



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA
CARRERA DE COMPUTACIÓN

**DESARROLLO E IMPLEMENTACIÓN DE UN ALGORITMO DE PREDICCIÓN DE
CONSUMO DE AGUA POTABLE Y VISUALIZACIÓN DE LOS DATOS DE
CONSUMO POR USUARIO Y SECTORES MEDIANTE REDES NEURONALES
RECURRENTES DENTRO DEL PROYECTO CEDÍA-TARPUQ**

Trabajo de titulación previo a la obtención del
Título de Ingeniero en Ciencias de la
Computación

AUTORES:

BRYAN ISRAEL CHUCHUCA ALVARRACIN

JONNATHAN DANIEL SICHA RODRIGUEZ

TUTOR: ING. ERWIN JAIRO SACOTO CABRERA, PhD

Cuenca-Ecuador

2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Bryan Israel Chuchuca Alvarracin con documento de identificación N° 0106330145 y Jonnathan Daniel Sicha Rodriguez con documento de identificación N° 0106232812 manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 8 de marzo del año 2022

Atentamente,

Bryan Israel Chuchuca Alvarracin

0106330145

Jonnathan Daniel Sicha Rodriguez

0106232812

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Bryan Israel Chuchuca Alvarracin con documento de identificación No. 0106330145 y Jonnathan Daniel Sicha Rodriguez con documento de identificación No. 0106232812, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto técnico: “Desarrollo e implementación de un algoritmo de predicción de consumo de agua potable y visualización de los datos de consumo por usuario y sectores mediante redes neuronales recurrentes dentro del proyecto CEDÍA-TARPUQ”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 8 de marzo del año 2022

Atentamente,

Bryan Israel Chuchuca Alvarracin
0106330145

Jonnathan Daniel Sicha Rodriguez
0106232812

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Erwin Jairo Sacoto Cabrera con documento de identificación N° 0301185229, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DESARROLLO E IMPLEMENTACIÓN DE UN ALGORITMO DE PREDICCIÓN DE CONSUMO DE AGUA POTABLE Y VISUALIZACIÓN DE LOS DATOS DE CONSUMO POR USUARIO Y SECTORES MEDIANTE REDES NEURONALES RECURRENTE DENTRO DEL PROYECTO CEDÍA-TARPUQ, realizado por Bryan Israel Chuchuca Alvarracin con documento de identificación N° 0106330145 y por Jonnathan Daniel Sicha Rodriguez con documento de identificación N° 0106232812, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 8 de marzo del año 2022

Atentamente,

Ing. Erwin Jairo Sacoto Cabrera, PhD

0301185229

Dedicatorias

A mis padres José y María, a mis hermanos/as por todo su apoyo y amor incondicional, por su trabajo, su esfuerzo y el sacrificio que han realizado para permitirme llegar a este momento tan especial en mi vida, ese sacrificio que valore día tras día, gracias por inculcar en mí, la valentía y el coraje para poder superar cualquier dificultad y hacer posible que pueda culminar mi carrera profesional, gracias por ser mi ejemplo y hacerme sentir afortunado de tener una familia tan especial y ejemplar, sin su apoyo no hubiese sido posible llegar a cumplir este gran sueño.

A Caleb que eres la razón de que todos los días me levante a esforzarme por el mañana y el presente, como en todos mis logros en este has estado presente.

Bryan Israel Chuchuca Alvarracin

A mis padres Julio y Angeles ,por haberme forjado como la persona que soy en la actualidad y haber dedicado todo sus esfuerzos,trabajo y apoyo incondicional para lograr muchos de mis logros en los cuales se incluye esta estancia de mis estudios, que para mi es muy grato tenerles en mi vida y compartir este momento tan importante.

A mis hermanos/as por el apoyo moral y sicológico que me brindaron día a día en el transcurso de mi carrera.

A mis amigos por toda la ayuda y el apoyo durante este período tan importante de mi educación académica.

Jonnathan Daniel Sicha Rodriguez

Agradecimientos

En primer lugar agradezco a Dios por darme la fortaleza y salud para culminar con el presente trabajo de titulación. A mi familia, mis padres José y María, mis hermanos/as quienes siempre me brindaron su apoyo incondicional en todas mis metas y proyectos que me planteado.

A mis profesores por la dedicación y apoyo que me han brindado durante mi formación académica, por el respeto a mis sugerencias e ideas, por la dirección y el rigor que me servirá para toda mi vida profesional. A mis amigos por confiar en mí, por su apoyo, por ser parte de este proceso, por los buenos y malos momentos que hemos compartido fuera del ambiente universitario.

Bryan Israel Chuchuca Alvarracin

En primera instancia agradezco a Dios por darme la fortaleza, sabiduría y salud para culminar este trabajo, a mi familia quienes estuvieron para mí siempre que lo necesitaba y me han apoyado en cada una de las decisiones que he tomado en mi vida. También agradezco a mis formadores, personas que me ayudaron a llegar hasta este punto de mi vida ya que con sus conocimientos compartidos he logrado importantes objetivos como es el culminar el desarrollo de mi tesis y obtener una gratificante titulación profesional.

Jonnathan Daniel Sicha Rodriguez

Contenido

Glosario de términos	1
Resumen	2
Abstract	3
1 Introducción	4
1.1 Descripción del problema	5
1.1.1 Antecedentes	5
1.1.2 Importancia y alcances	5
1.1.3 Delimitación	6
1.2 Objetivos generales y específicos	6
1.2.1 Objetivo General	6
1.2.2 Objetivos específicos	6
2 Marco de referencia teórico	7
2.1 Internet of things (IoT)	7
2.1.1 Modelos de comunicación	7
2.1.2 Seguridad	9
2.1.3 Privacidad	9
2.1.4 Interoperabilidad	9
2.1.5 Entornos para aplicaciones IoT	10
2.2 Sistemas inteligentes	11
2.3 Inteligencia Artificial	11
2.3.1 Técnicas utilizadas en inteligencia artificial	12
2.4 Machine Learning	12
2.4.1 Tipos de aprendizaje	12
2.5 Aplicaciones de Machine Learning	14
2.6 Redes Neuronales	14
2.6.1 Red Neuronal Recurrente (RNN)	15
2.7 Arquitecturas LSTM	19
2.7.1 Modelo LSTM univariado	19
2.7.2 Modelos LSTM multivariados	20
2.7.3 Validación del modelo (RNN)	22

2.7.4	Técnicas de validaciones en redes neuronales	22
2.8	Series de tiempo en arquitecturas LSTM	23
2.8.1	Componentes de una serie de tiempo	24
2.8.2	Tendencia de una serie de tiempo	25
2.8.3	Medidas de error de pronóstico series de tiempo	25
2.9	Preprocesamiento de datos en series de tiempo y arquitecturas LSTM	26
2.9.1	Selección de datos	26
2.9.2	Preprocesado de datos	26
2.9.3	Selección de Características	27
2.10	Técnicas de visualización para la predicción de una serie temporal mediante una arquitectura LSTM	27
2.11	Consumo de agua potable	28
2.11.1	Gestión de demanda	29
3	Solución Propuesta	30
3.0.1	Primera fase	30
3.0.2	Segunda fase	30
3.0.3	Limpieza de valores ausentes	31
3.0.4	Tercera fase	32
3.0.5	Cuarta fase	32
3.0.6	Quinta fase	32
3.0.7	Sexta fase	33
4	Desarrollo de la solución propuesta	34
4.1	Arquitecturas implementadas	34
4.1.1	Codificador-decodificador LSTM	34
4.1.2	Codificador-decodificador donde la parte del decodificador consta de una capa bidireccional	35
4.1.3	Salida vectorial para múltiples pasos	36
4.1.4	CNN-LSTM de Múltiples cabezales para varias salidas	37
5	Resultados	40
5.1	Codificador-decodificador LSTM	40
5.2	Codificador-decodificador donde la parte del decodificador consta de una capa Bidireccional	41
5.3	Salida Vectorial para múltiples pasos	42
5.4	CNN-LSTM de Múltiples cabezales para varias salidas	43
5.5	Selección del mejor modelo	44
5.6	Visualización de los datos	45

6 Cronograma de actividades y presupuesto	48
6.1 Cronograma de Actividades	48
6.2 Presupuesto	52
7 Conclusiones y Recomendaciones	53
7.1 Conclusiones	53
7.2 Recomendaciones	54
Bibliografía	55

Lista de Figuras

2-1	Red neuronal. Figura tomada [1]	15
2-2	Red neuronal recurrente. Figura tomada [2]	16
2-3	Estructura modelo LSTM. Figura tomada [3]	18
2-4	Arquitectura encoder-decoder LSTM. Figura tomada [4]	21
2-5	Estructuras Modelos LSTM múltiples entradas salidas de varios pasos . Figura tomada [5]	22
2-6	Ejemplos técnicas de visualización. Figura tomada [6]	28
2-7	Indicadores de consumo de agua Ecuador. Figura tomada [7]	29
3-1	Diagrama de la solución propuesta	30
4-1	Arquitectura encoder-decoder	35
4-2	Arquitectura encoder-decoder con una capa Bidireccional en el decoder	36
4-3	Arquitectura salida Vectorial para múltiples pasos	37
4-4	Arquitectura CNN-LSTM de Múltiples cabezales para varias salidas	38
5-1	Grafica comparativa de error MAE vs RMSE arquitectura 1	41
5-2	Gráfica comparativa de error MAE vs RMSE arquitectura 2	42
5-3	Grafica comparativa de error MAE vs RMSE arquitectura 3	43
5-4	Gráfica comparativa de error MAE vs RMSE arquitectura 4	44
5-5	Función de perdida (loss) mejor modelo	45
5-6	Evaluación del mejor modelo real vs predicción	45
5-7	Consumo de agua potable diferentes usuarios	46
5-8	Consumo de agua de un usuario con predicción a 3 meses con aumento y reducción del 10 %	46
5-9	Cantidad y monetización del consumo de agua de un usuario con predicción a 3 meses con aumento y reducción del 10 %	47

Lista de Tablas

4-1	Hiperparametros definidos para los diferentes modelos implementados	39
5-1	Hiperparametros probados en la arquitectura 1	40
5-2	Hiperparametros probados en la arquitectura 2	41
5-3	Hiperparametros probados en la arquitectura 3	42
5-4	Hiperparametros probados en la arquitectura 4	43
5-5	Hiperparametros mejor modelo	45
6-1	Representa las actividades del Objetivo Especifico 1	48
6-2	Representa las actividades del Objetivo Especifico 2	49
6-3	Representa las actividades del Objetivo Especifico 3	49
6-4	Representa las actividades del Objetivo Especifico 4	50
6-5	Actividades desarrolladas para el proyecto	51
6-6	Presupuesto para el proyecto	52

Glosario de términos

CEDIA: Corporación Ecuatoriana para el desarrollo de la Investigación y la Academia.

RNN: Recurrent Neural Network o Red Neuronal Recurrente.

Epoch: Paso completo del conjunto de datos de entrenamiento a través del algoritmo.

IoT: Internet de las cosas.

IA: Inteligencia Artificial.

MAE: Mean Absolute Error o Error Absoluto Medio.

MSE: Mean Squared Error o Error Cuadrático Medio.

RMSE: Root Mean Square Error o Raíz del Error Cuadrático Medio.

LSTM: Long short-term memory o memoria a largo plazo.

CNN: Red neuronal convolucional.

Resumen

En Ecuador el consumo de agua potable continuamente experimenta un crecimiento constante a causa de la población y otros factores como: las precipitaciones, temperatura y enfermedades, por lo cual existe la necesidad de reducir y optimizar el consumo de agua potable dentro de los diferentes sectores. En este trabajo se plantea pronosticar el consumo de agua potable con alta precisión, empleando técnicas de machine learning con un modelo de red neuronal recurrente de tipo long short-term memory con el objetivo de analizar el comportamiento de consumo de agua potable, para este fin se utiliza un dataset genérico con muestras de cuarenta años, en los cuatro modelos propuestos se aplican variaciones en sus hiperparámetros, estos modelos contienen múltiples salidas representando a los usuarios y realiza una proyección a múltiples fechas, por último se realiza la selección del mejor modelo haciendo uso de las métricas mean absolute error y mean square error obteniendo un modelo de aprendizaje optimizado que se ajusta al pronóstico de consumo de agua potable por usuario y sector.

Palabras clave: consumo de agua potable, LSTM, machine learning, series de tiempo.

Abstract

In Ecuador, drinking water consumption is constantly growing due to population and other factors such as rainfall, temperature and diseases, so there is a need to reduce and optimize the consumption of drinking water in different sectors. In this work we propose to forecast drinking water consumption with high accuracy, using machine learning techniques with a recurrent neural network model of long short-term memory type in order to analyze the behavior of drinking water consumption, for this purpose a generic dataset with samples of forty years is used, in the four proposed models variations in their hyperparameters are applied, These models contain multiple outputs representing the users and perform a projection to multiple dates. Finally, the best model is selected using the mean absolute error and mean square error metrics to obtain an optimized learning model that adjusts to the drinking water consumption forecast by user and sector.

key words: drinking water consumption, LSTM, machine learning, time series.

1 Introducción

El agua es un recurso natural vital para los seres humanos, animales y plantas que a través de los años se ha venido disminuyendo por destrucción de fuentes naturales y la minería, tal como lo detallan los autores en [8, 9]. En consecuencia cada vez se observa más ríos en sequía así como también la disminución de su calidad, en [10] nos conlleva a pensar sobre el futuro de las próximas generaciones si no se planifica y no se toma las decisiones acertadas acerca del consumo responsable del agua en este caso de uso doméstico tratando de mejorar el panorama del futuro de este recurso natural.

En [11], se indica que el planeta tierra tiene el 70% de la superficie total cubierta por agua, alrededor del 97% del agua es agua salada, y esta se encuentra distribuida en los mares. Escasamente el 3% del total del agua es dulce, y esta se encuentra distribuida un poco más de la tercera parte en los glaciares congelados y los casquetes de los polos y en las altas montañas del planeta. Según [12], las regiones con gran demanda de agua pueden hacer frente a los escasos de agua hasta la actual fecha, pero para el 2025 se provee que muchas regiones de la población pueden llegar a vivir en casos de agua ya sea esta moderada o severa. El agua potable de uso doméstico, en [13], se indica que es complejo comprender todo el preprocesamiento y costo que conlleva para su potabilización, para que esta sea libre de gérmenes o patógenos, no obstante, la mayoría de la población no tiene aún agua potable en sus viviendas, por lo que deben buscar en sitios lejanos que no reúnen las condiciones de limpieza requerida y se convierte en agua con gérmenes o patógenos que pueden producir ciertas enfermedades.

Frente a la carencia de control del consumo óptimo de agua en todos los usuarios, según [14], se producen altos índices de captación de agua, dentro de esta situación se encuentra la ciudad de Cuenca, Ecuador, la cual requiere aproximadamente 120.000 m³ de agua para afrontar la creciente demanda para una población de 524.563 habitantes. Dado esto en [14], los autores afirman que en 2050 la capacidad de sus habitantes incrementará a 901.499, por lo que será necesario captar 4610 l/s de agua más, si se mantiene el mismo ritmo de consumo y demanda de agua potable presente, se dará un aumento de captación a un 71,85 por ciento de la presente captación.

Por lo tanto el diseño de un modelo basado en inteligencia artificial se encarga de predecir el consumo con una reducción de agua o un aumento, lo cual nos permite informar a los consumidores de agua potable registrados de manera anticipada sobre el historial de su

consumo, la predicción del consumo relacionado con los consumos anteriores y por último los valores a pagar si se continúa consumiendo de la misma manera en conjunto si se disminuyera su consumo en un porcentaje establecido, en efecto los usuarios mediante la visualización de los resultados de la red neuronal recurrente replantarían su conducta acerca del consumo del agua, además del aporte que estarían haciendo al cuidado del medio ambiente y a los recursos naturales.

1.1. Descripción del problema

1.1.1. Antecedentes

De acuerdo con estudios realizados sobre el consumo de agua focalizados a diferentes usuarios, los resultados señalan que:

- “Un problema que se presenta en la actualidad es el aumento de la población mundial, llegando a ser 6000 millones hasta el año 2000 y se incrementará hasta 9000 millones en el 2050. En Ecuador, la población actual es de 17.511.000 y se tiene previsto que la población crecerá hasta un máximo de aproximadamente 22 millones en el año 2060” [15]. La opinión de [16] sugiere que 20 *L/persona/día* suele ser suficiente para beber, cocinar, la higiene de los alimentos, lavarse las manos y la cara, pero no para otras prácticas de higiene.
- Según [17], en la mayoría de los sectores ocurre que a menor temperatura menor consumo y viceversa, esto debido a que cuando la temperatura es elevada el clima es seco y la humedad relativa baja por lo que existe mayor consumo doméstico en riego de jardines y cuando la temperatura es baja es probable que existan más precipitaciones por lo que no se necesita consumir agua de riego de jardines ni en otros usos por lo que los usuarios residenciales disminuyen notablemente el consumo.
- “La norma de edificación ecuatoriana 2011 especifica que el equipamiento de una vivienda debe estar entre 200 y 350 *litros/habitante/día*, lo que refleja un rango muy amplio que impide la existencia de un control de consumo-[18], con el objetivo de garantizar la continuidad del servicio de distribución de agua potable en la Ciudad de Cuenca, Ecuador, teniendo en cuenta el crecimiento poblacional, en [14], explica que se debe extraer 610 *l/s* de agua si se mantiene el consumo y la demanda de agua potable vigente, el aumento corresponde a 71,85 % a los depósitos actuales.

1.1.2. Importancia y alcances

En los antecedentes mencionados se evidencia el panorama general del crecimiento poblacional y su relación con el consumo de agua planteándonos un escenario de consumo exponencial

en la población para los próximos años, por lo cual se pretende llegar a la conciencia de los usuarios para lograr reducir el consumo de agua potable.

El desarrollo de una red neuronal recurrente tiene como objetivo cumplir con este propósito mediante la predicción del consumo aproximado para posteriores fechas procurando llegar a la conciencia del cliente a consumir un porcentaje menor a cambio de reducir el valor monetario a pagar por el servicio de distribución y consumo de agua potable.

Además, en [19] menciona los excelentes resultados que brindan las redes neuronales recurrentes de tipo LSTM para la proyección de series de tiempo y en el reconocimiento de patrones los cuales proporcionarán datos estadísticos que apoyarán a los gerentes responsables de la distribución del agua potable, los cuales servirán para la toma de decisiones y planificación anticipada necesaria para prestar un servicio eficiente en la distribución de agua potable.

1.1.3. Delimitación

En el campo de la informática en el área de Machine Learning se pretende desarrollar una red neuronal recurrente para la predicción del consumo de agua potable por usuario y sector usando series de tiempo con datos genéricos durante el periodo académico septiembre 2021 - enero 2022.

1.2. Objetivos generales y específicos

1.2.1. Objetivo General

Desarrollar e implementar un algoritmo de predicción de consumo de agua potable y visualización de los datos de consumo por usuario y sectores mediante Redes neuronales recurrentes dentro del proyecto CEDIA-TARPUQ.

1.2.2. Objetivos específicos

- **OE1.** Estudiar los fundamentos de IoT, series temporales, redes neuronales recurrentes, y técnicas de visualización.
- **OE2.** Diseñar un módulo inteligente para predicción y visualización de consumo de agua potable por usuario.
- **OE3.** Diseñar un módulo inteligente para predicción y visualización de consumo de agua potable por sector.
- **OE4.** Diseñar las pruebas de validación de los algoritmos y del módulo de monitoreo.

2 Marco de referencia teórico

En el presente capítulo se plantean los diversos conocimientos generales y las metodologías de aprendizaje automático.

2.1. Internet of things (IoT)

El concepto general del internet de las cosas se fundamenta en la interconexión de dispositivos masivos con la integración de algún tipo de inteligencia con el objetivo de capturar y compartir información que aporte conocimiento para la toma de decisiones [20], la información puede ser analizada y aplicada en diversas áreas ya que IoT mantiene la posibilidad de personalizar los servicios según las necesidades de las organizaciones [21], no obstante, las diversas implementaciones de los dispositivos de IoT están sujetos a problemas específicos lo cual requiere de una solución puntual [22]. Las aplicaciones de IoT generan datos masivos los cuales pueden ser analizados mediante algoritmos de aprendizaje automático teniendo como fuente sensores interconectados en la nube [23].

2.1.1. Modelos de comunicación

Según [24], cómo se manejan y transmiten estos objetos inteligentes en términos de comunicación. El Internet Architecture Committee (IAB) ha publicado un documento que sirve como guía de networking para objetos inteligentes (RFC 7452) y describe todos los dispositivos relacionados con IoT y hasta los cuatro modelos de comunicación utilizados en Internet. Así, las principales características de cada modelo son:

Dispositivo a dispositivo

El modelo de comunicación de dispositivo a dispositivo muestra que dos o más dispositivos pueden conectarse y comunicarse entre ellos sin un servidor de aplicaciones intermedio [24]. Estos dispositivos pueden comunicarse a través de diferentes tipos de redes, que incluyen: Red IP e internet. Sin embargo, los protocolos como Bluetooth, ZWave y ZigBee se usan comúnmente para la comunicación directa entre dispositivos [21, 25].

La comunicación y mensajería de dispositivo a dispositivo en IoT requiere un protocolo de comunicación para que exista esta funcionalidad. En su mayor parte, este tipo de comunicación se usa actualmente en aplicaciones que admiten la automatización del hogar a través

de paquetes pequeños o programación de datos que operan a tasas de transferencia subjetivamente bajas [20, 25]. Algunos de los usos de los dispositivos IoT son cambiar, calentar, apagar y transmitir pequeñas cantidades de información.

El problema de este método de comunicación con los usuarios es que existen protocolos de comunicación entre dispositivos que obligan a seleccionar dispositivos de la misma familia o utilizar el mismo protocolo para poder trabajar [24, 26].

Dispositivo a la nube

Los dispositivos IoT pueden conectarse directamente a servidores que se encuentran alojados en la nube, como:

Proveedores de servicios de diferentes aplicaciones que pueden intercambiar la data de diferentes orígenes y monitorear el tráfico de mensajes [20]. Este proceso nos permite crear un tipo de conexión entre un dispositivo y una red IP, utilizando un tipo de conexión que ya conocemos (conexión WiFi tradicional por cable o Ethernet), y luego conectarnos a los servicios provistos [24, 27].

Este tipo de conexión es utilizado por marcas conocidas y dispositivos IoT, incluidos los termostatos de aprendizaje de Nest Labs y los televisores inteligentes Samsung. En el contexto de un termostato de aprendizaje, el dispositivo puede enviar datos a un repositorio de datos en la nube. La base de información de la nube te da una idea del consumo de energía de tu hogar [21]. Por otro lado, este tipo de conexión en la nube facilita el acceso remoto al control del termostato a través de un teléfono inteligente o una plataforma web de trabajo.

Sin embargo, pueden surgir problemas de compatibilidad al intentar integrar otros dispositivos de desarrollo [23]. Los dispositivos y los servicios en la nube a menudo usan la misma tecnología y pueden ser del mismo proveedor. Con el registro entre dispositivos y la compatibilidad con la nube, los usuarios a menudo confían en un servicio en la nube específico para limitar el uso de servicios alternativos [22].

Dispositivo a puerta de enlace

Para la comunicación entre el dispositivo y la puerta de enlace, el dispositivo principal debe conectarse a un dispositivo más pequeño que actúe como intermediario para acceder a los servicios proporcionados por la nube [24]. En la mayoría de los casos, este tipo de conexión significa que debe comprar un software o una aplicación que actúe como una puerta de enlace local entre su dispositivo y el servicio en la nube. Por lo tanto, las puertas de enlace brindan seguridad y otras características que les permiten realizar una función conocida como conversión de datos a protocolo [20].

Intercambio de datos a través del back-end

Este modelo intercambia datos a través del back-end, funciona gracias a estrategias de comunicación entre diferentes objetos inteligentes que pueden ser validados por diferentes usuarios [22]. Esto ya se ha logrado e incluso superado en esta época. Por lo tanto, el usuario puede acceder a los datos cargados por otros sensores a través de la aplicación [24, 28].

Este modelo de comunicación de un solo dispositivo a la nube brinda a Internet Society acceso a grandes cantidades de información recopilada por varias empresas. “Los dispositivos IoT cargan datos a un proveedor de servicios de aplicaciones” [24]. Esto permite que la arquitectura se comunique a través del backend, agregue datos y analice los datos recopilados. Un ejemplo es la aplicación Map My Fitness para este tipo de comunicación. Es uno de los dispositivos que está en aumento debido a la gran cantidad de usuarios que utilizan esta aplicación [22]. Lo que hace este dispositivo es recopilar datos de estado físico que se nos proporcionarán para futuros análisis.

2.1.2. Seguridad

En [24], la seguridad de datos para Internet de las cosas incluye ciertas consideraciones necesarias para proteger la información de las personas de la divulgación en entornos de IoT. En este entorno, prácticamente a cualquier entidad física o lógica se le asigna un identificador único y la capacidad de comunicarse de forma autónoma a través de Internet o redes similares. Los puntos finales (cosas) en un entorno IoT también se comunican con otros puntos finales porque transmiten datos de forma autónoma [24]. La interoperabilidad de las cosas es esencial para el funcionamiento del IoT. Los datos enviados desde un punto final en particular no deberían plantear ningún problema de privacidad. Sin embargo, incluso los datos fragmentados pueden exponer información confidencial cuando se recopilan, agregan y analizan en múltiples puntos finales [21, 24].

2.1.3. Privacidad

Según [22], los consumidores deben exigir mejores medidas de seguridad y privacidad para evitar que se vuelvan vulnerables a la vigilancia corporativa y las filtraciones de datos. Pero antes de que los consumidores exijan un cambio, deben ser notificados, lo que requiere más transparencia por parte de la empresa.

2.1.4. Interoperabilidad

En [29], los autores afirman que: “La interoperabilidad entre los sistemas de IoT es fundamental para lograr el máximo valor. En promedio, el potencial de las aplicaciones de IoT. La interoperabilidad se requiere al 40 % del valor y cerrar al 60 % en algunas configuraciones”.

Según [30], la interoperabilidad significa “la capacidad de los sistemas de TIC y los procesos de negocio que soportan para intercambiar datos y permitir el intercambio de información y conocimiento”.

En [31], por otro lado, menciona las diferentes capas que se enfocan en la interoperabilidad de las plataformas de IoT, con siete capas, pero las cuatro más relevantes, solo se enfoca y muestra los aspectos que afectan:

Interoperabilidad técnica: Por lo general, se asocia con los protocolos de comunicación y la infraestructura requerida para que estos protocolos funcionen.

Interoperabilidad sintáctica: Generalmente asociado con el formato y la codificación de los datos. B. XML, JSON y RDF.

Interoperabilidad semántica: Asociada a un entendimiento común del significado del contenido intercambiado (información).

Interoperabilidad organizacional: Está relacionado con la capacidad de la organización para transmitir y transferir información de manera eficaz.

2.1.5. Entornos para aplicaciones IoT

En [24], se indica los diferentes dispositivos de IoT se pueden implementar en diferentes entornos, para mejorar la recepción o la implementación de datos representativos en los entornos a continuación se detallan algunos entornos donde se pueden implementar estos dispositivo.

- Cuerpo humano
- Hogar
- Puntos de Venta
- Oficinas
- Fabricas
- Obras
- Vehículos
- Ciudades
- Exteriores

IoT continuamente transforma las industrias ya que nos ofrece la optimización de costos y productividad dentro de las empresas mediante el desarrollo de sistemas inteligentes, sin embargo, “la privacidad de la información juega un papel importante en las arquitecturas de las arquitecturas de IoT” [32] debido al incremento de los dispositivos dentro de una red publica o privada requiere de la mejora continua de la seguridad dentro de las organizaciones [33, 34, 35].

2.2. Sistemas inteligentes

Según [36, 37], los sistemas inteligentes poseen una gran capacidad de adaptación a problemas complejos brindando buenos resultados en diferentes áreas, la construcción de estos sistemas en la forma tradicional consiste en un proceso de datos usando varios paradigmas, de estos sistemas se espera el desarrollo de software que requieren simular la manera en la que los expertos humanos proceden para resolver un problema, encontrando esta solución mediante un proceso de búsqueda por criterios específicos de un área determinada, inicialmente las estrategias que se desarrollaron se basan en: Inteligencia Artificial y Técnicas de Machine Learning con aplicaciones en distintas áreas. En ciencias de la computación se utilizan principalmente agentes de software que consisten en un programa que calcula su resultado a partir de la entrada que recibe [38].

2.3. Inteligencia Artificial

Según [39, 40, 41], la inteligencia artificial es la capacidad de las máquinas para usar habilidades humanas como el pensamiento lógico, el aprendizaje, la creatividad y las habilidades de planificación. La IA permite que los sistemas tecnológicos perciban su entorno, interactúen con él, resuelvan problemas y actúen intencionalmente. Las máquinas reciben, procesan y responden a los datos (preparados o recopilados de un sensor). Los sistemas de IA pueden adaptar su comportamiento hasta cierto punto, analizar el impacto de comportamientos anteriores y actuar de forma autónoma. Este concepto se dio a conocer en la década de 1950 a través de las obras publicadas por Alan Turing en las que se buscaba descubrir cómo una computadora podía tener el poder de pensar o al menos brindar noticias todas basadas sobre las obras propuestas por Ada a mediados del siglo XIX. Aunque no tiene una definición universal, la inteligencia artificial se puede entender como: el esfuerzo por automatizar las tareas que normalmente realizan los humanos [40, 41]. Es a partir de este concepto que surge que surge la opción de pensar que los programas informáticos primitivos conocidos como inteligencia artificial simbólica diseñados para resolver una partida de ajedrez de cartas pueden ser entendidos como inteligencia artificial, esta programación depende directamente en un conjunto de reglas establecidas para ejecutar el juego [40]. En [41], indica el problema al que se enfrentan estos tipos de programas es que se ven limitados cuando resuelven tareas

más complejas como el reconocimiento facial o hacen predicciones futuras, por lo que es conveniente cambiar el paradigma en el que decide, asegurando que es el mismo el que define las reglas con las que se obtienen resultados a partir de datos dados, este paradigma se conoce con el nombre aprendizaje automático.

2.3.1. Técnicas utilizadas en inteligencia artificial

En [41], a partir de la perspectiva de la Ingeniería (inteligencia artificial) menciona distintas herramientas en la solución de inconvenientes, estas herramientas se muestran en diversas técnicas, mismas que dan recursos primordiales en las zonas de la resolución de problemas:

- Robótica
- Redes neuronales
- Sistemas expertos
- Algoritmos genéticos
- Análisis de datos
- Reconocimiento de patrones
- Reconocimiento de lenguaje natural

2.4. Machine Learning

El machine learning o aprendizaje automático es una rama del análisis de datos basado en inteligencia artificial que automatiza la creación y el uso de modelos estadísticos, que le da a un ordenador la capacidad de aprender de los datos, se basa en esos sistemas. Pueden analizar, estudiar y aprender de datos, modelos y determinar las acciones a tomar con mínima interacción de los seres humanos [42]. A continuación, se hallan las subcategorías de aprendizaje automático que definen los tipos de aprendizaje del algoritmo.

2.4.1. Tipos de aprendizaje

Los arquetipos de algoritmos de aprendizaje automático que concurren son: supervisado, no supervisado y por refuerzo.

Aprendizaje Supervisado

Un escenario supervisado se basa en la idea de un profesor cuya tarea principal es proporcionar al agente una medida precisa de su error comparable directamente con los valores de salida [43]. En este tipo de aprendizaje conocemos las respuestas correctas que el algoritmo debería ser capaz de predecir, el rol que desempeña el supervisor consta en comparar directamente estas salidas con las predicciones del algoritmo.

Clasificación El algoritmo al recibir nuevos valores predice la categoría o grupo a la que pertenece mediante el entrenamiento y los patrones encontrados, estos valores deben ser categóricos o discretos.

- K Vecinos más cercanos
- Clasificación de Naïve Bayes.
- Máquinas soporte vectorial.
- Redes neuronales para clasificación.
- Bosques aleatorios de clasificación

Regresión En este método el algoritmo predice un nuevo valor fijo respecto a los datos de entrenamiento.

- Árboles de decisión
- Regresión lineal
- Redes para regresión
- Regresión polinomial
- Vectores de soporte clasificación
- Bosques aleatorios

Aprendizaje no Supervisado

En este método el algoritmo predice un nuevo valor de acuerdo a las observaciones dadas sin importar las etiquetas de salida.

- Regresión múltiple o lineal.
- Regresión polinómica.
- Árbol de decisión.
- Random Forest.

Agrupación (Clustering) La agrupación en clústeres consiste en la técnica de análisis exploratoria de la data para establecer la información u objetos con características similares en grupos significativos sin tener un conocimiento previo de su estructura [42].

Aprendizaje por refuerzo

Este aprendizaje por refuerzo combina el aprendizaje supervisado y no supervisado, en este método se utiliza retroalimentaciones o también llamadas recompensas que evalúan la validez de los resultados precedidos [44]. En este tipo de aprendizaje si la respuesta fue acertada él se procede como el aprendizaje supervisado, por tanto, es un proceso donde se debe probar y obtener su error para tener como objetivo las recompensas o las penalizaciones.

2.5. Aplicaciones de Machine Learning

Entre los beneficios del machine learning destacan los mencionados a continuación:

- Detección de rostros.
- Antivirus.
- Forecasting.
- Análisis de imágenes.
- Procesamiento de textos.
- Genética.

2.6. Redes Neuronales

En [45, 46], indica que el término neurona se abrevia una vez que se complementa con otras neuronas, porque de esta manera se puede imitar al cerebro y, como tal, se puede completar el criterio. Según [46], una red neuronal se define como un grupo de neuronas concatenadas entre sí a través de una entrada secuencial que la salida de todas las neuronas devuelve a las primeras neuronas en funcionamiento de la arquitectura y después de completar el proceso de concatenación, la neurona genera el resultado final de toda la red, un grupo de valores de entrada en las condiciones establecidas a partir y dando lugar a una salida, esta arquitectura se puede observar en la Fig. 2-1.

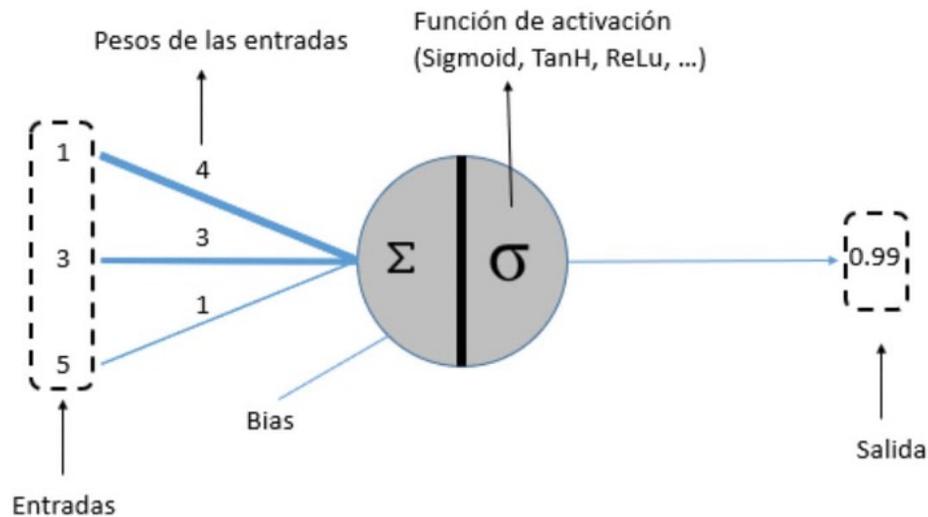


Figura 2-1: Red neuronal. Figura tomada [1]

Las diferentes funciones de activación son:

- Lineal
- No lineales
- Sigmoide
- ReLu
- Tangente hiperbólica
- Softmax

2.6.1. Red Neuronal Recurrente (RNN)

“Las redes neuronales recurrentes, o Recurrent Neural Networks(RNN) en inglés, son una clase de redes para analizar datos de series temporales permitiendo tratar la dimensión de tiempo” [47, 48]. Las redes recurrentes son dinámicas, es decir, también tienen conexiones ascendentes, en estos modelos la salida es dependiente de las variables en conjunto con los valores actuales de las características de entrada de la red, valores de entrada anteriores, etc. La arquitectura de la red neuronal recurrente se puede observar en la Fig. 2-2.

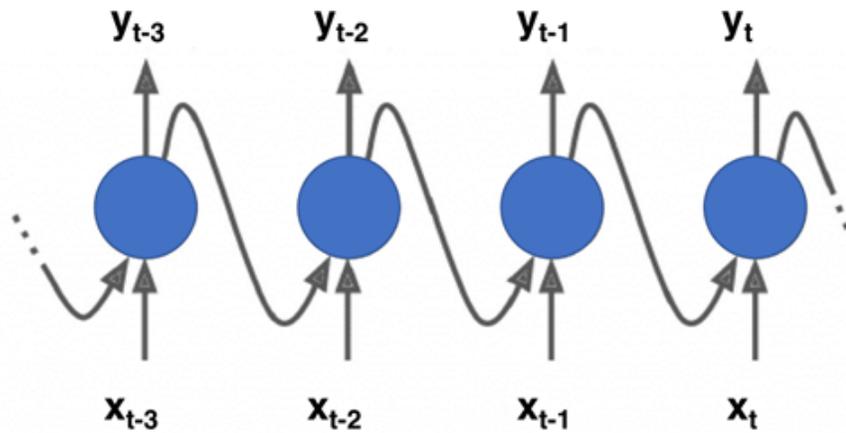


Figura 2-2: Red neuronal recurrente. Figura tomada [2]

Capas

En [45], indica que dentro de una red de neuronas puede concatenarse con otra, sin embargo, no todas son entre sí porque el proceso secuencial sigue una estructura determinada por una serie de capas. En las redes neuronales, una capa es una de las neuronas que no interactúan entre sí porque están al mismo nivel, pero que forman el vector de peso para una siguiente capa. El conjunto de capas de neuronas, según [46], forma el mapeo a partir del cual se obtiene el resultado de la red, este mapeo característico se compone de tres grupos: la capa de entrada, la capa de salida y las capas intermedias, cuyo número varía según su modelo. Cada una de estas capas actúa de forma independiente, convirtiendo la información característica recibida en factores que se comunican a nivel de arquitectura.

Parámetros

Los parámetros son la parte más relevante de un modelo de aprendizaje automático porque estos son variables ajustadas a conveniencia del algoritmo y son necesarios para los modelos que aprenden de los datos y estos modelos a su vez nos sirven para hacer predicciones [49].

Hiperparámetro

Los hiperparámetros son cambiadores de configuración que se realizan en gran parte fuera del modelo, sus costos generalmente no se pueden estimar a partir de los datos y el desarrollador del modelo sabe a detalle como ajustar el algoritmo de aprendizaje [50]. Para encontrar el valor óptimo de los hiperparámetros es necesario especificar la configuración y la estructura de toda la red neuronal como son: número de neuronas ocultas, número de capas función de activación, todos estos hiperparámetros son cambiantes hasta que el algoritmo de aprendizaje tenga su aprendizaje óptimo.

Lookback

Lookback es la cantidad de pasos de tiempo que se muestra en el modelo, y para lograr el costo óptimo del lookback va a depender del tipo de datos y la correlación de tiempo. Los valores para lograr este costo óptimo están en el rango 8-200 que se ha utilizado con éxito para el entrenamiento [51]. La red neuronal recurrente de tipo LSTM tiene la capacidad de aprender a menor tiempo mayor cantidad de datos, dado que este tipo de red pueden lograr almacenar más información a diferencia de una red neuronal recurrente normal.

Batch size

El tamaño de la pila (Batch size) se refiere al tamaño del lote o el número total de datos. Es decir, cuando se divide un dataset grande, este es el tamaño de lotes tomados de la parte del train para entrenar el modelo en cada época (epoch) con un número de datos en específico [45]. El número de fracciones también es interpretado como el número a iterar de cada epoch de entrenamiento con los datos de train del dataset.

Epoch

La época (epoch), según [45, 46] es aquel que indica el total de veces que se va a pasar los datos de entrenamiento a la red, también se deduce que el conjunto de datos completo se pasará al modelo dependiendo las épocas que se definan. Esto significa que, al tener varias épocas, el conjunto de datos se pasará varias veces a través de la arquitectura de la red propuesta.

Neuronas dentro de una capa oculta

La cantidad de neuronas que puede poseer una red neuronal depende de los datos de acceso, por lo que debe experimentar un poco con la cantidad de neuronas que usa para mejorar el rendimiento [45].

Función de activación

Esta función está predefinida en cada capa oculta y pertenece a la función de activación de las neuronas artificiales que simula la respuesta de las neuronas biológicas a estímulos específicos [46]. Aquí, la red neuronal recurrente se entrena y utiliza una función de activación exclusiva para el entrenamiento del modelo.

Factor de aprendizaje

Factor de aprendizaje se refiere a la rapidez con la arquitectura de la red neuronal aprende esto puede ser a una tasa de aprendizaje alta o baja, pero también brinda una rápida aproximación a los pesos óptimos, pero a valores más bajos los pesos necesitan ser reajustados

de forma más óptima [45]. Por lo que resulta muy importante encontrar el costo adecuado para el factor de aprendizaje.

Long short-term memory (LSTM)

Las LSTM al tener una arquitectura RNN son ideales para tratar con problemas de dependencia a largo plazo ya que recuerda valores en intervalos propuestos, generalmente la estructura de una LSTM contiene una unidad de compuerta de entrada protege el estado de la memoria de la perturbación por las entradas que no tienen relación, una compuerta de salida protege el estado de las otras células de entradas irrelevantes, una compuerta de olvido la cual permite que la memoria de la célula olvide eventos irrelevantes y una función de activación que calcula los estados de las celdas de memorias internas como se muestra en la Fig. 2-4. La conexión de la LSTM es de extremo a extremo, la salida de la capa anterior sirve para alimentar la siguiente capa obteniendo finalmente la salida [52]. Con la evolución de las LSTM se han creado variaciones con aplicaciones en el reconocimiento del habla y escritura, traducción y la predicción de series temporales [53].

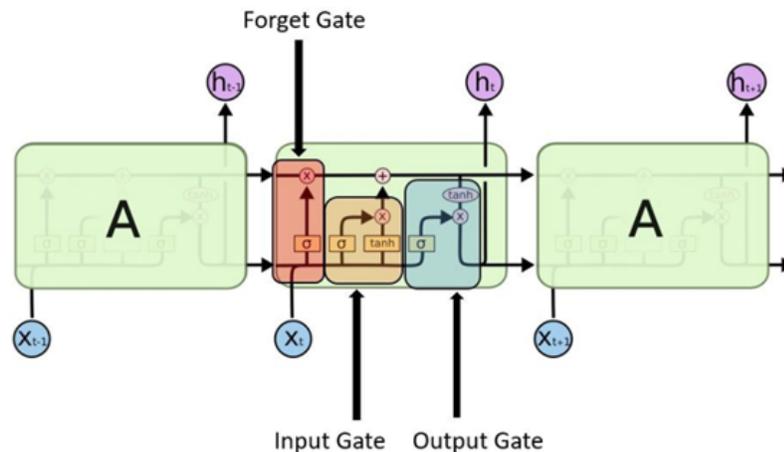


Figura 2-3: Estructura modelo LSTM. Figura tomada [3]

TimeDistributed

En [54], describe como un contenedor de clase, aplicar esta capa a cada segmento temporal de ingreso posibilita edificar modelos que poseen arquitecturas de uno a varios, de varios a varios.

RepeatVector

Esta capa repite la entrada 2D proporcionada varias veces para crear una salida 3D [54].

Preprocesamiento de datos

Esta fase radica en la limpieza, transformación y reducción para mejorar la calidad de los datos y obtener mejores resultados en el entrenamiento de la red neuronal, el preprocesamiento contiene técnicas que nos ayudan a extraer una representación de los datos originales [55].

Fase de entrenamiento

En esta fase se tiene datos con grandes dimensiones los cuales se separan para el entrenamiento de la red neuronal recurrente y permite hacer nuevas predicciones mediante los patrones encontrados [45].

Fase de prueba

Los datos restantes de la separación se usan para evaluar si las predicciones de la red neuronal recurrente son correctas, en caso de que las predicciones sean erróneas se debería incrementar el número de observaciones dentro de los datos [46].

2.7. Arquitecturas LSTM

A continuación se presentan los diferentes modelos con arquitecturas LSTM.

2.7.1. Modelo LSTM univariado

Según [56], son arquitecturas para series de tiempo con inconvenientes conformados por una sola serie de visualizaciones y es necesario un modelo para aprender de la serie de visualizaciones pasadas, para pronosticar el siguiente valor en la sucesión.

Clasificación del modelo LSTM univariantes

- Vanilla
- Apilado
- Bidireccional
- CNN-LSTM
- ConvLSTM

Vanilla LSTM

LSTM vanilla o los RNN clásicos son un tipo de arquitectura donde poseen unidades LSTM dentro de una capa oculta, dando así una capa de salida dependiente para hacer las predicciones. Cuando se entrena esta red neuronal utilizando retro propagación los gradientes pueden desaparecer o tender al infinito [57].

LSTM apilado

En esta arquitectura se tiene la posibilidad de acumular varias capas LSTM una encima de otra por lo cual su salida será producida por la sucesión de capas LSTM [56, 58]. Esto se puede lograr para cada paso de tiempo que se defina en el modelo con el argumento `return sequences True` en cada una de las capas, posibilitando así tener una entrada 3d de los datos en cada capa.

LSTM bidireccional

LSTM bidireccional para proyección de series temporales 1D consiste en envolver la primera capa oculta dentro de otra capa denominada bidireccional [59].

CNN LSTM

Una red neuronal convolucional, o CNN es un tipo de red neuronal hecha para laborar con datos de imágenes bidimensionales. La CNN podría ser bastante eficaz para sustraer y aprender automáticamente propiedades de datos de sucesión unidimensionales [60].

ConvLSTM

ConvLSTM se desplegó para leer datos en el espacio temporal bidimensional, sin embargo, se puede adaptar para su uso con predicciones de series de tiempo univariadas [59].

2.7.2. Modelos LSTM multivariados

Segun [61], las series de tiempo multivariantes son todos los dataset en los cuales hay muchas características para cada paso de tiempo.

Modelos series de tiempo multivariantes:

- Modelos con entradas múltiples
- Modelos de múltiples series paralelas
- Modelos de entrada múltiple con salida de varios pasos
- Modelos de múltiples entradas en paralelo y salidas de varios pasos

Serie de entrada múltiple

Las redes neuronales pueden dominar más de una serie temporal de entrada y una serie temporal de salida paralelas, según la serie temporal de acceso. La serie de tiempo de entrada es paralela porque cada serie contiene observaciones para el mismo paso de tiempo [61].

Modelos LSTM de varios pasos

Modelo de salida vectorial: En [61], indica que los modelos LSTM pueden generar vectores directamente que pueden interpretarse como predicciones de niveles múltiples. Aquí, los pasos de tiempo en cada serie de tiempo de salida se predicen como vectores.

Modelo codificador-decodificador: Este modelo especialmente diseñado para predecir secuencias de salida de longitud variable y se denomina EncoderDecoderLSTM [54]. Se creó un ejemplo de este modelo para un problema predictivo en el que en donde la secuencia de entrada y salida se denominan secuencia por secuencia, o para problemas como traducir un guión de un idioma a otro [62]. Como su nombre lo indica, este modelo consta de dos submodelos, un codificador y un decodificador como se puede observar en la Fig. 2-4, que se pueden utilizar para la predicción de series temporales de varios niveles.

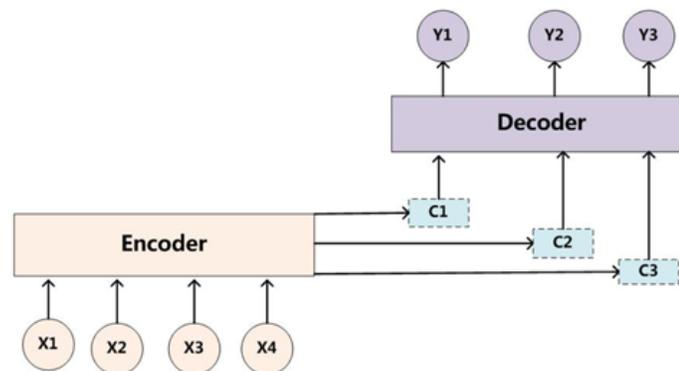


Figura 2-4: Arquitectura encoder-decoder LSTM. Figura tomada [4]

Entrada múltiple salida de varios Pasos

Los diferentes tipos de modelos LSTM presentados en esta sección se pueden combinar para series de tiempo con predicciones multivariadas y de varios pasos.

Múltiples entradas en paralelo y múltiples salidas

La predicción de series de tiempo multivariadas tiene la desventaja de que las series de salida están separadas, pero dependen de la serie de tiempo de entrada y la serie de salida requiere múltiples pasos de tiempo [54]. Estas salidas de un conjunto cambiante de datos de entrada difieren del aprendizaje tradicional de salida única en que son multivariadas y tienen la propiedad única de que las salidas tienen diferentes tipos de datos en diferentes pasos de tiempo. En la Fig 2-5 se puede observar como esta estructurada esta arquitectura.

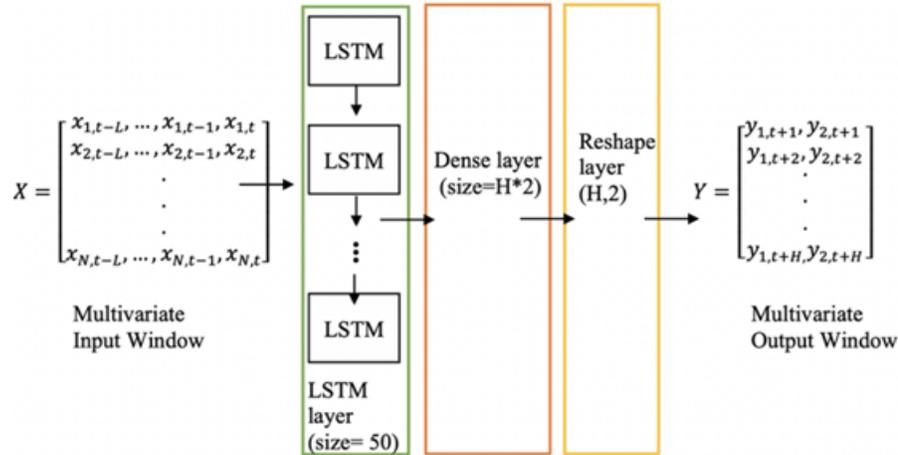


Figura 2-5: Estructuras Modelos LSTM múltiples entradas salidas de varios pasos . Figura tomada [5]

2.7.3. Validación del modelo (RNN)

Segun [51], una vez que se entrena la red neuronal, requiere una validación adecuada y la visualización de datos distintos a los utilizados en el proceso de entrenamiento.

Tipos de validación en modelos (RNN)

Bootstrapping: El método bootstrap le permite inferir la distribución de variables estadísticas de un grupo de muestras de población. En estadística, bootstrap se refiere a cualquier técnica que utiliza muestreo con recuperación [47]. El proceso de muestreo de recuperación consiste en devolver cada muestra de población recolectada aleatoriamente a esa población. Lo principal que muestra esta técnica es obtener las variables estadísticas deseadas.

Cross-Validation (LOOCV): Este método consiste en dividir los datos de entrenamiento y validación, estos datos se utilizan para calcular el error y variarán mucho según las observaciones que elija. Para evitar esto, el proceso se repite tantas veces como visualizaciones disponibles. Cada iteración excluye diferentes observaciones, ajusta el modelo al resto y calcula el error con esas observaciones [47].

2.7.4. Técnicas de validaciones en redes neuronales

Walk forward validation

Esta técnica es parecida a la validación acumulativa en donde las separaciones son las mismas y las validaciones se realizan cada vez que se actualicen los datos [63]. La duración de su

entrenamiento se puede ajustar para que coincida con los distintos datos que se separaron en su validación respectiva. Esto es muy común después de completar una validación de walkforward.

Validación cruzada

Las técnicas de validación cruzada se basan en dividir el dataset en diferentes subconjuntos, uno de los cuales se selecciona como prueba y se usa junto con el otro subconjunto para ejecutar el modelo de clasificación [63]. A continuación se presentan las técnicas de validación cruzada:

K-iteraciones: Los datos de la muestra se separan en K subconjuntos, de los cuales uno de ellos es utilizado como datos de prueba y lo demás ($K - 1$ subconjuntos) como datos de entrenamiento. Este proceso es repetido por K veces, permitiendo que cada subconjunto sea utilizado como datos prueba, pronosticando sus resultados.

Aleatoria: Es parecido al caso anterior, no obstante, la separación entrenamiento y prueba de los resultados finales se calculan de igual manera, como el promedio de las métricas calculadas en cada partición.

Dejando uno fuera: En inglés Leave one out cross validation (LOOCV), separa los datos de manera que en cada iteración se tenga una muestra para los datos de prueba y lo demás para los datos de entrenamiento.

Validación por división de datos: Este procedimiento de validación interna se basa en dividir de manera aleatoria la muestra original en 2 submuestras, una donde se lleva a cabo el modelo de entrenamiento y otra donde se valida el modelo prueba.

2.8. Series de tiempo en arquitecturas LSTM

En [64, 65], deducen que las series de tiempo muestran patrones de comportamiento tales como tendencias, periodos y estacionalidad, pero también anomalías como cambios estructurales, valores atípicos, etc. La flexibilidad de construir tipologías de modelos LSTM les permite identificar y predecir fácilmente patrones de comportamiento en la serie facultando que las redes neuronales representen comportamientos complejos, independientemente del dominio del conocimiento del que proceden los datos o la frecuencia temporal.

2.8.1. Componentes de una serie de tiempo

Dentro de las series de tiempo existen 4 factores que pueden atribuir características para una relación de una serie temporal.

Tendencia secular

Las tendencias se ajustan a diferentes esquemas en periodos largos de tiempo. Algunos seguirán subiendo, algunos caerán, mientras que otros permanecerán igual por un período de tiempo o por un período de tiempo. Una variación de la serie en sí. Se considera el resultado de una fuerza duradera que influye en su crecimiento o declive [65].

Variación estacional

Las fluctuaciones en los datos debidas a influencias estacionales se denominan componentes estacionales. Esta fluctuación corresponde a una serie de movimientos que se repiten cada año con aproximadamente la misma intensidad en el mismo mes (o trimestre) del año [65].

Variación cíclica

Las series de tiempo a menudo representan una secuencia alterna de puntos por encima y por debajo de una línea que demuestra la tendencia que dura más de un periodo de tiempo. Esta variabilidad se conserva incluso después de eliminar la variabilidad o la tendencia estacional e irregular [65].

Variación Irregular

La series de tiempo a menudo cambian a factores a corto plazo esto debido a repeticiones que afectan en la series temporal, esta variación es aleatoria no se puede deducir su patrón de comportamiento [64].

.

Existen dos tipos de variaciones:

- **Irregularidades:** Las fluctuaciones causadas por hechos especiales y fácilmente reconocibles como elecciones, inundaciones, huelgas y terremotos.

- **Fluctuaciones aleatorias o accidentales:** La causa no se puede precisar exactamente, pero tiende a equilibrarse a largo plazo.

2.8.2. Tendencia de una serie de tiempo

Tendencia lineal

Las tendencias de la serie se caracterizan por tener un movimiento de aumento o disminución constante a largo plazo de una serie temporal. Estas tendencias a largo plazo se pueden observar en muchas series de negocios comerciales e industriales, exportaciones y manufactura, a menudo convergen linealmente.

Tendencia no lineal

La tendencia de la serie no es lineal, por lo cual tiene un comportamiento curvilíneo. Las tendencias no lineales que pueden ocurrir en una serie incluyen, entre otras, tendencias polinómicas, logarítmicas y potenciales.

2.8.3. Medidas de error de pronóstico series de tiempo

Para determinar si un modelo es aceptable para el pronóstico de las series de tiempo se deben tomar en cuenta medidas de error como las que mencionamos a continuación:

Error absoluto medio

Esta medida de error (MAE, por sus siglas en inglés) se genera con la predicción, en la cual validamos el rendimiento de las predicciones, diferenciando el promedio de las predicciones obtenidas y los valores reales [66].

Error cuadrático medio

En esta medida de error (MSE, por sus siglas en inglés) se toma la diferencia entre los valores reales y los pronosticados elevando al cuadrado y a la vez penalizando las diferencias mas grandes entre los valores, por tanto, debemos obtener pronósticos con errores moderados y evitar la frecuencia de errores grandes [67].

Raíz cuadrada del error cuadrático medio

Esta medida de error (RMSE, por sus siglas en inglés) se calcula al igual que MSE, con la diferencia de que realiza la raíz cuadrada al finalizar el cálculo, esta medida también castiga las diferencias mayores, esta en las mismas unidades de la serie original por lo que es mas fácil interpretar su valor [67].

2.9. Preprocesamiento de datos en series de tiempo y arquitecturas LSTM

El preprocesamiento de datos es un paso esencial para entrenar un modelo LSTM usando series de tiempo, podemos entender por preprocesamiento a cualquier proceso realizado para convertir datos sin procesar en datos en un formato más fácil y manejable.

2.9.1. Selección de datos

Los conjuntos de datos a veces tienen la posibilidad de ser pequeños en lo que otros son tremendamente gigantes en tamaño, en particular una vez que cuentan con un enorme conjunto de propiedades, provocan que sean bastante difícil de procesar un modelo [68]. Una vez que se tiene este tipo de conjuntos de datos de alta magnitud y se usa cada una de las características para la construcción de modelos de Machine Learning, el resultado de no realizar un preprocesado de datos puede ser un modelo bajo en rendimiento.

2.9.2. Preprocesado de datos

Integración de datos

Los datos pueden estar distribuidos en distintas ubicaciones que corresponden a la misma información, tienen la posibilidad de exponer formatos diferentes en los diferentes períodos de tiempo, gracias a modificaciones del instrumento de recolección, de forma que deberán ser sometidos a un proceso de unión [55].

Análisis exploratorio

La exploración exploratoria de datos (EDA) es usado por los científicos de datos para examinar y averiguar conjuntos de datos, así realizar un resumen de sus primordiales propiedades, usando constantemente procedimientos de visualización de datos. Esto ayuda a establecer de mejor forma como manipular las características de los datos para obtener las respuestas que se requiere, lo cual posibilita a los científicos de datos hallar patrones, identificar anomalías, probar un modelo o revisar premisas.

Reconocimiento y limpieza datos

Para esto, se selecciona una muestra de los datos, logrando así interpretar la validez de cualquier característica para cualquier atributo y mejorar la calidad de los datos. Para este método se deberá revisar la proporción de datos nulos y los outliers (datos erróneos), para minimizar las maneras de un posible error una vez que sean procesados por el algoritmo [55].

2.9.3. Selección de Características

Transformación de datos:

Esto involucra la transformación del tipo de ciertos atributos, en caso de que fuera primordial, teniendo presente que transformar el tipo de un atributo a otro puede modificar la semántica de dicho atributo. Este paso está bastante ligado al algoritmo que procesará los datos. Las técnicas usualmente utilizadas son:

Escalado o Normalización Transforma las características en un rango especificado.

Estandarización: Transforma las características de manera que la distribución resultante tiene una media de 0 y una desviación estándar de 1.

Reducción de datos

- Selección de atributos, hay numerosas técnicas para realizar la labor de selección de los atributos importantes como son: Procedimientos basados en filtros, por medio de los cuales se filtran los datos anteriores a ser procesados por el algoritmo.
- La sustracción de atributos podría ser vista como una proyección del espacio de análisis, ya que posibilita cambiar el espacio de atributos, obteniendo otro espacio de atributos que represente la misma información de distinta forma.
- Creación de Atributos: Si se muestran patrones complicados en los datos se construirá unas diferentes técnicas: creación guiada por los datos o por el modelo.

2.10. Técnicas de visualización para la predicción de una serie temporal mediante una arquitectura LSTM

Las técnicas de visualización son representaciones visuales y estas son utilizadas para analizar de manera fácil el comportamiento de los datos masivos que generan las organizaciones tanto en el campo de la ciencia, ingeniería y matemáticas, según los autores [69, 70] son representaciones para el ser humano representadas por computadora en la Fig. 2-6 se puede observar estas representaciones, dando la posibilidad de detallar el formato y la forma de los datos para así determinar información relevante sobre el análisis o investigaciones complejas para la toma de decisiones e identificación de patrones. Entre las técnicas de visualización que pueden ser utilizadas para la visualización de la predicción de la serie temporal están:

Gráfico lineal: Esta es utilizada para observar la relación entre los elementos con un periodo de tiempo dado.

Gráfica de barras Esta gráfica es utilizada para comparar diferentes categorías donde cada una de estas posee diferentes cantidades.

Gráfica de dispersión: Esta gráfica representa la variación de dos elementos.

Gráfico Circular: Esta gráfica es utilizada para representar las fracciones de lo que representa un solo elemento.



Figura 2-6: Ejemplos técnicas de visualización. Figura tomada [6]

De todas las técnicas antes mencionadas las redes neuronales de tipo LSTM con el uso de series de tiempo pueden ser aplicadas para posibilitar la optimización de la gestión del consumo de agua potable.

2.11. Consumo de agua potable

El comportamiento respecto al consumo de agua potable en los seres humanos esta sujeto a muchos factores y variables tales como su cronología y el estado fisiopatológico, así también en la distribución influye directamente factores como la edad, sexo, raza, volumen corporal, temperatura, metabolismo, estado de salud, actividades físicas, dieta, medicación y entre otros [71]. “En la ciudad de Cuenca existen tres captaciones importantes en tres de los cuatro ríos de Cuenca: Planta El Cebollar, capta el río Tomebamba, con una capacidad instalada de 1000 l/s, Planta de Tixán, capta el río Machángara, con 860l/s y Planta de Sustag, capta el río Yanuncay, con 460 l/s, además otras 16 plantas de menor escala distribuidas en: Iquis, Cumbe, San Pedro, Quingeo, Tutupali, Sinincay, Santa Ana, Sayausi, Checa, entre otras,

con una capacidad total de 213 l/s, adicionalmente se encuentra en construcción la planta de culebrillas con una capacidad de 150 l/s. Para garantizar la continuidad del servicio de agua potable existen 33 centros de reserva ubicados en varios sectores de la ciudad, con una capacidad total de 120000 m³ [72], en la Fig. 2-7 podemos observar el consumo mensual de agua en m³ de las diferentes provincias del Ecuador, ubicándose la provincia del Azuay en segundo lugar, donde se encuentra la ciudad de Cuenca.

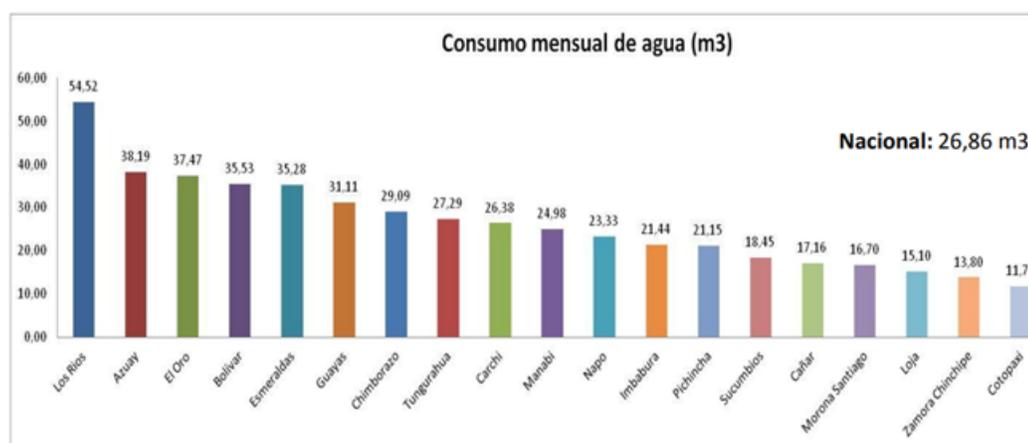


Figura 2-7: Indicadores de consumo de agua Ecuador. Figura tomada [7]

2.11.1. Gestión de demanda

El crecimiento de la poblacional muestra una elevación continua por tanto la gestión de la demanda de agua necesaria para cumplir con las necesidades básicas de los usuarios requiere de una planificación para las infraestructuras, estrategias para la distribución, recursos legales, campañas de concientización y recursos tecnológicos para controlar y regular el consumo de agua potable, sin embargo, la gestión se ha venido realizando mediante responsabilidad directa de los consumidores teniendo en cuenta factores como la temperatura, lluvias, humedad y por otro lado el factor social como el número de habitantes por domicilio, educación, precio del agua, así como también es importante mencionar el factor cultural en el cual están los valores y las normas [73].

3 Solución Propuesta

En la Fig. 3-1 se puede observar la solución propuesta y sus fases para el desarrollo del proyecto:

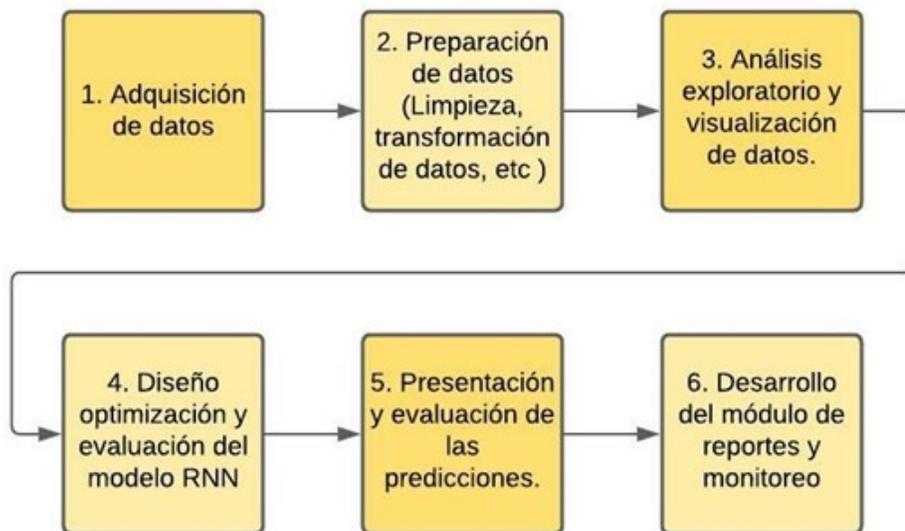


Figura 3-1: Diagrama de la solución propuesta

3.0.1. Primera fase

El proyecto se compone de 6 fases expuestas en la Fig. 3-1, la primera consiste en la adquisición de los datos de consumo de agua potable por usuario y sectores, en este caso los datos usados son genéricos.

3.0.2. Segunda fase

Consiste en la preparación de los datos, para lo cual se procede a recopilar datos históricos de los consumidores residenciales, una vez obtenidos los datos continuamos con el preprocesamiento de los datos, esto incluye todas aquellas modificaciones efectuadas sobre los datos con la finalidad de que puedan ser interpretadas por el algoritmo, esta fase es importante

para lograr que información no deseada influya en el ajuste del modelo. Según [74, 68], algunos de los pasos que se incluyen en el preprocesamiento son la transformación de datos, normalización, integración, limpieza de ruido, selección de atributos e imputación de valores perdidos.

3.0.3. Limpieza de valores ausentes

En [55], los autores deducen que los algoritmos en su gran mayoría no contienen o aplican observaciones, por lo cual cuando el conjunto de datos tiene valores ausentes podemos emplear las siguientes técnicas:

- Eliminar las observaciones incompletas.
- Eliminar las variables que contienen valores perdidos
- Intentar estimar los valores perdidos utilizando el resto de la información disponible.

Estandarización y escalado de variables numéricas

Según [68], la escala es importante si necesitamos especificar que un cambio de una cantidad no es lo mismo que otro cambio. Con la ayuda del escalado, se asegura de que, si bien algunas funciones son interesantes, no se utilizan como predictor principal. La normalización implica la transformación o conversión de datos en una distribución normal.

Transformación de datos

Con el dataset libre de valores ausentes, estandarizado y escalado procedemos a transformar la data con el objetivo de mejorar el conjunto de datos. Con esta transformación buscaremos crear nuevas columnas indexadas con el fin de tener un mejor orden y continuidad que es lo más importante. Entre otras cosas más se planea unir los dataset que contiene el conjunto de datos a fin de tener un dataset maestro que contenga toda la información necesaria para el respectivo análisis. Esto se realiza para tener una data muy bien estructurada.

Separación de Train y Test

Con el conjunto de datos bien estructurado procedemos a separar la data para el respectivo entrenamiento y validación. Esto con el fin de entrenar el modelo con el conjunto de datos de entrenamiento y realizar las predicciones con el conjunto de datos de prueba. Para realizar esta separación tomaremos en cuenta la serie de tiempo, por tanto no se alterara el orden, esto se realiza con el fin de no perder continuidad en la data.

3.0.4. Tercera fase

Consiste en el análisis exploratorio y visualización de los datos que nos sirve para darnos una idea de cómo se encuentran nuestros datos, comprenderlos y seleccionar sus principales características antes de decidir la técnica a implementar en el modelo RNN propuesto, esto nos permite descubrir patrones y anomalías para probar hipótesis o supuestos.

3.0.5. Cuarta fase

Es el desarrollo, optimización y evaluación de los modelos de RNN, en esta fase posterior al análisis de datos nos centramos en la creación de 4 diferentes arquitecturas que son:

- Codificador-decodificador LSTM.
- Codificador-decodificador LSTM donde la parte del decodificador consta de una capa bidireccional.
- Salida Vectorial para múltiples pasos
- CNN-LSTM de Múltiples cabezales para varias salidas

Por tanto, realizaremos una serie de experimentos a fin de encontrar el mejor modelo para realizar las predicciones. Seleccionado el mejor modelo procedemos al entrenamiento con el conjunto de datos preprocesado adquiriendo resultados que nos permitirán ajustar los parámetros y optimizar el modelo de la RNN, con el fin de mejorar la precisión del algoritmo se utilizarán medidas de error que se calcularán después de haber entrenado la red. Las medidas de error a utilizar son las siguientes:

- MAE o error absoluto medio nos permiten conocer la diferencia entre la predicción y su valor real.
- RMSE o el error cuadrático medio de la raíz calculará la cantidad entre dos conjuntos de datos, mediante una comparación de un valor de predicción con un valor conocido.

3.0.6. Quinta fase

Consiste en la presentación y evaluación de las predicciones, posterior al desarrollo del modelo procedemos a realizar predicciones con el conjunto de datos de prueba separados en la fase 2. Una vez realizadas las predicciones podemos evaluar qué tan precisas son las predicciones comparadas con los meses anteriores y validar que tan bueno es el modelo.

3.0.7. Sexta fase

Consiste en desarrollar un módulo de reportes utilizando técnicas de visualización, estas gráficas muestran el comportamiento del consumo del agua potable con aumento y reducción del 10 por ciento a la predicción de los siguientes meses con la finalidad de contribuir a la reducción del consumo de agua potable por usuario y sector, además posterior a este trabajo se prevé monitorear en tiempo real el consumo de agua mediante sensores inteligentes obteniendo más información sobre el consumo diario de agua.

4 Desarrollo de la solución propuesta

En la presente sección se describe el desarrollo de la propuesta de solución y la implementación de las diferentes arquitecturas propuestas que son para la predicción del consumo de agua potable por usuario y sector.

El [75], indica que un modelo de red neuronal recurrente requiere de preprocesamiento de las nuevas entradas ya que deben tener el mismo formato y las dimensiones establecidas por el número de pasos hacia atrás en el tiempo y el número de proyecciones que vamos a obtener como resultado. Las entradas que recibe la red neuronal recurrente para las nuevas predicciones se estable el número de fechas hacia a tras las cuales son seleccionadas para obtener las nuevas proyecciones de consumo ya sean estas fechas mensuales o trimestrales. En el preprocesamiento de los datos se busca obtener el mismo formato que se usó para el entrenamiento y la validación del modelo RNN, además, la longitud del tamaño del conjunto de datos para la obtención de nuevas predicciones debe ser igual al número de pasos de mira hacia atrás establecidos en el diseño de la red neuronal recurrente. Una vez finalizado el preprocesamiento de los datos para las nuevas predicciones enviamos a predecir a través del modelo, el cual nos devolverá un conjunto de datos para las fechas de proyección establecidas durante el diseño de la red neuronal recurrente.

4.1. Arquitecturas implementadas

En esta sección se describe la implementación de las 4 arquitecturas definidas en la fase 4 de la propuesta de solución para la predicción del consumo de agua potable por usuario y sector.

4.1.1. Codificador-decodificador LSTM

En la Fig. 2-5, se puede apreciar como esta compuesta la arquitectura encoder-decoder, a continuación en la Fig. 4-1 se muestra la arquitectura implementada, donde la primera capa es un codificador el cual se encarga de interpretar la entrada, donde se considera la forma de la entrada de los datos en este caso el tamaño de la ventana de 2 meses de aprendizaje, para la salida de consumo de 3 meses de 10 usuarios ,posterior agregamos una capa LSTM de tamaño 125, esta recibe los datos de entrada de la primera capa, esto producirá un vector de salida en cada paso de tiempo, luego agregaremos una capa repeat vector que devuelve todas las salidas que el codificador observó en el pasado, mientras que repite la última salida

del codificador que representa (factor de repetición, muestras, características), esta capa nos ayuda para la parte del decodificador, por ultimo esta la capa del decodificador, donde esta compuesta por una capa LSTM de 125 la cual espera un vector del ultimo paso de tiempo pasándole a una capa Dense de 64 y 32 para conectar las neuronas con la salida de la capa anterior LSTM, esta genera un vector en cada paso de tiempo, por lo que debemos aplicar la capa densa explicada anteriormente en cada paso de tiempo y predecir un consumo a la vez. Para hacer esto, keras ha desarrollado una capa específica llamada TimeDistributed, aplica la misma capa densa a cada paso de tiempo.

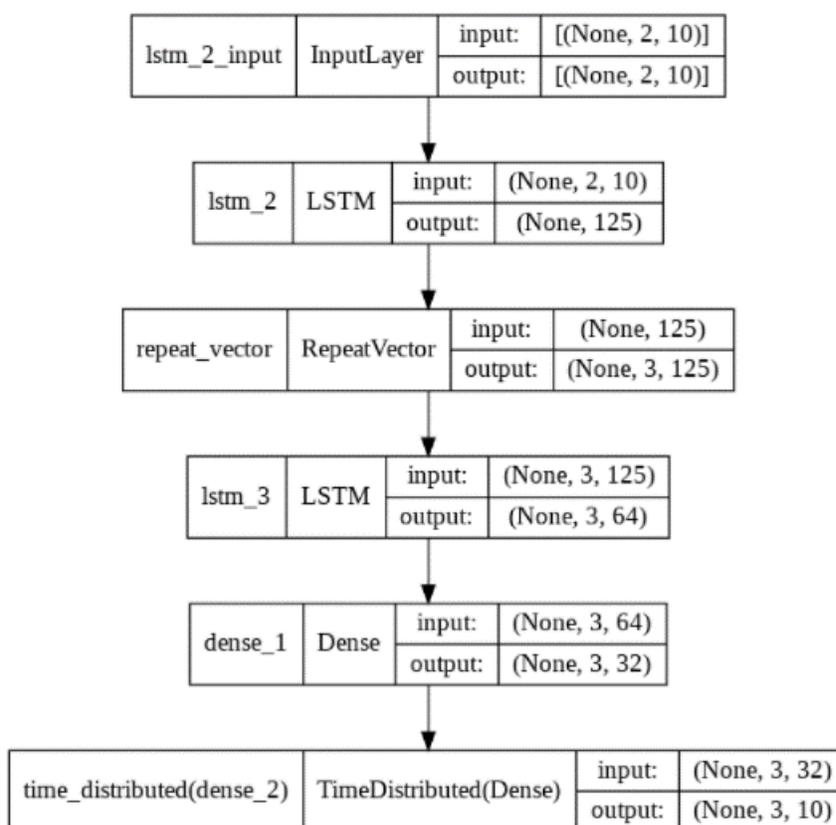


Figura 4-1: Arquitectura encoder-decoder

4.1.2. Codificador-decodificador donde la parte del decodificador consta de una capa bidireccional

En la Fig. 4-2 se puede observar la arquitectura implementada del modelo codificador-decodificador con integración de una capa bidireccional en el decodificador por lo general esto es mejor, porque la información en el momento siguiente depende de la información en el momento anterior.

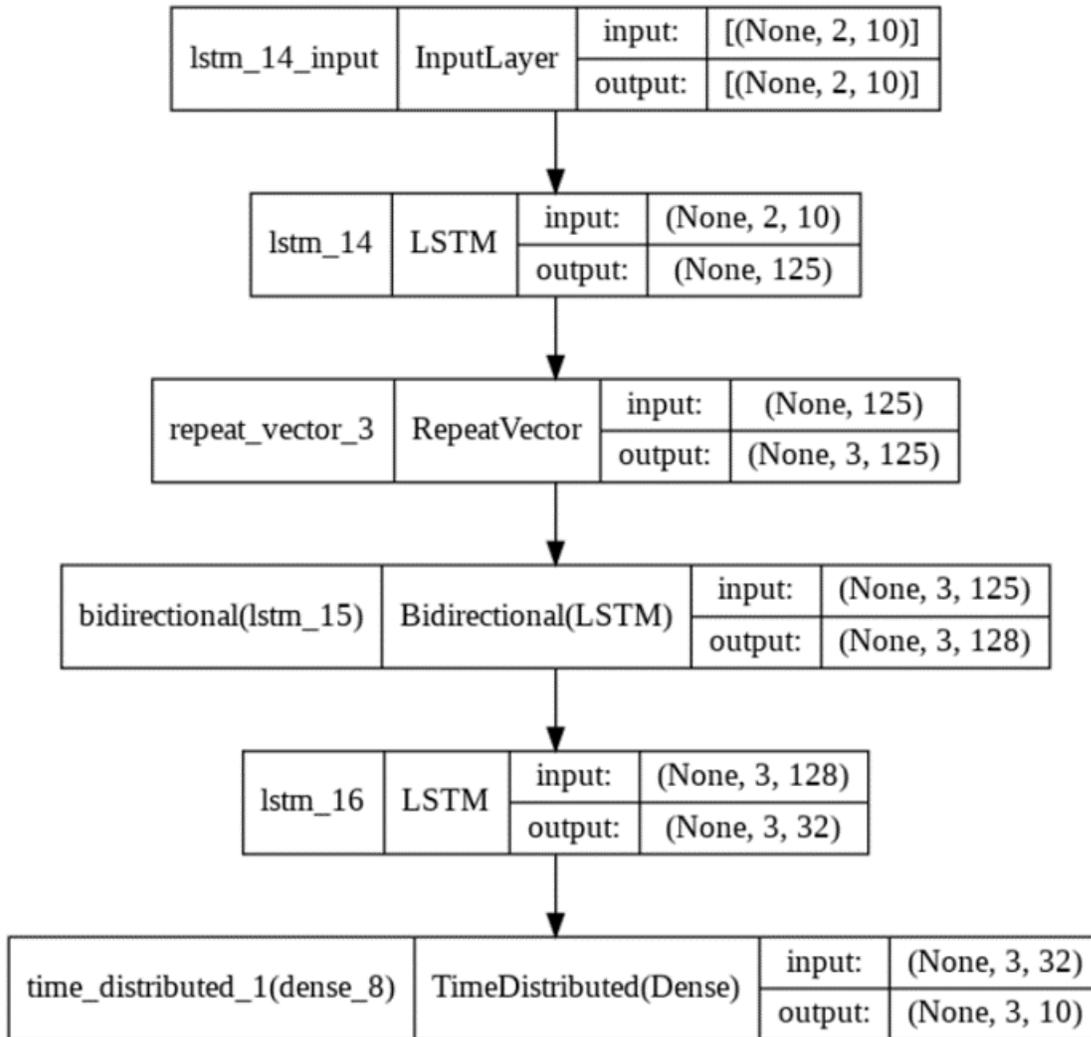


Figura 4-2: Arquitectura encoder-decoder con una capa Bidireccional en el decoder

4.1.3. Salida vectorial para múltiples pasos

En la Fig. 4-3 se puede observar la arquitectura implementada del modelo Salida Vectorial para múltiples pasos, en la primera parte, la capa densa se agrega como un nodo FORECAST-RANGE x n-features, y en la siguiente capa se modifica en consecuencia. En esta estructura, los pasos de tiempo de cada serie se combinan, y en el entrenamiento y la predicción, cada resultado debe interpretarse como un paso de tiempo específico para una serie específica. Las capas Conv1D también se utilizan para representar objetos 3D al principio de la arquitectura. Esta estructura también puede denominarse modelo multicanal. Tradicionalmente, se utilizan CNN con múltiples canales. En esta estructura, cada canal corresponde a una serie temporal y, de manera similar, se extraen características de contorno por separado para cada serie temporal.

Dado que todas las características extraídas se combinan antes de enviarse a la capa LSTM, es posible que falten algunas características generales de cada serie temporal.

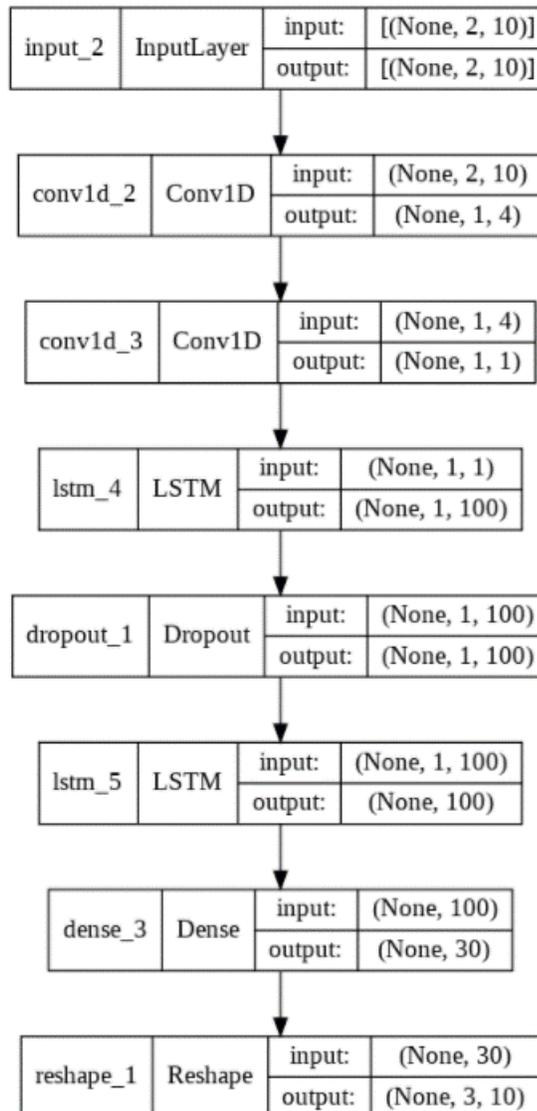


Figura 4-3: Arquitectura salida Vectorial para múltiples pasos

4.1.4. CNN-LSTM de Múltiples cabezales para varias salidas

En la Fig. 4-4 se puede observar la arquitectura implementada del modelo Salida Vectorial para múltiples pasos, el marco de varios cabezales utiliza múltiples capas CNN ONEDIMENSIONAL para procesar cada serie temporal y extraer características de contorno que son independientes de cada serie temporal. Estas CNN individuales se denominan “cabezas” se suavizan, fusionan y modifican antes de alimentar la capa LSTM. Es decir, las compilacio-

nes de varios cabezales usan varias CNN en lugar de una sola CNN, como las compilaciones de varios canales. Por lo tanto, puede tener éxito en preservar las características importantes de cada serie de tiempo y puede hacer predicciones más precisas.

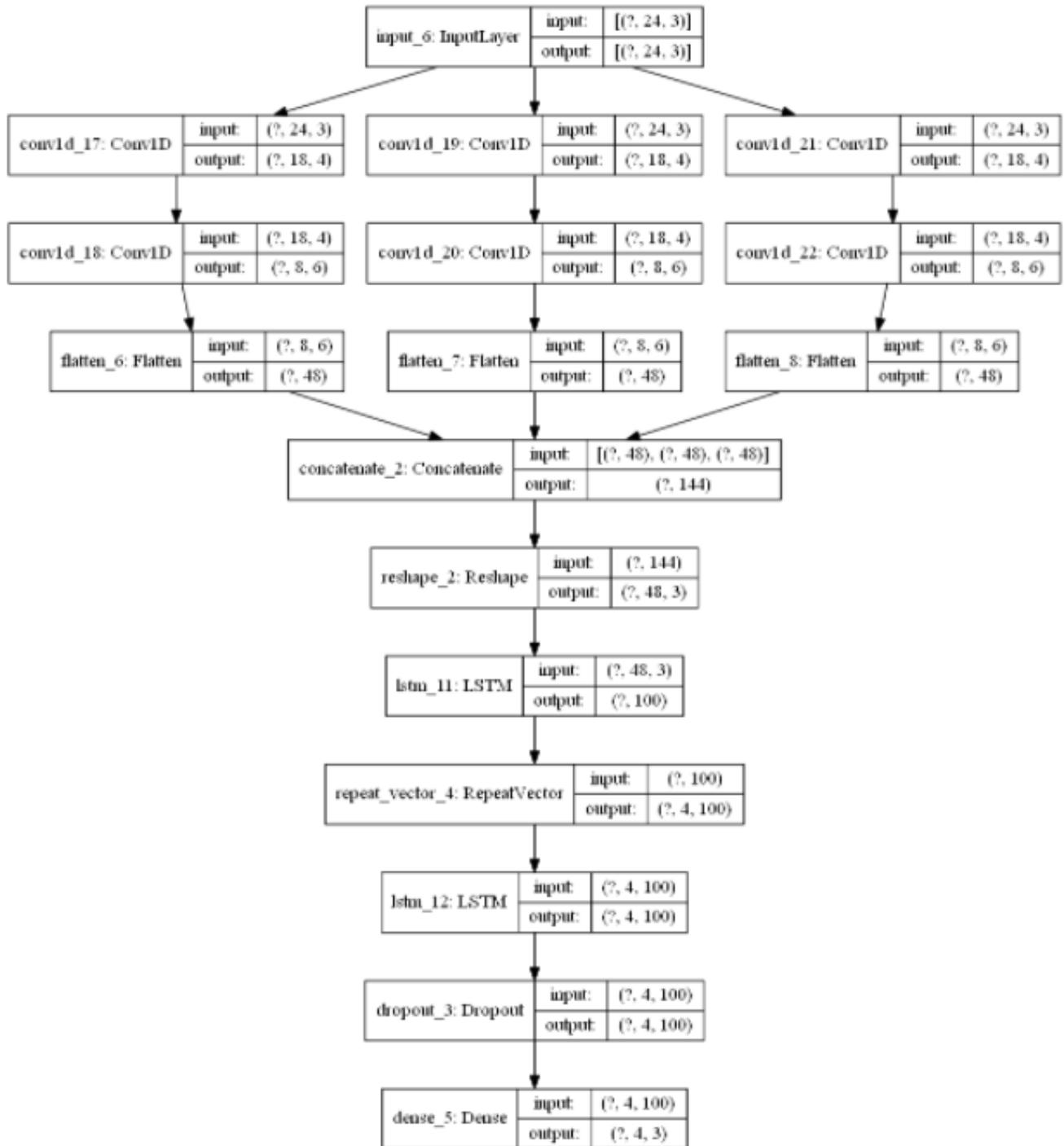


Figura 4-4: Arquitectura CNN-LSTM de Múltiples cabezales para varias salidas

En base a las investigaciones realizadas, la Tabla 4-1 contiene los diferentes hiperparametros y las posibles combinaciones que son para las diferentes arquitecturas implementadas.

HIPERPARAMETROS	VALORES
INPUT VENTANA	2,3,4,5
NODES	125,64,32
EPOCH	100,200,500
FILTERS	1,2,3
Validación	0.00001
Batch size	16,34,64
ACT HID	Relu, Sigmoid
ACT OUT	Relu, Sigmoid

Tabla 4-1: Hiperparametros definidos para los diferentes modelos implementados

Las predicciones obtenidas requieren de un proceso de edición en la forma de datos, en el cual se agregan los índices de las nuevas fechas que se incrementan mensualmente, posteriormente obtenemos gráficas de reportes del consumo histórico, predicción y la reducción en un cierto porcentaje en la predicción.

Finalmente, se calculan los consumos en valores monetarios tanto en la predicción como en la predicción reducida, mediante estos cálculos y visualización aportamos con información valiosa para el usuario quien podrá tomar decisiones respecto a su comportamiento en el consumo de agua potable

5 Resultados

Sobre la base de las pruebas realizadas se presentarán en este capítulo, se analizarán las hiperparametros definidos en la tabla 4-1, se han realizado pruebas en cada una de las arquitecturas implementadas para el presente trabajo, basándose en el desempeño del error RMSE y MAE, así como los hiperparametros seleccionados para el modelo que mejor sea su evaluación, según la evaluación del modelo el mejor va ha ser el que mas se aproxime a 0 su RMSE o MAE.

5.1. Codificador-decodificador LSTM

En la tabla 5-1 se muestran todos los experimentos y resultados obtenidos con los diferentes hiperparametros en la arquitectura de la Fig. 4-1.

N Experimento	Hiperparametros probados en el modelo	MAE	RMSE
	$[input, ventana, n, odes, e, poch, b, atch, act_h, id, act_o, u]$		
N1	[[2,3], 125,64,32, 100, 16 ,relu, sigmoid]	0.039	0.048
N2	[[3,3], 125,64,32, 100, 16 ,relu, sigmoid]	0.036	0.044
N3	[[4,3], 125,64,32, 100, 16 ,relu, sigmoid]	0.034	0.043
N4	[[5,3], 125,64,32, 100, 16 ,relu, sigmoid]	0.058	0.073
N5	[[2,3], 125,64,32, 200, 34 ,relu, sigmoid]	0.044	0.058
N6	[[2,3], 125,64,32, 200, 64 ,relu, sigmoid]	0.109	0.131
N7	[[2,3], 125,64,32, 500, 16 ,relu, sigmoid]	0.029	0.038

Tabla 5-1: Hiperparametros probados en la arquitectura 1

En la Fig. 5-1 se muestran la comparación de las evaluaciones de la arquitectura con el error MAE y RMSE en los diferentes experimentos realizados con los hiperparametros definidos.

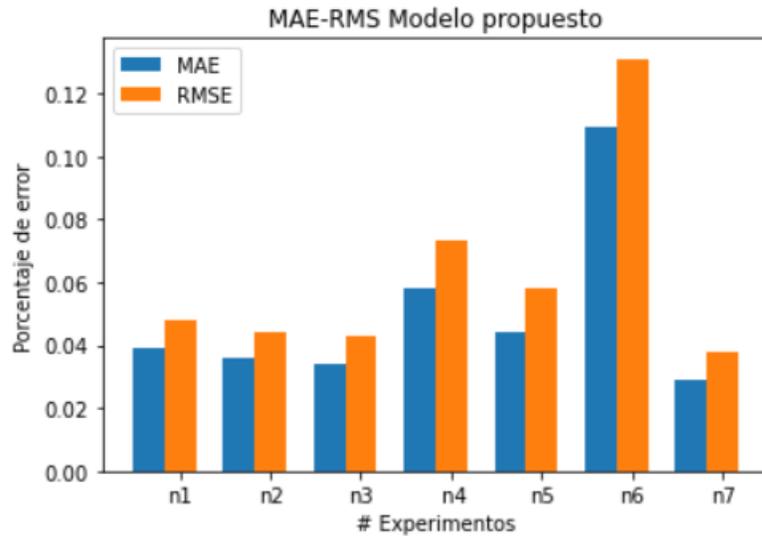


Figura 5-1: Grafica comparativa de error MAE vs RMSE arquitectura 1

En esta arquitectura se puede apreciar que los porcentajes de error RMSE y el MAE están en el rango de 0.02 a 0.13, el error mas cercano a 0 obtenido con el MAE es de 0.029, por lo tanto es un buen modelo para la predicción de consumo de agua, ya que mientras el error sea mas bajo se considera que el modelo se ajusta de manera óptima para la predicción.

5.2. Codificador-decodificador donde la parte del decodificador consta de una capa Bidireccional

En la tabla 5-2 se muestran todos los experimentos y resultados obtenidos con los diferentes hiperparametros en la arquitectura de la Fig. 4-2.

N Experimento	Hiperparametros probados en el modelo	MAE	RMSE
	$[input, entana, n, odes, e, poch, b, atch, act_{hid}, act_{ou}]$		
N1	[[2,3], 125,64,32, 100, 16 ,relu, sigmoid]	0.035	0.046
N2	[[3,3], 125,64,32, 100, 32 ,relu, sigmoid]	0.038	0.048
N3	[[3,3], 125,64,32, 100, 16 ,relu, sigmoid]	0.038	0.049
N4	[[3,3], 125,64,32, 500, 64 ,relu, sigmoid]	0.113	0.139
N5	[[4,3], 125,64,32, 100, 16 ,relu, sigmoid]	0.044	0.055
N6	[[2,3], 125,64,32, 500, 16 ,relu, sigmoid]	0.033	0.041
N7	[[2,3], 125,64,34, 500, 128 ,relu, sigmoid]	0.154	0.178

Tabla 5-2: Hiperparametros probados en la arquitectura 2

En la Fig. 5-2 se muestran la comparación de evaluación de la arquitectura con el error

MAE y RMSE en los diferentes experimentos realizados con los hiperparametros definidos.

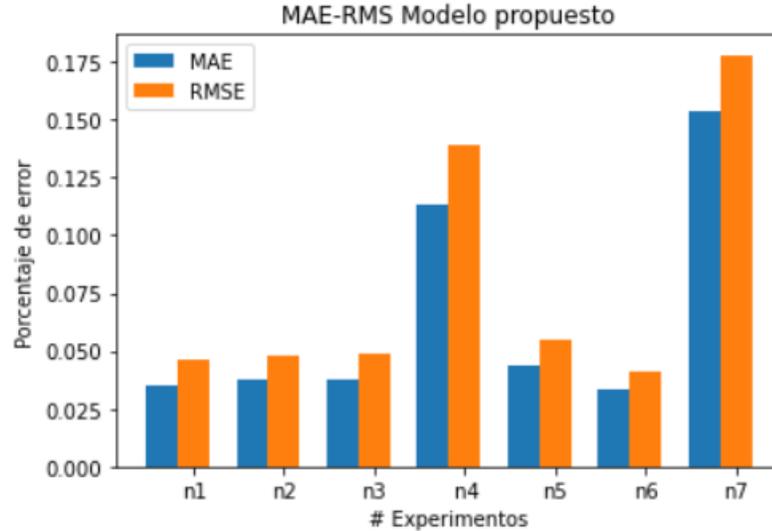


Figura 5-2: Gráfica comparativa de error MAE vs RMSE arquitectura 2

En esta arquitectura se puede apreciar que los porcentajes de error RMSE y MAE están en el rango de 0.03 y 0.17, por lo tanto no es un buen modelo para la predicción de consumo de agua ya que la evaluación de las predicciones se alejan mas de 0.

5.3. Salida Vectorial para múltiples pasos

En la tabla 5-3 se muestran todos los experimentos y resultados obtenidos con los diferentes hiperparametros en la arquitectura de la Fig. 4-3.

N Experimento	Hiperparametos probados en el modelo	MAE	RMSE
	$[input, ventana, n, odes, e, poch, b, atch, act, hid, act, ou]$		
N1	$[[3,3], 125,64,32, 500, 16, relu, sigmoid]$	0.520	0.591
N2	$[[3,3], 125,64,32, 100, 32, relu, sigmoid]$	0.260	0.290
N3	$[[2,3], 125,64,32, 100, 64, relu, sigmoid]$	0.249	0.272
N4	$[[4,3], 125,64,32, 100, 64, relu, sigmoid]$	0.183	0.269
N5	$[[4,3], 125,64,32, 100, 128, relu, sigmoid]$	0.521	0.591
N6	$[[4,3], 125,64,32, 500, 32, relu, sigmoid]$	0.285	0.312
N7	$[[2,3], 125,64,32, 500, 32, relu, sigmoid]$	0.520	0.590

Tabla 5-3: Hiperparametros probados en la arquitectura 3

En la Fig. 5-3 se muestran la comparación de evaluación de la arquitectura con el error MAE y RMSE en los diferentes experimentos realizados con los hiperparametros definidos.

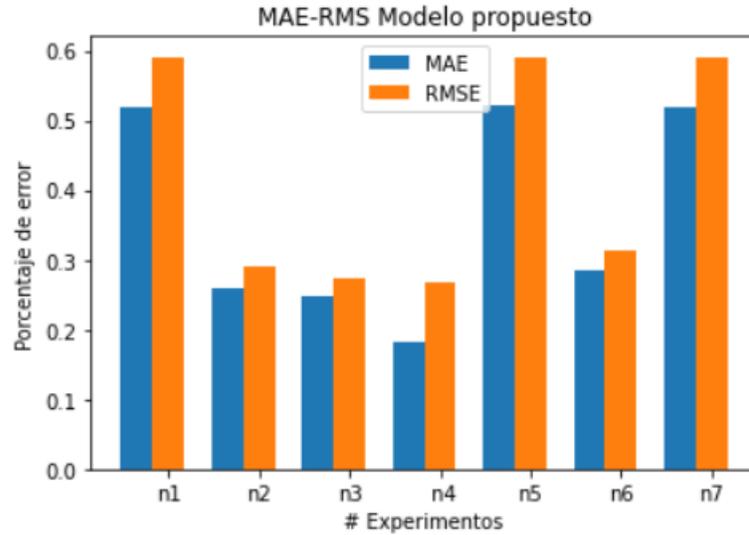


Figura 5-3: Grafica comparativa de error MAE vs RMSE arquitectura 3

En esta arquitectura se puede observar que los porcentajes de error RMSE así como también el MAE están en el rango de 0.18 y 0.59, por lo tanto no es un buen modelo para la predicción de consumo de agua, ya que mientras el error se aleja mas de 0 se considera que el modelo no se ajusta de manera óptima para la predicción.

5.4. CNN-LSTM de Múltiples cabezales para varias salidas

En la tabla 5-4 se muestran todos los experimentos y resultados obtenidos con los diferentes hiperparametros en la arquitectura de la Fig. 4-4.

N Experimento	Hiperparametros probados en el modelo	MAE	RMSE
	$[i nput, entana, n odes, e poch, b atch, act_h id, act_o u, f iltr, kernel_s ize]$		
N1	$[[2,3], 100,100, 500, 32 ,relu, linear , 2 , 2]$	0.049	0.061
N2	$[[4,3], 100,100, 50, 16 ,relu, linear , 2 , 2]$	0.063	0.079
N3	$[[3,3], 100,100, 100, 16 ,relu, linear , 2 , 2]$	0.103	0.120
N4	$[[2,3], 100,100, 100, 32 ,relu, linear , 2 , 3]$	0.111	0.131
N5	$[[2,3], 100,100, 100, 64 ,relu, linear , 1 , 2]$	0.046	0.058
N6	$[[2,3], 100,100, 100, 128 ,relu, linear , 2 , 2]$	0.097	0.126
N7	$[[2,3], 100,100, 500, 16 ,relu, linear , 2 , 2]$	0.041	0.050

Tabla 5-4: Hiperparametros probados en la arquitectura 4

En la Fig. 5-4 se muestran la comparación de evaluación de la arquitectura con el error MAE y RMSE en los diferentes experimentos realizados con los hiperparametros definidos.

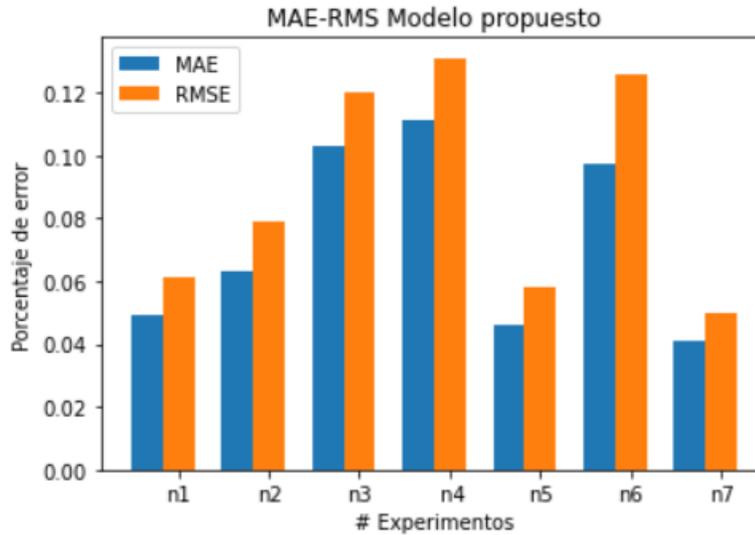


Figura 5-4: Gráfica comparativa de error MAE vs RMSE arquitectura 4

En esta arquitectura se puede apreciar que los porcentajes de error RMSE y MAE están en el rango de 0.04 y 0.13, por lo tanto no es un buen modelo para la predicción de consumo de agua, ya que mientras el error sea mas bajo, se considera que el modelo se ajusta de manera óptima para la predicción, en este caso el error del algoritmo se aleja mas en comparación de la arquitectura 1.

5.5. Selección del mejor modelo

En la Fig. 5-1 se observa que el error va desde 0.02 y se elevan hasta 0.59 en los diferentes experimentos realizados, por lo tanto en la tabla 5-5 se presentan los hiperparametros del mejor experimento número 7 de la arquitectura propuesta 1 donde la evaluación del error es la mas cercana a 0 con un MAE de 0.02, para este modelo seleccionado también se evalúa la función de perdida loss que se observa en la Fig. 5-5, esta función estima la desviación entre los valores predichos de la red neuronal y los valores reales de las observaciones utilizadas durante el entrenamiento. Mientras menor sea la desviación entre los valores reales y las predicciones, el resultado sera más eficiente en la RNN. La ultima evaluación sera la predicción de la RNN ante los valores reales dentro del conjunto de datos como se puede observar en la Fig. 5-6, obteniendo que las predicciones de la arquitectura observada en la Fig. 4-1 se ajustan a los valores reales, por lo tanto se recomienda usar este modelo para la predicción del consumo de agua por usuario y sector.

Hiperparametros mejor modelo	MAE	RMSE
$[input, entana, n, odes, e, poch, b, atch, act_h, id, act_o, u, f, iltr]$		
$[[2,3] , [125,64,32] , 500 , 34 , relu, sigmoid]$	0.029	0.038

Tabla 5-5: Hiperparametros mejor modelo

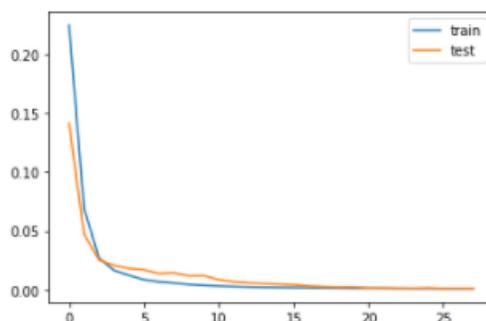


Figura 5-5: Función de perdida (loss) mejor modelo

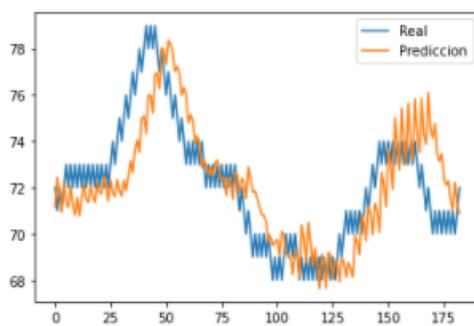


Figura 5-6: Evaluación del mejor modelo real vs predicción

5.6. Visualización de los datos

En la Fig. 5-7 se puede observar la técnica de visualización gráfica lineal, esta representa todos los consumos de agua potable de los usuarios a lo largo del tiempo, en la Fig. 5-8 se aprecia el consumo real y el pronostico de consumo para 3 meses de un usuario seleccionado, tanto con un aumento del 10 % y una disminución del 10 %, también en la Fig. 5-9 se observa un gráfico de barras describiendo el pago mensual por el consumo en 3 meses con un aumento y una reducción del 10 %, mostrándole al cliente el valor a pagar dependiendo del consumo de agua.



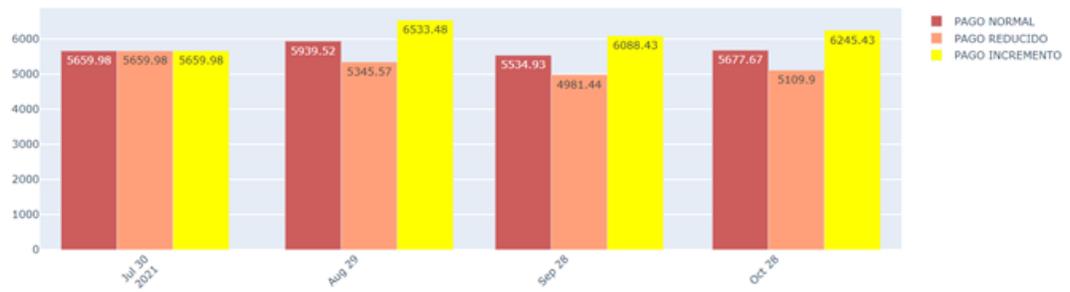
Figura 5-7: Consumo de agua potable diferentes usuarios

Reporte de consumo de agua



Plot 1 Consumo de agua potable a través del tiempo

Figura 5-8: Consumo de agua de un usuario con predicción a 3 meses con aumento y reducción del 10 %



Plot 2 Comparacion de pagos para los siguientes meses

Date	CONSUMO_NORMAL	PAGO_NORMAL	CONSUMO_REDUCIDO	PAGO_REDUCIDO	CONSUMO_INCREMENTO	PAGO_INCREMENTO
2021-07-30	1825.80	5659.98	1825.80	5659.98	1825.80	5659.98
2021-08-31	1915.98	5939.52	1724.38	5345.57	2107.57	6533.48
2021-09-30	1785.46	5534.93	1606.92	4981.44	1964.01	6088.43
2021-10-31	1831.51	5677.67	1648.36	5109.90	2014.66	6245.43

Table 1 Información consumo vs pago

Figura 5-9: Cantidad y monetización del consumo de agua de un usuario con predicción a 3 meses con aumento y reducción del 10%

6 Cronograma de actividades y presupuesto

En el presente capítulo se presenta el cronograma con las actividades completadas para el desarrollo de los algoritmos de pronóstico y los presupuestos del proyecto.

6.1. Cronograma de Actividades

En la tabla 6-5 se describe las actividades y sus respectivas fechas realizadas para el presente trabajo.

Definición del equipo

Cliente/Product Owner (PO): Dr. Jairo Sacoto, Dr. Remigio Hurtado.

Equipo:

- Desarrollador 1: Bryan Israel Chuchuca A (D1).
- Desarrollador 2: Jonnathan Daniel Sicha R (D2).

Descripción de actividades

OE1. Estudiar los fundamentos de IoT, series temporales, redes neuronales recurrentes, y técnicas de visualización

No.	Actividad
1	Estudio de los fundamentos de IoT.
2	Estudio de las series temporales.
3	Estudio de las redes neuronales recurrentes.
4	Estudio de métricas de evaluación de los algoritmos.
5	Estudio de las técnicas de visualización.
6	Realizar la documentación de la información para la creación de los algoritmos (redacción de introducción y estado del arte).

Tabla 6-1: Representa las actividades del Objetivo Específico 1

OE2. Diseñar un módulo inteligente para predicción y visualización de consumo de agua potable por cliente.

No.	Actividad
1	Recopilación de datos.
2	Preprocesamiento de los datos.
3	Análisis exploratorio y visualización de los datos.
4	Desarrollo del modelo RNN referencial para predicción por usuario.
5	Desarrollo de experimentos para selección del mejor modelo.
6	Ajuste de parámetros, optimización y evaluación de la RNN seleccionada.
7	Presentación de resultados de la RNN.
8	Reunión con el docente tutor.

Tabla 6-2: Representa las actividades del Objetivo Específico 2

OE3. Diseñar un módulo inteligente para predicción y visualización de consumo de agua potable por sector

No.	Actividad
1	Preprocesamiento de los datos.
2	Análisis exploratorio y visualización de los datos.
3	Desarrollo del modelo RNN referencial para predicción por sector.
4	Desarrollo de experimentos para selección del mejor modelo.
5	Ajuste de parámetros, optimización y evaluación de la RNN seleccionada.
6	Presentación de resultados de la RNN.
7	Reunión con el docente tutor.
8	Redacción de metodología propuesta y diseño de experimentos.

Tabla 6-3: Representa las actividades del Objetivo Específico 3

OE4. Diseñar las pruebas de validación de los algoritmos y del módulo de monitoreo.

No.	Actividad
1	Diseño de pruebas de validación de los algoritmos.
2	Análisis de errores del algoritmo.
3	Ajuste de parámetros para mejora del algoritmo.
4	Presentación de las nuevas predicciones.
5	Presentación convergencia de la red en el testing.
6	Desarrollo e implementación de reportes y gráficas de resultados.
7	Redacción de resultados, discusiones y conclusiones.
8	Reunión con el docente tutor.

Tabla 6-4: Representa las actividades del Objetivo Específico 4

Objetivo	Actividad	Semana	Fecha Inicio	Fecha Fin	Días	Horas	Responsables
OE.1	Act. 1	1	19/10/2021	20/10/2021	1	4	D1,D2
	Act. 2	1	20/10/2021	21/10/2021	1	4	D1,D2
	Act. 3	1	21/10/2021	22/10/2021	1	4	D1,D2
	Act. 4	1	22/10/2021	23/10/2021	1	4	D1,D2
	Act. 5	1	23/10/2021	24/10/2021	1	4	D1,D2
	Act. 6	2	24/10/2021	25/10/2021	1	8	D1,D2
OE. 2	Act. 1	2	26/10/2021	28/10/2021	2	18	D1,D2
	Act. 2	3	29/10/2021	4/11/2021	3	26	D1,D2
	Act. 3	4	4/11/2021	5/11/2021	1	10	D1,D2
	Act. 4	5	8/11/2021	19/11/2021	10	80	D1,D2
	Act. 5	6	22/11/2021	24/11/2021	2	32	D1,D2
	Act. 6	7	24/11/2021	26/11/2021	2	26	D1,D2
	Act. 7	7	29/11/2021	30/11/2021	1	10	D1,D2
	Act. 8	7	30/11/2021	1/12/2021	1	2	PO,D1,D2
OE. 3	Act. 1	8	1/12/2021	3/12/2021	2	24	D1,D2
	Act. 2	8	3/12/2021	3/12/2021	1	8	D1,D2
	Act. 3	8	6/12/2021	16/12/2021	10	80	D1,D2
	Act. 4	9	16/12/2021	20/12/2021	3	32	D1,D2
	Act. 5	10	20/12/2021	23/12/2021	3	24	D1,D2
	Act. 6	10	23/12/2021	23/12/2021	1	10	D1,D2
	Act. 7	10	3/12/2021	3/12/2021	1	2	PO,D1,D2
	Act. 8	11	3/1/2022	6/1/2022	3	26	D1,D2
OE. 4	Act. 1	12	6/1/2022	11/1/2022	4	24	D1,D2
	Act. 2	12	11/1/2022	12/1/2022	1	8	D1,D2
	Act. 3	12	12/1/2022	14/1/2022	2	28	D1,D2
	Act. 4	13	14/1/2022	18/1/2022	3	32	D1,D2
	Act. 5	13	18/1/2022	19/1/2022	1	9	D1,D2
	Act. 6	14	19/1/2022	21/1/2022	3	28	D1,D2
	Act. 7	15	21/1/2022	25/1/2022	3	32	D1,D2
	Act.8	15	26/1/2022	26/1/2022	1	1	PO,D1,D2

Tabla 6-5: Actividades desarrolladas para el proyecto

Total, de Horas: 600 Horas

Horas Bryan Israel Chuchuca Alvarracin 300 Horas

Horas Jonnathan Daniel Sicha Rodriguez 300 Horas

Fecha de Inicio: martes 19 de octubre de 2021

Fecha de Fin: martes 25 de enero de 2022

6.2. Presupuesto

En la tabla 6-6 se describe el presupuesto estimado para el presente trabajo.

Denominacion	Cantidad	Costo unitario	Total
	unidades	dólares	dólares
1. Bienes			
Copias	50	0,01	0,5
Impresiones	250	0,05	12,5
Empastados	2	10	20
2. Tecnológico			
Computador Portátil	2	800	1600
Celular Inteligente	2	300	600
Servidor en Nube	1	90	90
3. Servicios			
Servicio de transporte	300	1,25	375
servicios de Internet	6	35	210
Taxis	20	3	60
Alimentación	150	2,5	375
4. Personal			
Estudiante/Desarrollador	600 horas(2 estudiantes)	8 por hora	4800
Asesoría especializada	36 horas(2 asesores)	45 por hora	1620
5. Otros			
Imprevistos	1	100	100
Total		1,394, 81	9,863, 00

Tabla 6-6: Presupuesto para el proyecto

7 Conclusiones y Recomendaciones

En este capítulo presentamos las conclusiones y las recomendaciones para el diseño, desarrollo e implementación de una red neuronal LSTM para la predicción del consumo de agua potable.

7.1. Conclusiones

- Durante este estudio se desarrollaron 4 modelos con diferentes arquitecturas, capas, funciones de activación, pasos de tiempo los cuales se implementaron según las fases establecidas en la propuesta de solución que va desde la lectura de datos, análisis exploratorio, pre-procesamiento de datos y por último la evaluación de cada uno de los modelos propuestos.
- Los resultados obtenidos con el modelo de red neuronal LSTM se aproximan al consumo histórico real, y se elige como mejor modelo el modelo codificador-decodificador en base a la medida de error con RMSE de 0.04 y MAE de 0.02. El hecho de que proporciona previsiones sectoriales con la ayuda del mismo modelo que las previsiones de consumo de agua potable para múltiples usuarios.
- Así mismo, se debe tener en cuenta el tamaño del conjunto de datos ya que debe cumplir un mínimo de 60 fechas para el uso de la red neuronal LSTM, además debemos definir un diseño de experimentos con distintos modelos, parámetros, tamaño de ventana, funciones de activación y entre otros, por último, el uso de al menos dos métricas es una muy buena idea para definir el modelo más adecuado y óptimo conforme a los datos utilizados.
- A futuro se espera la optimización del modelo de red neuronal LSTM para destinarlo a aplicaciones de tipo administrativo del recurso vital de manera que las predicciones influyan directamente en la toma de decisiones ya sea en temas de construcción de infraestructura, nuevas fuentes de captación de agua y otros, además podríamos optimizar la forma de distribución de agua potable para que los usuarios tengan un servicio conforme a sus necesidades y asimismo lograr que más usuarios tengan acceso a un recurso natural indispensable como lo es el agua potable, por otro lado se espera el desarrollo de un front end amigable para el usuario donde se refleje la predicción de los valores de consumo acorde a su histórico con el fin de llegar a su conciencia y así reduzca su consumo.

7.2. Recomendaciones

- Realizar más combinaciones de experimentos con los hiperparámetros para así obtener más resultados y poder mejorar el modelo para las predicciones a futuro.
- Ejecutar el pronóstico de consumo de agua con diferentes conjuntos de datos para visualizar el comportamiento de los modelos propuestos, dando resultados diferentes a futuro ya sea el pronóstico anual o trimestral.
- Analizar más modelos que se ajusten al pronóstico de series de tiempo multivariantes de múltiples pasos para compararlos con los modelos propuestos y obtener un modelo con menor error RMSE y MAE.

Bibliografía

- [1] Estructura redes neuronales artificiales, 2019.
- [2] Redes neuronales recurrentes, 2019.
- [3] Recurrent neural networks and lstm, 2019.
- [4] Arquitectura encoder-decoderlstm, 2020.
- [5] An ensemble approach for multi-step ahead energy forecasting of household communities, 2019.
- [6] Tipos-de-graficos-y-diagramas-para-la-visualizacion-de-datos, 2015.
- [7] ecuadorencifras.gob.ec, 2012.
- [8] Carlos Fernández Jáuregui and A Crespo Milliet. El agua, recurso único. *El derecho humano al agua: situación actual y retos del futuro*. *Icaria. Pág*, pages 19–38, 2008.
- [9] Ruth Marina Agudelo. El agua, recurso estratégico del siglo xxi: strategic resource in the 21st century. *Revista Facultad Nacional de Salud Pública*, 23(1):91–102, 2005.
- [10] Cruz García Lirios, Javier Carreón Guillén, Jorge Hernández Valdés, María Montero López Lena, and José Marcos Bustos Aguayo. Actitudes, consumo de agua y sistema de tarifas del servicio de abastecimiento de agua potable. *Polis. Revista Latinoamericana*, (34), 2013.
- [11] Félix Pérez Pérez and José Félix Pérez Gutiérrez. Agua y medio ambiente. *Observatorio Medioambiental*, 9:9, 2006.
- [12] Heinrich-Martin Schreyer. Infundir nueva vida: La tecnología del abastecimiento de agua se emplea para enfrentarse con el problema crucial a escala mundial de la disminución de las fuentes de agua potable. *Revista ABB*, (2):19–24, 2008.
- [13] Mynor Romero. Tratamientos utilizados en potabilización de agua. *Boletín Electrónico [Internet]. [citado 2012 jun 16]*, 8:1–12, 2008.
- [14] Eduardo Molina, Felipe Quesada, Andrea Calle, Jessica Ortiz, and Diana Orellana. Consumo sustentable de agua en viviendas de la ciudad de cuenca. *Ingenius. Revista de Ciencia y Tecnología*, (20):28–37, 2018.

-
- [15] María Teresa Baquero. Ahorro de agua y reutilización en la edificación en la ciudad de Cuenca, Ecuador. *Estoa. Revista de la Facultad de Arquitectura y Urbanismo de la Universidad de Cuenca*, 2(3):71–81, oct. 2013.
- [16] Guy Howard, Jamie Bartram, Ashley Williams, Alycia Overbo, Jo-Anne Geere, World Health Organization, et al. Domestic water quantity, service level and health. 2020.
- [17] María Auxiliadora Mora Carrión. Análisis de la variación estacional del consumo residencial de agua potable de la ciudad de Cuenca. B.S. thesis, Universidad del Azuay, 2019.
- [18] MIDUVI. Norma hidrosanitaria nhe agua. ministerio de desarrollo urbano y vivienda. <https://www.habitatyvivienda.gob.ec/>, 2011.
- [19] Youru Li, Zhenfeng Zhu, Deqiang Kong, Hua Han, and Yao Zhao. Ea-lstm: Evolutionary attention-based lstm for time series prediction. *Knowledge-Based Systems*, 181:104785, 2019.
- [20] Marcelo Alcaraz. Internet de las cosas. *Universidad Católica Nuestra Señora de la Asunción*, (1):2–3, 2014.
- [21] Jordi Salazar and Santiago Silvestre. Internet de las cosas. *Techpedia. České vysoké učení technické v Praze Fakulta elektrotechnická*, 2016.
- [22] Juan Manuel Cueva Lovelle, Jose Ignacio Rodriguez Molano, Carlos Enrique Montenegro Marin, et al. Introducción al internet de las cosas. *Redes de Ingeniería*, 6, 2015.
- [23] Cesar Emmanuel Delgado Tapia and William Geovanny Valencia Astudillo. Diseño e implementación de prototipo iot para el monitoreo remoto de la calidad del agua para la crianza de tilapias en estanques. B.S. thesis, 2021.
- [24] I Made Wirawan, Irawan Dwi Wahyono, Gilang Idfi, and Gradiyanto Radityo Kusumo. Iot communication system using publish-subscribe. In *2018 International Seminar on Application for Technology of Information and Communication*, pages 61–65, 2018.
- [25] Erwin J Sacoto-Cabrera, Gabriel León-Paredes, and Walter Verdugo-Romero. Lorawan: Application of nonlinear optimization to base stations location. In *Communication, Smart Technologies and Innovation for Society*, pages 515–524. Springer, 2022.
- [26] Juan Aranda, Erwin J Sacoto Cabrera, Daniel Haro Mendoza, and Fabián Astudillo Salinas. 5g networks: A review from the perspectives of architecture, business models, cybersecurity, and research developments. *Novasinerгия*, 4, 2021.

- [27] Erwin J Sacoto Cabrera, Sonia Palaguachi, Gabriel A León-Paredes, Pablo L Gallegos-Segovia, and Omar G Bravo-Quezada. Industrial communication based on mqtt and modbus communication applied in a meteorological network. In *The International Conference on Advances in Emerging Trends and Technologies*, pages 29–41. Springer, 2020.
- [28] Erwin Sacoto Cabrera. *Análisis basado en teoría de juegos de modelos de negocio de operadores móviles virtuales en redes 4G y 5G*. PhD thesis, Universitat Politècnica de València, 2021.
- [29] James Manyika, Michael Chui, Peter Bisson, Jonathan Woetzel, Richard Dobbs, Jacques Bughin, and Dan Aharon. Unlocking the potential of the internet of things. *McKinsey Global Institute*, 1, 2015.
- [30] ORGANISMOS DE LA UNIÓN EUROPEA. Comisión europea. *Comunicación de la Comisión sobre el recurso al principio de precaución COM/2000/0001 final*, 2, 2016.
- [31] Bröring Arne, Zappa Achille, Vermesan Ovidiu, Främling Kary, Zaslavsky Arkady, Gonzalez-Usach Regel, Szmeja Pawel, Carlos E Palau, Jacoby Michael, Ivana Podnar Zarko, et al. Advancing iot platforms interoperability. 2018.
- [32] Cuzme Rodríguez, Fabian Geovanny, et al. El internet de las cosas y las consideraciones de seguridad. Master’s thesis, Quito/PUCE/2015, 2018.
- [33] Maciej Kranz. *Internet of things*. Editorial Almuzara, 2017.
- [34] Erwin Jairo Sacoto Cabrera, Luis Guijarro, and Patrick Maillé. Game theoretical analysis of a multi-mno mvno business model in 5g networks. *Electronics*, 9(6):933, 2020.
- [35] V Vimos, E Sacoto, and DX Morales. Conceptual architecture definition: Implementation of a network sensor using arduino devices and multiplatform applications through opc ua. In *2016 IEEE International Conference on Automatica (ICA-ACCA)*, pages 1–5. IEEE, 2016.
- [36] Martin Molina et al. *Métodos de resolución de problemas: Aplicación al diseño de sistemas inteligentes*. Martin Molina, 2006.
- [37] Laura Cristina Lanzarini, Waldo Hasperue, Leonardo César Corbalán, Javier López, César Armando Estrebou, Franco Ronchetti, Juan Andrés Maulini, and Augusto Villa Monte. Sistemas inteligentes. In *XIII Workshop de Investigadores en Ciencias de la Computación*, 2011.
- [38] José del Sagrado. Aspectos básicos de los sistemas inteligentes.
- [39] Lasse Rouhiainen. Inteligencia artificial. *Madrid: Alienta Editorial*, 2018.

-
- [40] François Chollet and Joseph J Allaire. Deep learning with r. shelter island. *Manning Publications Co. Biometrics*, 76:361–362, 2018.
- [41] Earl B Hunt. *Artificial intelligence*. Academic Press, 2014.
- [42] Brian Romero Rojas. Una introducción a los modelos de machine learning. B.S. thesis, 2020.
- [43] Giuseppe Bonaccorso. *Machine learning algorithms*. Packt Publishing Ltd, 2017.
- [44] Masashi Sugiyama. *Statistical reinforcement learning: modern machine learning approaches*. CRC Press, 2015.
- [45] Raquel Flórez López and José Miguel Fernández Fernández. *Las redes neuronales artificiales*. Netbiblo, 2008.
- [46] Maximilian Alber, Sebastian Lapuschkin, Philipp Seegerer, Miriam Hägele, Kristof T Schütt, Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Sven Dähne, and Pieter-Jan Kindermans. innvestigate neural networks! *J. Mach. Learn. Res.*, 20(93):1–8, 2019.
- [47] Mauricio Arichávala, Cristian Jérez, et al. Determinación de la presión máxima de compresión de un motor de encendido provocado basado en una red neuronal artificial recurrente. *Ingenius. Revista de Ciencia y Tecnología*, (19):9–18, 2018.
- [48] Larry R Medsker and LC Jain. Recurrent neural networks. *Design and Applications*, 5:64–67, 2001.
- [49] Olmar Canchano, Luis Coronado, Paola Sánchez Sanchez, et al. Redes neuronales para pronóstico de series de tiempo hidrológicas del caribe colombiano. *Investigación y desarrollo en TIC*, 10(2):18–31, 2019.
- [50] Leonardo Villalobos-Arias, Christian Quesada-López, Alexandra Martínez, and Marcelo Jenkins. Técnicas de ajuste de hiperparámetros de algoritmos de aprendizaje automático para la estimación de esfuerzo: un mapeo de literatura. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (E42):305–318, 2021.
- [51] Arjun Singh Saud and Subarna Shakya. Analysis of look back period for stock price prediction with rnn variants: A case study on banking sector of nepse. *Procedia Computer Science*, 167:788–798, 2020.
- [52] Sergio Andrés Álvarez Pedreros, Héctor Allende Cid, and Joaquim B Bento. Deep learning y series de tiempo oceanográficas.
- [53] Jian Cao, Zhi Li, and Jian Li. Financial time series forecasting model based on ceemdan and lstm. *Physica A: Statistical Mechanics and its Applications*, 519:127–139, 2019.

- [54] Shengdong Du, Tianrui Li, Yan Yang, and Shi-Jinn Horng. Multivariate time series forecasting via attention-based encoder–decoder framework. *Neurocomputing*, 388:269–279, 2020.
- [55] Beatriz López Porrero. *Limpieza de datos: reemplazo de valores ausentes y estandarización*. PhD thesis, Universidad Central “Marta Abreu” de Las Villas. Facultad de Matemática . . . , 2011.
- [56] Andrés Regal, Juandiego Morzán, Carlos Fabbri, Gonzalo Herrera, Gabriela Yaulli, Andrea Palomino, and Claudia Gil. Proyección del precio de criptomonedas basado en tweets empleando lstm. *Ingeniare. Revista chilena de ingeniería*, 27(4):696–706, 2019.
- [57] Yuting Wu, Mei Yuan, Shaopeng Dong, Li Lin, and Yingqi Liu. Remaining useful life estimation of engineered systems using vanilla lstm neural networks. *Neurocomputing*, 275:167–179, 2018.
- [58] David García Ballester. *Predicción del precio de billetes de avión a partir de una red neuronal caracterizada por unidades LSTM*. PhD thesis, Universitat Politècnica de València, 2020.
- [59] Jason Brownlee. *Deep learning for time series forecasting: predict the future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery, 2018.
- [60] Ioannis E Livieris, Emmanuel Pintelas, and Panagiotis Pintelas. A cnn–lstm model for gold price time-series forecasting. *Neural computing and applications*, 32(23):17351–17360, 2020.
- [61] Marcos Daniel Guerrero Alvarado. Pronóstico del consumo energético en edificaciones a través de series multivariadas usando modelos de redes neuronales convolucionales. B.S. thesis, Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas . . . , 2021.
- [62] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. Lstm-based encoder-decoder for multi-sensor anomaly detection. *arXiv preprint arXiv:1607.00148*, 2016.
- [63] MC Acosta-Cervantes, MG Villarreal-Marroquín, and M Cabrera-Ríos. Estudio de validación de un método para seleccionar técnicas de pronóstico de series de tiempo mediante redes neuronales artificiales. *Ingeniería, investigación y tecnología*, 14(1):53–63, 2013.
- [64] John Villavicencio. Introducción a series de tiempo. *Puerto Rico*, 2010.
- [65] F Alejandro Nava. *Procesamiento de series de tiempo*. Fondo de Cultura Económica, 2015.

-
- [66] Cort J Willmott and Kenji Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate research*, 30(1):79–82, 2005.
- [67] MOISES ISRAEL HERRERA RAMOS et al. Análisis de desempeño de redes neuronales evolutivas para el pronóstico de series de tiempo. 2019.
- [68] Francisco Herrera, José Riquelme, and Roberto Ruiz. Preprocesamiento de datos. *presentado en Reunión Red Nacional DM & ML*, 2004.
- [69] Matthew Sadiku, Adebowale E Shadare, Sarhan M Musa, Cajetan M Akujuobi, and Roy Perry. Data visualization. *International Journal of Engineering Research And Advanced Technology (IJERAT)*, 2(12):11–16, 2016.
- [70] Chandrajit Bajaj and C Fl John Wiley. Data visualization techniques. 1998.
- [71] Jordi Salas-Salvadó, Francisco Maraver, Leocadio Rodríguez-Mañas, Miguel Sáenz de Pipaon, Isidro Vitoria, and Luis A Moreno. Importancia del consumo de agua en la salud y la prevención de la enfermedad: situación actual. *Nutrición Hospitalaria*, 37(5):1072–1086, 2020.
- [72] Plantas de potabilización etapa ep, 2016.
- [73] Deibys Gildardo Manco Silva, Jhoniers Guerrero Erazo, and Ana María Ocampo Cruz. Eficiencia en el consumo de agua de uso residencial. *Revista Ingenierías Universidad de Medellín*, 11(21):23–38, 2012.
- [74] Francisco Herrera. Big data: Preprocesamiento y calidad de datos. *novática*, 237:17, 2016.
- [75] Daniel Rubio Adeva. Red neuronal: análisis de series temporales de entidades bancarias. 2021.