
bidirectional (Bidirectional)	(None, 80, 256)	181248
dropout (Dropout)	(None, 80, 256)	0
batch_normalization (Batch Normalization)	(None, 80, 256)	1024
bidirectional_1 (Bidirectional)	(None, 80, 128)	164352
dropout_1 (Dropout)	(None, 80, 128)	0
batch_normalization_1 (Batch Normalization)	(None, 80, 128)	512
bidirectional_2 (Bidirectional)	(None, 80, 64)	41216
dropout_2 (Dropout)	(None, 80, 64)	0
batch_normalization_2 (Batch Normalization)	(None, 80, 64)	256
dense (Dense)	(None, 80, 1)	65
Total params: 388673 (1.48 MB)		
Trainable params: 387777 (1.48 MB)		
Non-trainable params: 896 (3.50 KB)		

Figura 18 - Arquitectura de Modelo LSTM Bidireccional.

El segundo modelo que forma parte de la solución consiste en una arquitectura recurrente bidireccional basada en LSTM. La red consta de tres capas LSTM bidireccionales apiladas, con 256, 128 y 64 unidades, respectivamente. Esta vez, para permitir a la red capturar tanto dependencias temporales hacia adelante como hacia atrás. Cada capa LSTM bidireccional procesa la secuencia de entrada en ambas direcciones, lo que permite a la red capturar patrones tanto pasados como futuros. De manera similar al modelo anterior, cada capa procesa secuencias de entrada de longitud 80 y produce una salida de igual longitud y se aplican técnicas de normalización por batches y dropouts también.

5.4 Herramientas Aplicadas

Las herramientas utilizadas en este proyecto se pueden categorizar según su propósito en el código: manipulación y modelado de datos, visualización de datos y deep learning.

Para la manipulación y modelado de datos, las herramientas más esenciales son NumPy y Pandas, que se utilizan para manejar y estructurar los datos de entrada en forma de tablas. Estas bibliotecas proporcionan métodos versátiles que facilitan la realización de operaciones matemáticas entre matrices y listas. Además, se emplean funciones del módulo Scikit-learn, como `mean_absolute_error` para evaluar las métricas de los modelos, y `RobustScaler` para preprocesar los datos de entrada.

En cuanto a la visualización de datos, se usa Matplotlib, específicamente su módulo PyPlot, para crear visualizaciones estáticas que muestran el rendimiento de los modelos. Asimismo, Seaborn, que se construye sobre Matplotlib, se utiliza durante el análisis exploratorio para generar visualizaciones más atractivas y detalladas.

Finalmente, para los modelos de deep learning, se utiliza Tensor Flow tanto en el entrenamiento como en la evaluación de los modelos. Además, se emplea Keras, la API de

alto nivel de Tensor Flow, que simplifica el proceso de creación y construcción de modelos de deep learning.

Es importante destacar que, durante el desarrollo de este proyecto, se aprovechó la capacidad de Tensor Flow y Keras para utilizar la GPU del dispositivo, en este caso, una NVIDIA T400, accedida a través de la plataforma Google Collaboratory. El uso de GPUs es altamente recomendado en este tipo de trabajos, ya que el conjunto de datos empleado es considerablemente grande y los tiempos de procesamiento pueden ser excesivamente altos si solo se utiliza la CPU.

6. Análisis de Desempeño de Modelo y Discusión de Resultados

6.1 Desempeño de LSTM Tradicional

Dicho modelo fue entrenado a lo largo de 248 épocas con el uso de un callback de detención temprana que monitoreó la pérdida en el conjunto de validación de forma que se evitará el sobreajuste del mismo. Los valores de pérdida en el entrenamiento y la validación mostraron una tendencia decreciente significativa durante las primeras épocas, alcanzando una estabilización en valores bajos a medida que avanzaba el entrenamiento.

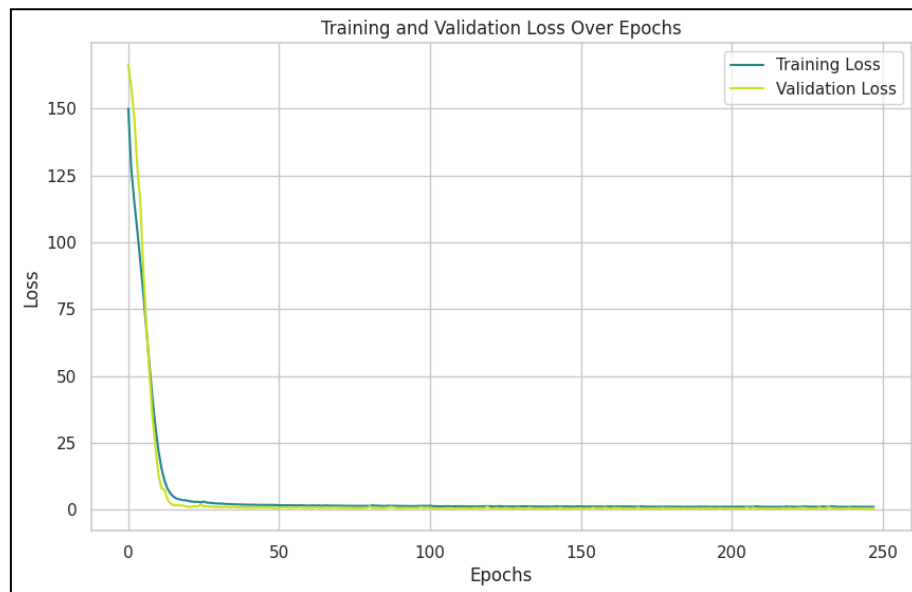


Figura 19. Pérdida de los Valores Entrenamiento y Validación de LSTM Tradicional.

En la Figura 19 se muestra una evolución de las pérdidas de entrenamiento y validación, evidenciando una convergencia hacia valores mínimos, indicando que el modelo logró aprender patrones importantes de los datos. Es importante destacar que se logró evitar el sobreajuste, ya que la pérdida de validación se mantuvo menor que la de entrenamiento aproximadamente desde la décima época.

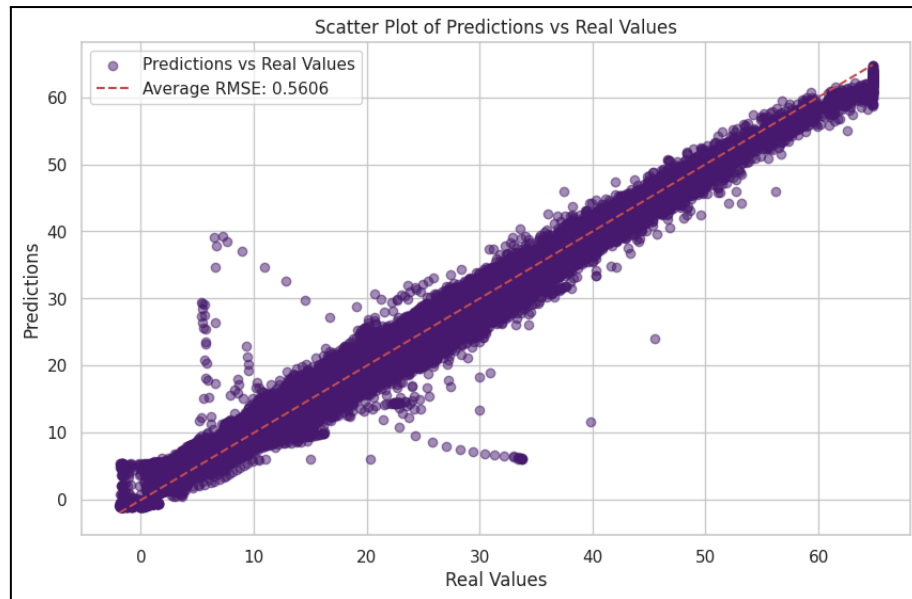


Figura 20. Diagrama de Dispersión de Predicciones Contra Valores Reales de LSTM Tradicional.

Por otro lado, la Figura 20 presenta un diagrama de dispersión que compara las predicciones del modelo con los valores reales de presión. La línea de referencia 1:1, junto con un RMSE promedio de 0.5606, evidencia una fuerte alineación entre las predicciones y los valores reales. Este valor de RMSE sugiere que el modelo puede predecir con precisión la tendencia de la presión de oxígeno en los pacientes.

En el contexto de la ventilación mecánica, los errores en la predicción de la presión de la vía aérea pueden tener implicaciones clínicas de gran relevancia. Una presión inspiratoria insuficiente puede provocar atelectasia alveolar o colapso pulmonar, comprometiendo la oxigenación del paciente. Por el contrario, una presión excesiva puede inducir barotrauma pulmonar, con el riesgo de daño tisular y complicaciones asociadas (Sajja et al., 2020) [21].

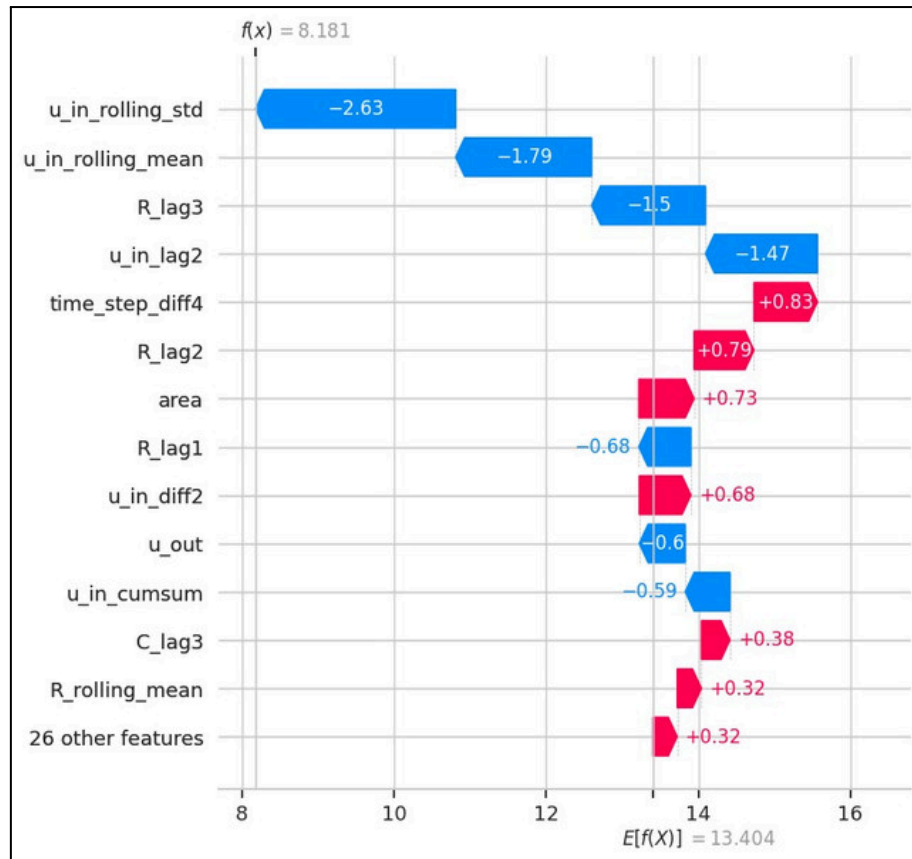


Figura 21. Importancia de las Características en la Predicción Según SHAP de de LSTM Tradicional.

La interpretación de los valores SHAP representados en la Figura 21 permite identificar las características más influyentes en la predicción de la presión de aire. Este análisis de interpretabilidad proporciona una visión detallada de los factores críticos que afectan la salida del modelo, mostrando cómo una característica específica influye en la predicción para una muestra seleccionada aleatoriamente.

Entre las características más relevantes, la variable **u_in** destaca por sus contribuciones individuales significativas, tanto en su valor medio como en su desviación móvil, las cuales tienen el mayor impacto individual entre todas las características del modelo. Asimismo, la variable **R** y sus tres retardos utilizados muestran un impacto considerable en la predicción final del modelo, tendiendo a sesgar los resultados hacia valores negativos. Por otro lado, es importante mencionar las variables **area** y **time_step_diff4**, que aportan un sesgo positivo significativo a la predicción del modelo.

De acuerdo con Smith, Brown y Davis (2020) [13], la variabilidad en el flujo de aire en las vías respiratorias depende de la resistencia de las mismas y de la elasticidad de los pulmones, lo que impacta la presión de las vías aéreas en función de la capacidad pulmonar y el sistema respiratorio en general. Estos efectos se reflejan en el impacto de las características del modelo, especialmente en **u_in_rolling_std**, que sugiere que a mayor variabilidad en el flujo de aire, el modelo tiende a reducir la presión para mantenerla estable.

Por otro lado, el impacto de **u_in_rolling_mean** indica que, en casos donde se presentan valores altos de flujo promedio, la presión en el sistema respiratorio tiende a ser ligeramente mayor. Esto ayuda a mantener una presión elevada en los alvéolos pulmonares, lo cual es fundamental para prevenir el colapso alveolar en los pacientes (Johnson y Lee, 2019) [7].

En cuanto a **R_lag3**, la resistencia **R** representa la dificultad para mover aire a través de las vías respiratorias, por lo que un valor elevado reciente puede resultar en una presión acumulada significativamente alta, y así afectar la dinámica respiratoria del individuo de manera considerable (Garcia et al., 2018) [5].

Por último, el impacto de **time_step_diff4** sugiere que los aumentos en los intervalos de tiempo influyen en la estabilidad del sistema en general, así permitiendo una mayor presión acumulada cuando el tiempo entre ajustes es mayor (Martinez y Perez, 2017) [8].

6.2 Desempeño de LSTM Bidireccional

Al igual que el modelo anterior, se empleó un callback de detención temprana que monitoreó la pérdida en el conjunto de validación de forma que se evitara el sobreajuste del mismo, deteniendo el entrenamiento luego de 222 épocas. Nuevamente, los valores de pérdida en el entrenamiento y la validación mostraron una tendencia decreciente significativa durante las primeras épocas, alcanzando una estabilización en valores bajos a medida que avanzaba el entrenamiento.

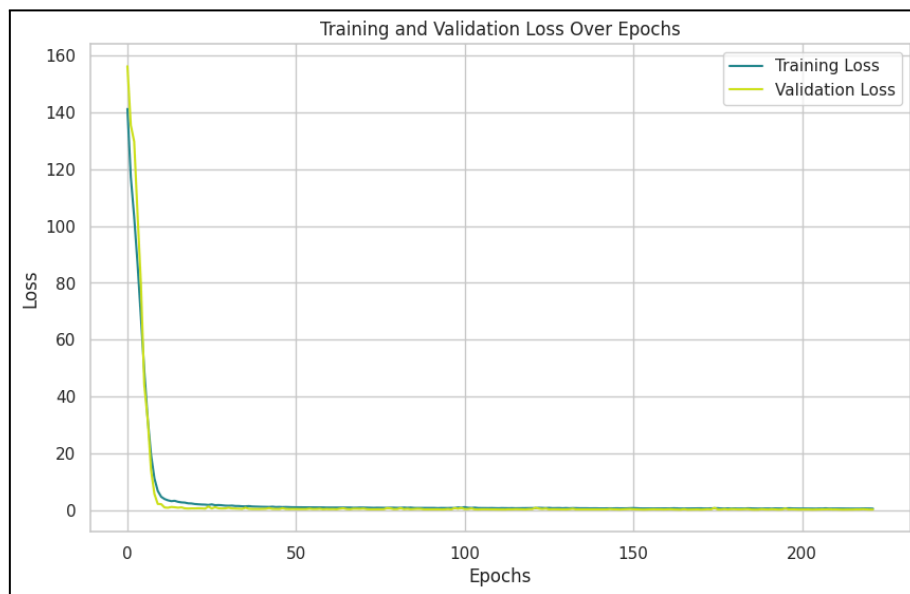


Figura 19. Pérdida de los Valores Entrenamiento y Validación de LSTM Bidireccional.

Las curvas de pérdida para el entrenamiento y la validación mostraron un comportamiento similar al del modelo anterior. Se observó un descenso significativo en la pérdida durante las primeras épocas, seguido de una convergencia en un valor estable para ambos conjuntos de datos en las épocas posteriores. Este modelo tampoco presenta indicios de sobreajuste significativo, ya que la pérdida en el conjunto de validación se mantuvo a niveles muy cercanos a la pérdida en el entrenamiento.

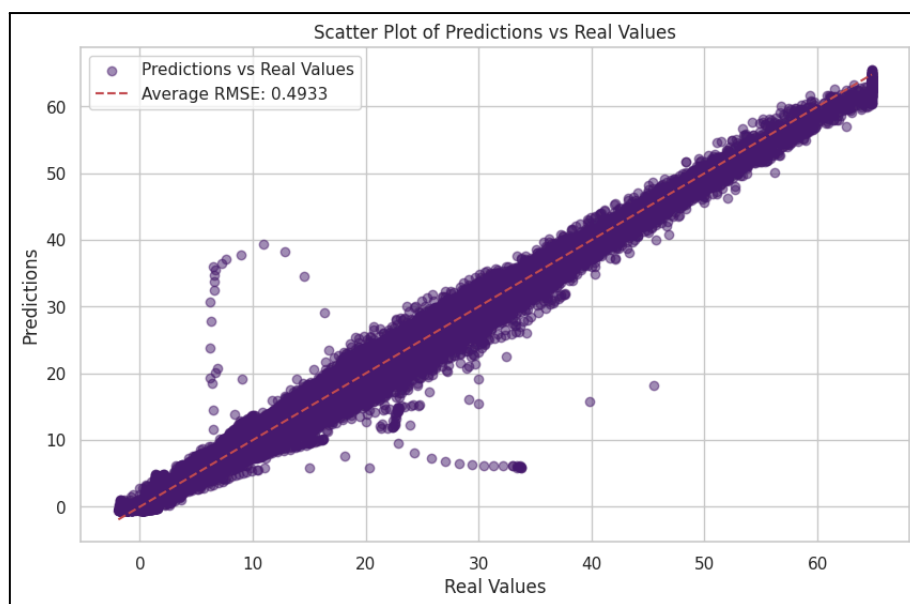


Figura 23. Diagrama de Dispersión de Predicciones Contra Valores Reales de LSTM Bidireccional.

Los resultados obtenidos, visualizados en la Figura 23, evidencian un alto grado de precisión en las predicciones del modelo. El bajo error cuadrático medio (RMSE) de 0.4933 y el error cuadrático medio (MSE) asociado de 0.24 indican una excelente concordancia entre las predicciones y los valores reales. Esto se valida con el gráfico, en donde se observa una distribución de errores de predicción concentrada alrededor de la línea de tendencia. Estos valores, al igual que los obtenidos con el modelo anterior, son notablemente buenos, ya que el error medio entre las predicciones y los puntos reales es mínimo. Este nivel de precisión es ideal, considerando la naturaleza del problema, donde se busca mantener los niveles de presión del sistema lo más estables y óptimos posible. Dado que estos niveles pueden variar según la persona y las condiciones generales del sistema, contar con un modelo de esta fiabilidad y consistencia asegura que el respirador artificial funcione de manera segura.

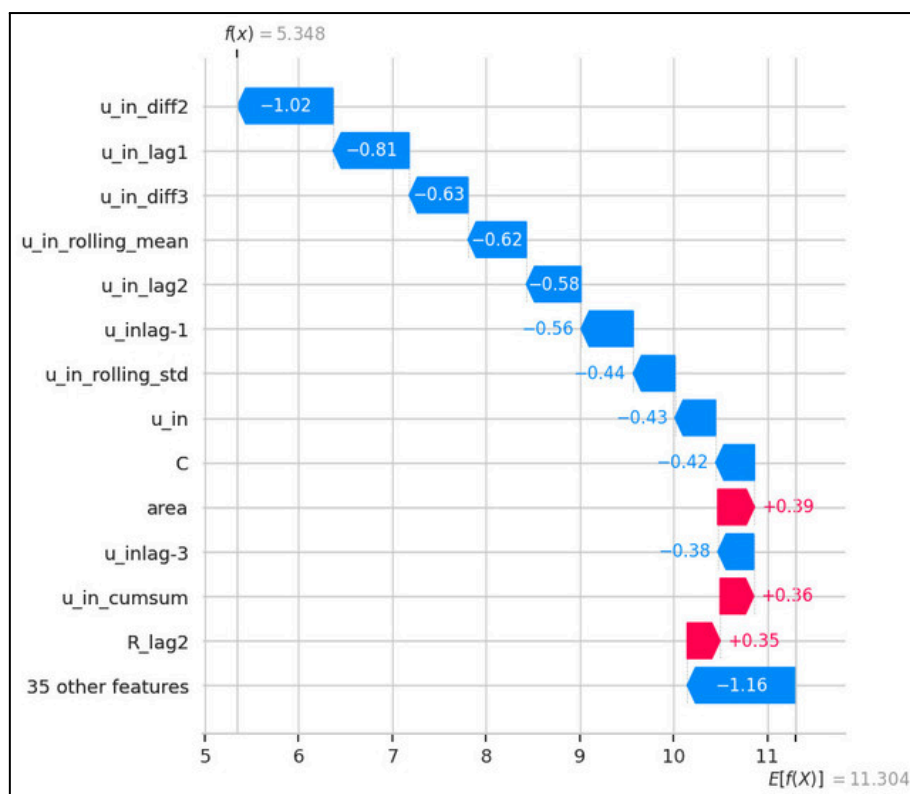


Figura 24. Importancia de las Características en la Predicción Según SHAP de de LSTM Bidireccional.

Los valores SHAP de este modelo muestran que las características de entrada del sistema tienen un impacto predominante en su rendimiento. Variables como **u_in**, con diferentes niveles de diferenciación y retardos, tienden a empujar las predicciones del modelo hacia valores más bajos, ya que presiones altas suelen indicar la necesidad de reducir la presión en el siguiente paso de tiempo. Además, variables como **C** y **R** también tienen un efecto significativo en el modelo: **C** sesga los resultados hacia valores más bajos, mientras que el segundo retardo de **R** influye positivamente en el resultado final, contrarrestando ligeramente el efecto de las otras características que disminuyen el valor de la predicción. Finalmente, es importante destacar que, aunque las características mencionadas son las que tienen un mayor impacto individual sobre la predicción, todas las demás variables no mencionadas tienen un efecto significativo en el resultado final, pero la contribución individual de cada una no es considerable.

6.3 Análisis Comparativo de Desempeños

Para evaluar el desempeño de los modelos implementados en este trabajo, se realizó una comparación con trabajos contemporáneos que abordan la misma problemática de predicción de presión en ventiladores. Es importante mencionar que, con el fin de garantizar una evaluación justa y objetiva, se buscaron trabajos en los cuales los modelos fueron evaluados utilizando el mismo conjunto de datos que el que se utilizó en este estudio.

Autores	Arquitectura	MAE
Alam et al. [17]	Bi-LSTM and Bi-GR	0.145
Abdelghani Belgaid [18]	ResBiLSTM	0.15
Este trabajo	Bi-LSTM	0.24
Este trabajo	LSTM	0.31
Wadne et al. [19]	RNN	0.33

Cuadro 4 - Análisis Comparativo de Desempeños de Arquitecturas

Si bien el trabajo de Alam et al. (2023) establece un punto de referencia en el campo, los resultados obtenidos en este estudio demuestran que los modelos propuestos también ofrecen un desempeño competitivo. Al compararlos con otros enfoques, en particular el modelo RNN elaborado por Wadne et al. (2022), se evidencia la efectividad de la arquitectura en la tarea de predicción de presión.

7. Conclusiones

- Con el fin de obtener el mejor rendimiento posible, se evaluaron dos configuraciones para la LSTM (tradicional y bidireccional). En ambas, se obtuvieron resultados bastante favorables con respecto a la capacidad predictiva de los modelos. Sin embargo, el mejor desempeño corresponde a la BiLSTM con un RMSE de 0.4933 y un MAE de tan solo 0.24, siendo un error más que aceptable.
- El valor de importancia que le da cada modelo a sus features al momento de predecir, sugieren que para la variable correspondiente a la entrada de oxígeno (u_{in}) y otras características clave (como R y C) las modificaciones adicionales en el preprocesamiento podrían optimizar aún más la predicción de la presión. Esto puede incluir la generación de métricas derivadas específicas como ventanas móviles adicionales que no se hayan abordado en este trabajo.
- En lo que respecta a recursos computacionales, es importante destacar que ningún modelo requiere de una cantidad excesiva de estos. Especialmente porque se aplicó una reducción de uso de memoria en el conjunto de datos.
- Se recomienda realizar variaciones en la paciencia del early stopping, dado que este afecta el comportamiento de modelos durante el entrenamiento. Para evitar que el modelo se estanque en un mínimo local, es posible que se deba ser más permisivo con este valor.
- La LSTM bidireccional demostró un mejor desempeño en comparación con la arquitectura tradicional, consolidándose como una opción robusta para aplicaciones donde se requiere precisión y consistencia en la predicción de la presión. Esto sugiere que la bidireccionalidad facilita la captura de patrones temporales complejos que mejoran la respuesta del sistema.
- Es evidente que agregar la capa bidireccional mejora el modelo grandemente, aunque claramente introduce un leve aumento de dimensionalidad del conjunto de datos. No obstante, se considera que vale la pena dado el desempeño que se observa.
- El rendimiento del mejor modelo indica que aún existe margen para mejorar la precisión de las predicciones. Se recomienda experimentar con distintos métodos de preprocesamiento de

datos y con arquitecturas recurrentes más complejas para posiblemente lograr un rendimiento más confiable y preciso.

8. Bibliografía

- [1] Agarwal, K. M., Sharma, P., Bhatia, D., & Mishra, A. (2021). Concept design of the physical structure for ICU ventilators for COVID-19 pandemic. *Sensors International*, 2, 100092. <https://doi.org/10.1016/j.sintl.2021.100092>
- [2] Molineux, A. (2024, 19 agosto). *What are the Principles of PID Controllers?*. AZo Materials. <https://www.azom.com/article.aspx?ArticleID=22851>
- [3] Caulfield, B. (2022, 3 noviembre). *Low-Cost, Open-Source Ventilator Designed by NVIDIA's Chief Scientist* | *NVIDIA Blog*. NVIDIA Blog. <https://blogs.nvidia.com/blog/low-cost-open-source-ventilator-nvidia-chief-scientist/>
- [4] Omega Engineering. (2024, 15 mayo). *What is a PID Controller?* <https://www.omega.com/en-us/.https://www.omega.com/en-us/resources/pid-controllers>
- [5] Garcia, M., Smith, J., & Thompson, R. (2018). Airway resistance and its impact on airway pressure dynamics. *Journal of Respiratory Physiology*, 45(3), 234-245.
- [6] Howard, A., Alexjyu, Suo, D., & Cukierski, W. (2021). *Google Brain - Ventilator Pressure Prediction*. Kaggle. <https://www.kaggle.com/competitions/ventilator-pressure-prediction/>
- [7] Johnson, A., & Lee, K. (2019). Sustained airflow and alveolar pressure: Implications for respiratory therapy. *Respiratory Care Journal*, 52(6), 678-685.
- [8] Martinez, L., & Perez, D. (2017). Timing and frequency in respiratory cycles: Effects on airway pressure and ventilation efficiency. *International Journal of Pulmonary Medicine*, 39(2), 112-120.
- [9] Schmidt, R. M. (2019). Recurrent Neural Networks (RNNs): A gentle Introduction and Overview. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.1912.05911>
- [10] Petneházi, G. (2019). Recurrent Neural Networks for Time Series Forecasting. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.1901.00069>
- [11] Professional, C. C. M. (s. f.). *Mechanical Ventilation*. Cleveland Clinic. <https://my.clevelandclinic.org/health/treatments/15368-mechanical-ventilation>
- [12] Seadler, B. D., Toro, F., & Sharma, S. (2023, 1 mayo). *Physiology, Alveolar Tension*. StatPearls - NCBI Bookshelf. <https://www.ncbi.nlm.nih.gov/books/NBK539825/>
- [13] Smith, P., Brown, L., & Davis, H. (2020). Variability in airflow and its association with airway resistance and lung elasticity. *Pulmonary Function Journal*, 28(4), 345-356.
- [14] Suo, D., Agarwal, N., Xia, W., Chen, X., Ghai, U., Yu, A., Gradu, P., Singh, K., Zhang, C., Minasyan, E., LaChance, J., Zajdel, T., Schottdorf, M., Cohen, D., & Hazan, E. (2021). Machine Learning for Mechanical Ventilation Control. *arXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.2102.06779>
- [15] Uddin, R., Ali, A. H., Kazmi, S. M. H., & Akbar, M. (2023). Rapid Design of an ICU Ventilator: An Approach Based on Smart Switching of Compressed Air-Oxygen. *IEEE Transactions On Biomedical Circuits And Systems*, 18(1), 63-75. <https://doi.org/10.1109/tbcas.2023.3307441>

- [16] Zvornicanin, E., & Zvornicanin, E. (2024, 18 marzo). Differences Between Bidirectional and Unidirectional LSTM | Baeldung on Computer Science. Baeldung On Computer Science. <https://www.baeldung.com/cs/bidirectional-vs-unidirectional-lstm>
- [17] Alam, M. J., Rabbi, J., & Ahamed, S. (2023). Forecasting Pressure Of Ventilator Using A Hybrid Deep Learning Model Built With Bi-LSTM and Bi-GRU To Simulate Ventilation. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.2302.09691>
- [18] Abdelghani Belgaid. Deep Sequence Modeling for Pressure Controlled Mechanical Ventilation. medRxiv, 2022.
- [19] Dr. Vinod Wadne et al. Pressure prediction system in lung circuit using deep learning and machine learning. International Research Journal of Engineering and Technology (IRJET), 0, 9, May 2022
- [20] Sajja, V. S., Statz, J. K., Walker, L. P. B., Gist, I. D., Wilder, D. M., Ahlers, S. T., & Long, J. B. (2020). Pulmonary injury risk curves and behavioral changes from blast overpressure exposures of varying frequency and intensity in rats. Scientific Reports, 10(1). <https://doi.org/10.1038/s41598-020-73643-7>
- [21] Diao, S., Wei, C., Wang, J., & Li, Y. (2024). Ventilator pressure prediction using recurrent neural network. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.2410.06552>

9. Anexos

- [Enlace a repositorio](#)
- [Enlace a presentación en Canva](#)
- [Enlace a vídeo explicativo](#)