ISSN: 1692-7257 - Volumen 1 – Número 47 - 2026

RCTA
Revista Colombiana de Tecnologías de Avanzada
UNIPAMPLONA

Digital Object Identifier: 10.24054/rcta.v1i47.4103

Arquitectura de oráculo blockchain con validación de datos basada en procesos de aprendizaje de maquina

Blockchain oracle architecture with data validation based on machine learning processes

PhD. Cristian Camilo Ordoñez Quintero[1], PhD. Hugo Armando Ordoñez Erazo[2], MSc. Juan Sebastián González Sanabria[3]

[1] ***Universidad de Nariño***, *Grupo de Investigación GREDIS, Pasto, Nariño-Colombia.*
[2] ***Universidad del Cauca***, *Grupo de Investigación GTI, Popayán, Cauca-Colombia*
[2] ***Universidad Pedagógica y Tecnológica de Colombia***, *Grupo de Investigación GIMI, Tunja, Boyacá-Colombia*

Correspondencia: *ccordonez@udenar.edu.co*

Recibido: *25 julio 2025.* ***Aceptado:*** *20 diciembre 2025.* ***Publicado:*** *01 enero 2026.*

Cómo citar: *C. C. Ordoñez Quintero, H. A. Ordoñez Erazo y J. S. González Sanabria, "Arquitectura de oráculo blockchain con validación de datos basada en procesos de aprendizaje de maquina", RCTA, vol. 1, n.º. 47, pp. 72-82, ene. 2026. Recuperado de https://ojs.unipamplona.edu.co/index.php/rcta/article/view/4103*

Resumen: La creciente dependencia de contratos inteligentes en ecosistemas blockchain ha puesto en evidencia la necesidad de contar con mecanismos confiables para validar los datos que estos consumen desde fuentes externas, comúnmente gestionadas por oráculos. Este artículo presenta una arquitectura de oráculo blockchain que incorpora un sistema de validación de datos basado en técnicas de Machine Learning, con el objetivo de fortalecer la calidad de los datos antes de su incorporación a la cadena de bloques. La solución integra modelos supervisados y no supervisados para la detección de anomalías, clasificación multietiqueta, análisis de series temporales y detección de valores atípicos. El proceso de validación sigue el enfoque CRISP-DM y se complementa con un indicador de integridad de datos basado en estadísticas descriptivas y mecanismos de votación por mayoría (hard voting), que permite estimar automáticamente la aceptabilidad de los datos. Los resultados experimentales, obtenidos en un entorno de pruebas sobre contratos inteligentes funcionales, demuestran mejoras en la detección de inconsistencias y manipulación de datos, así como en la confiabilidad de las decisiones automatizadas. Esta propuesta aporta una estrategia sistemática y replicable para mitigar riesgos asociados al consumo de datos en aplicaciones blockchain.

Palabras clave: blockchain, aprendizaje de máquina, tratamiento de datos, toma de decisiones.

Abstract: The growing reliance on smart contracts in blockchain ecosystems has highlighted the need for reliable mechanisms to validate the data these contracts consume from external sources, commonly managed by oracles. This article presents a blockchain oracle architecture that incorporates a data validation system based on machine learning techniques, with the goal of enhancing data quality before it is incorporated into the blockchain. The solution integrates supervised and unsupervised models for anomaly detection, multi-label classification, time series analysis, and outlier detection. The

validation process follows the CRISP-DM methodology and is complemented by a data integrity indicator based on descriptive statistics and majority voting mechanisms (hard voting), which allows for the automatic estimation of data acceptability. Experimental results, obtained in a testing environment using functional smart contracts, demonstrate improvements in the detection of inconsistencies and data manipulation, as well as in the reliability of automated decisions. This proposal provides a systematic and replicable strategy to mitigate risks associated with data consumption in blockchain applications.

Keywords: blockchain, machine learning, data processing, decision-making.

1. INTRODUCCIÓN

La irrupción de la tecnología blockchain ha transformado profundamente la manera en que se conciben y ejecutan los sistemas de información distribuidos [1]. A diferencia de las arquitecturas tradicionales centralizadas, blockchain propone un modelo descentralizado en el que múltiples nodos participan de manera colaborativa para registrar, verificar y almacenar transacciones de forma inmutable [2]. Esta propiedad ha dado lugar al desarrollo de contratos inteligentes, programas de computador que se ejecutan de manera automática al cumplirse ciertas condiciones previamente establecidas [3]. Estos contratos han cobrado relevancia en un amplio espectro de aplicaciones, desde sistemas financieros descentralizados (DeFi) hasta cadenas de suministro, sistemas notariales, identidad digital y gobernanza distribuida [4].

No obstante, a pesar de sus beneficios estructurales en cuanto a transparencia, descentralización y auditabilidad, los contratos inteligentes presentan una limitación fundamental: su incapacidad para acceder directamente a datos del mundo exterior a la blockchain [5]. Esto se debe a que la cadena de bloques está diseñada como un entorno cerrado y determinista, en el que todas las operaciones deben ser reproducibles de manera idéntica por todos los nodos. En consecuencia, los contratos inteligentes dependen de intermediarios denominados oráculos, encargados de suministrar la información externa necesaria para su ejecución [5]. Esta información puede incluir precios de activos, resultados de eventos, condiciones climáticas, o cualquier otro dato que no se origine directamente dentro de la red blockchain.

Los oráculos representan, por tanto, un componente crítico dentro del ecosistema blockchain, al actuar como puentes entre el mundo off-chain y on-chain [6]. Sin embargo, esta misma función los convierte en uno de los eslabones más vulnerables del sistema. A diferencia de la cadena de bloques, que es inmutable y resistente a manipulaciones gracias a mecanismos de consenso como Proof of Work o Proof of Stake, los oráculos introducen un punto de confianza que puede ser comprometido [5]. Un oráculo que suministre datos incorrectos, manipulados o incompletos puede desencadenar la ejecución errónea de contratos inteligentes, con consecuencias potencialmente catastróficas en entornos de alta criticidad [5].

Este fenómeno, conocido como el "problema del oráculo", ha sido identificado como una de las principales barreras para la adopción masiva de soluciones blockchain en contextos donde la integridad de los datos es fundamental [6], [7]. A pesar de los avances en técnicas de descentralización de oráculos, como Chainlink o Band Protocol, persisten desafíos técnicos en torno a la verificabilidad, la transparencia y la calidad de los datos proporcionados [8]. En muchos casos, los sistemas actuales carecen de mecanismos formales para validar la consistencia, exactitud o confiabilidad de los datos externos, delegando esta responsabilidad en arquitecturas externas poco auditables o centralizadas.

Frente a este panorama, surge la necesidad de desarrollar nuevos enfoques que permitan incorporar validaciones explícitas y replicables sobre los datos que ingresan a la blockchain a través de los oráculos [6]. Particularmente, el campo de la Ciencia de Datos y el Aprendizaje Automático (Machine Learning) ofrece herramientas potentes para el análisis de datos, la detección de patrones anómalos y la predicción basada en comportamiento histórico [9].

Así las cosas, este artículo propone una arquitectura de oráculo blockchain con un sistema de validación de datos basado en procesos de Machine Learning. La propuesta tiene como objetivo mejorar la integridad de los datos externos que se utilizan en contratos inteligentes. Para lograrlo, se implementa una cadena de procesamiento que incluye modelos supervisados y no supervisados, orientados a la detección de anomalías en los datos enviados por

oráculo. Este flujo de validación está estructurado según las fases del modelo CRISP-DM (Cross-Industry Standard Process for Data Mining), lo que garantiza una metodología sistemática, replicable y alineada con buenas prácticas de ingeniería de datos [10].

La arquitectura incluye además un componente de decisión basado en un mecanismo de votación por mayoría (hard voting) [11]. Este enfoque permite no solo rechazar datos anómalos o inconsistentes, sino también establecer umbrales objetivos de aceptabilidad que puedan ser auditados y trazados en tiempo real. En términos funcionales, la solución se implementa como una capa intermedia entre la fuente de datos externa y el contrato inteligente desplegado en la blockchain, actuando como un filtro inteligente que refuerza la seguridad y la confiabilidad del sistema.

Este artículo se estructura por medio de las siguientes secciones. En la Sección 2, se presentan los conceptos clave necesarios para contextualizar el problema de investigación, así como una revisión breve pero sustancial del estado del arte, enfocada en oráculos, contratos inteligentes y validación de datos en entornos blockchain. Posteriormente, la Sección 3 expone los métodos utilizados para el desarrollo de la investigación. Allí se detalla el enfoque técnico adoptado, el flujo de procesamiento de datos y la arquitectura propuesta. En la Sección 4, se describe la implementación de la propuesta y se presentan los resultados obtenidos tras su implementación. Finalmente, la Sección 5 recoge las conclusiones del trabajo y plantea posibles líneas de investigación futura.

2. TRABAJOS RELACIONADOS

Para identificar la literatura más relevante de los últimos cinco años, se realizó una búsqueda sistemática en las bases de datos Scopus y Web of Science (WOS), utilizando cadenas específicas relacionadas con arquitecturas de oráculos en blockchain y la aplicación de técnicas de machine learning. A partir de esta revisión, se seleccionaron los estudios más significativos que abordan la convergencia entre los oráculos blockchain y los métodos de machine learning (ML), evidenciando cómo esta integración ha dado lugar a arquitecturas avanzadas orientadas a garantizar la integridad, fiabilidad y trazabilidad de los datos off-chain antes de su incorporación en contratos inteligentes.

Para iniciar los autores en [12] crean un enfoque relevante es el modelo TCO-DRL (Trust-Aware and Cost-Optimized Reinforcement Learning), que utiliza técnicas de reinforcement Learning profundo para seleccionar dinámicamente nodos oráculo basándose en múltiples dimensiones de reputación y coste. Este modelo logra reducir las asignaciones a nodos maliciosos y optimizar costos sin comprometer la calidad de los datos.

Por otro lado, la plataforma DeepThought, propuesta en 2022, incorpora un mecanismo híbrido de votación verificada y reputación, donde usuarios colaboran en validar información y se combinan resultados mediante un esquema de reputación distribuida que mejora la resistencia a corrupciones humanas [13].

De igual manera se consideran estudios como el de blockchain para IIoT, que presentan un esquema de selección de nodos oráculos basado en Verifiable Random Functions (VRF) y reputación, junto con un algoritmo de filtrado de datos en ventana deslizante (sliding window). Este enfoque mejora la calidad de servicio, reduce variabilidad y aumenta la precisión en entornos con nodos maliciosos [14].

Dentro del dominio ML aplicado a blockchain, un mapeo sistemático reciente analiza 159 artículos relacionados con anomalías, clasificación y uso de datos on-chain, identificando tendencia creciente en detección de anomalías, retos de estandarización y escalabilidad, así como la escasa interoperabilidad entre cadenas [14].

Autores como [15] han identificado que gran parte de los trabajos se centran en detección de anomalías o patrones atípicos en los datos de blockchain, usando técnicas supervisadas y no supervisadas, lo que resulta esencial para la validación de datos antes de su uso dentro de un sistema de oráculo.

La literatura también destaca el uso de técnicas ML para detectar comportamiento adversarial o manipulaciones maliciosas, como ataques flash loan o Sybil, mediante clustering, autoencoders, filtrado estatístico y redes neuronales en grafos; también se han estudiado agentes RL para ajustar dinámicamente puntuación de confianza y rechazo de datos erróneos [16].

Además, marcos como Oraichain integran oráculos AI-centric que ejecutan AI off-chain y validan los modelos mediante pruebas de ejecución verificables; inspiran nuevas capas de confianza para alimentar contratos inteligentes con datos procesados por ML previamente verificados [7].

Por último, revisiones críticas de la literatura subrayan desafíos inherentes: latencia en obtención de datos, calidad variable de fuentes, problemas de interoperabilidad entre protocolos y cadenas, escalabilidad de los modelos ML distribuidos, y la necesidad de técnicas de privacy-preserving ML como aprendizaje federado o SGD con privacidad diferencial en entornos permissioned [17].

3. PROPUESTA

La arquitectura propuesta de oráculo blockchain con validación de datos basada en técnicas de machine learning se fundamenta en una revisión exhaustiva de la literatura reciente, que respalda la pertinencia y factibilidad de cada uno de sus componentes, en la Figura 1 se observa cada uno de los componentes.

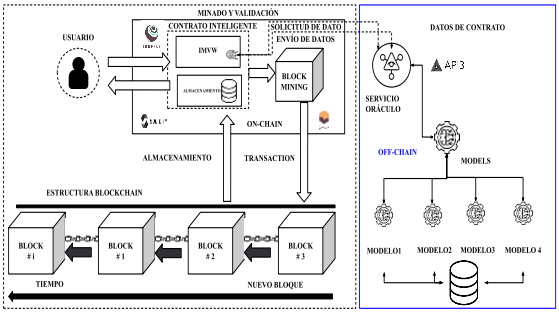

Fig. 1 *Arquitectura propuesta.* ***Fuente:*** *elaboración propia.*

En primera instancia, la implementación del contrato inteligente se desarrolla en Solidity y se prueba mediante herramientas como Truffle y Ganache, ampliamente reconocidas en la literatura como entornos robustos para el diseño, compilación, despliegue y verificación local de contratos en blockchain [18]. Esta capa permite la automatización de acuerdos mediante condiciones programadas que requieren insumos externos para su ejecución.

La adquisición de datos externos se realiza a través de oráculos descentralizados como API3, los cuales ofrecen una interfaz segura y verificable entre el mundo físico y la cadena de bloques. Estos oráculos son operados directamente por proveedores de datos, lo que reduce la posibilidad de manipulación intermedia y mejora la trazabilidad de la fuente [19].

La propuesta incorpora un módulo de validación basado en un esquema de votación dura ponderada multinivel (IMWV), que recibe las predicciones de múltiples modelos de clasificación y produce una decisión consolidada con pesos asignados según el rendimiento histórico de cada modelo. Esta técnica ha demostrado ser altamente efectiva para mejorar la precisión en tareas de clasificación multietiqueta y detección de anomalías en datos heterogéneos [20].

Una vez validados, los datos se integran en la cadena mediante un proceso de minería simulado localmente. Esta inclusión garantiza la integridad criptográfica, trazabilidad y resistencia a manipulaciones, aspectos esenciales en el contexto de contratos inteligentes vinculados a variables externas.

Adicionalmente, la arquitectura contempla un mecanismo de aprendizaje continuo, mediante el cual los modelos de machine learning se actualizan periódicamente a partir de los datos validados e incorporados en la cadena. Esta retroalimentación permite adaptar los modelos a nuevas condiciones y mantener un alto nivel de desempeño predictivo, en línea con las tendencias actuales en online learning y aprendizaje federado distribuido.

4. IMPLEMENTACIÓN

La construcción de la propuesta se llevó a cabo siguiendo la metodología CRISP-DM (Cross Industry Standard Process for Data Mining), adaptada al contexto de validación de datos en oráculos blockchain. A continuación, se detallan las fases desarrolladas como se guía en la figura 2:

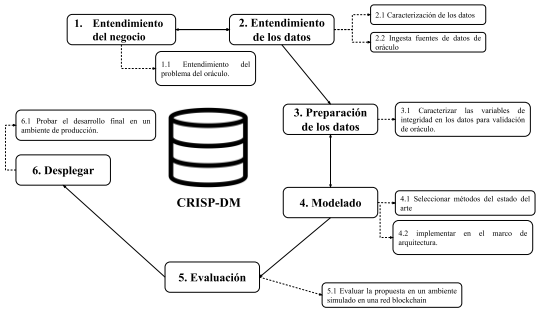

Fig. 2 *Implementación de arquitectura y modelos ml para la validación de datos de oráculo.* ***Fuente:*** *elaboración propia*

4.1 Entendimiento del negocio

En esta primera fase se abordó el entendimiento del problema del oráculo, centrado en su rol crítico dentro de la ejecución automática de contratos inteligentes que dependen de información proveniente del mundo real (off-chain). Se identificó que la principal problemática reside en la falta de mecanismos efectivos para validar la veracidad, consistencia y completitud de los datos proporcionados por los oráculos, lo que puede

derivar en decisiones contractuales erróneas o manipuladas.

4.2 Datos

En este estudio se centra en tomar un contrato inteligente altamente volátil, ejemplo de ello los contratos futuros de café, las variables seleccionadas temperatura, humedad relativa, altitud, precipitaciones, variedad y precio de café fueron incluidas debido a su relevancia directa en la validación de datos externos utilizados por contratos inteligentes en dominios agroclimáticos de los contratos futuros [21]. Estas variables corresponden a fuentes reales que comúnmente alimentan los oráculos de información meteorológica y agrícola, por lo que su papel dentro de la arquitectura se fundamenta en la necesidad de garantizar coherencia, correlación interna y consistencia temporal antes de permitir su uso dentro del contrato inteligente.

Cada variable cumple un propósito específico dentro del proceso de validación:

- Temperatura, humedad y precipitaciones permiten detectar anomalías meteorológicas abruptas o valores imposibles.
- Altitud actúa como variable de control espacial en el análisis climático.
- Variedad (categoría agrícola) permite validar consistencia entre el tipo de cultivo y su respuesta climática.

Precio del café se incorpora como variable que puede ser manipulada en oráculos financieros, siendo crítica para evitar ejecuciones erróneas de contratos futuros.

Posteriormente, se procedió a la caracterización de los datos disponibles (2.1), provenientes de sensores ambientales, estaciones meteorológicas, APIs comerciales y registros históricos agrícolas. Además, se desarrolló un mecanismo de ingesta de datos desde fuentes externas a través de oráculos (2.2), conectando con servicios como API3 para proveer datos de manera descentralizada y verificable.

4.3 Preparación de los datos

Una vez comprendida la naturaleza de las fuentes, se inició la fase de preparación de los datos (3.1), en la cual se diseñaron criterios para identificar atributos de integridad de los datos, como consistencia interna, puntualidad, completitud y fiabilidad de origen.

Para construir el conjunto de entrenamiento de los modelos, se implementó un proceso de etiquetado binario que clasifica cada evento como válido o inválido. La etiqueta se asigna considerando tres criterios:

- Consistencia interna: coherencia entre variables meteorológicas y agrícolas.
- Coherencia temporal: comparación directa con el comportamiento histórico registrado mediante técnicas de series temporales.
- Verificación por umbrales: rangos provistos por fuentes oficiales o datos de referencia externos (API3 y estaciones climáticas).

Estos atributos fueron utilizados como insumos para entrenar modelos que validan la calidad de los datos antes de su incorporación en la cadena de bloques.

Tabla 1: *Etiquetado de datos oráculo*

Altitud	Temperatura	Humedad	Precisión	Precisión por altitud	Consistencia por temperatura	Consistencia por riego	Completitud	Confiabilidad
1424.7	19.57	73.58	1	1	1	1	1	1
1770.4	19.48	77.08	1	1	1	1	1	1
1639.2	23.44	76.40	1	1	0	1	1	1
1559.2	19.50	67.31	1	1	0	1	0	1
1293.6	19.63	67.24	1	1	0	0	1	1

4.4 Modelado

Durante la fase de modelado, se procedió a seleccionar métodos del estado del arte en aprendizaje automático (4.1), los modelos empleados Random Forest, Support Vector Machines (SVM), K-Nearest Neighbors (KNN) y Extra Trees fueron escogidos debido a su eficacia reportada en tareas de clasificación multietiqueta, detección de anomalías y validación de datos heterogéneos. Cada algoritmo fue entrenado sobre un conjunto de datos previamente depurado y etiquetado siguiendo un procedimiento binario que distingue entre eventos válidos e inválidos, basándose en criterios de consistencia interna, coherencia temporal y verificación por umbrales técnicos.

Para asegurar estabilidad y comparabilidad entre los modelos, el dataset se dividió mediante una partición estratificada del 70% para entrenamiento,

15% para validación y 15% para prueba, manteniendo la proporción entre clases. El ajuste de hiperparámetros se llevó a cabo mediante Grid Search [22], optimizando métricas como la precisión, el recall y el puntaje F1 para cada algoritmo. Las mejores configuraciones encontradas se utilizaron posteriormente en el proceso de agregación.

La integración de los modelos dentro del sistema se realizó a través del esquema IMWV (Incremental Multi-Level Weighted Hard Voting). En este mecanismo, cada modelo genera su predicción individual y aporta un voto ponderado según su desempeño histórico en las métricas seleccionadas. El sistema consolida estas predicciones mediante una votación dura que considera tanto la decisión binaria como el peso asignado, permitiendo obtener una evaluación final más estable y representativa que la producida por cualquier modelo de manera independiente. Esta arquitectura, además, facilita la detección temprana de inconsistencias, ya que cualquier discordancia significativa entre los modelos puede ser utilizada como señal adicional de anomalía además implementados dentro de la arquitectura del oráculo

4.5 Evaluación

La arquitectura resultante fue sometida a una evaluación experimental dentro de un entorno simulado con tecnología blockchain (5.1).

La arquitectura experimental fue definida y documentada previamente a la fase de resultados, con el fin de garantizar la trazabilidad y reproducibilidad completa del estudio. Para la simulación del entorno blockchain se empleó Ganache, permitiendo la creación de nodos locales controlados y replicables. El proceso de compilación, despliegue y verificación de contratos inteligentes se realizó mediante Truffle, mientras que la interacción programática con la red se llevó a cabo utilizando Web3.py.

Las pruebas se ejecutaron en un sistema operativo Windows 11 Pro, sobre un equipo ASUS TUF F15 equipado con un procesador Intel Core i5 de 12ª generación, 8 GB de memoria RAM DDR4 y una unidad de almacenamiento SSD NVMe. El módulo de validación basado en aprendizaje automático fue implementado en Python 3.X, utilizando librerías especializadas como Scikit-learn, Pandas, NumPy y Matplotlib para el procesamiento, entrenamiento, evaluación y visualización de los modelos.

Toda esta configuración fue integrada explícitamente en la sección metodológica con el objetivo de proporcionar un marco de referencia claro que permita replicar el experimento bajo las mismas condiciones técnicas.

Para evaluar la efectividad de la arquitectura propuesta, se desarrolló un experimento compuesto por 60 pruebas secuenciales, cada una representando un ciclo de validación completo. Cada ciclo incluyó las siguientes etapas:

- Recepción del dato externo proveniente del oráculo simulado.
- Validación mediante el sistema ML + IMWV, donde los modelos ejecutan la clasificación individual y el mecanismo de votación dura ponderada determina la decisión final.
- Transmisión del dato validado al contrato inteligente desplegado en la red local simulada.
- Registro automático del estado de la ejecución, etiquetado como éxito o fallo según el comportamiento del sistema.

Todas las ejecuciones fueron registradas y documentadas, permitiendo identificar de forma precisa los eventos que originaron los fallos tempranos observados en las pruebas iniciales, los cuales se asociaron a condiciones de sincronización y al proceso de validación del hash de integridad. La consistencia del entorno y la configuración experimental hizo posible reproducir estos eventos de forma controlada, confirmando la estabilidad del sistema una vez superada la fase inicial de ajuste.

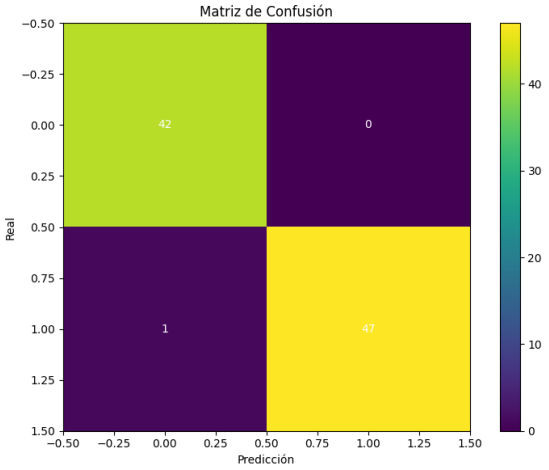

Fig. 3 *Matriz de confusión etiquetas reales y predichas de la propuesta* ***Fuente:*** *elaboración propia*

La Figura 3 presenta la matriz de confusión correspondiente al desempeño del clasificador implementado. En esta representación se observa

que el modelo identificó correctamente 42 instancias negativas (TN) y 47 instancias positivas (TP). La ausencia de falsos positivos (FP = 0) evidencia que el sistema no emitió predicciones positivas incorrectas, lo cual se traduce en una especificidad. Por otra parte, únicamente se registró un falso negativo (FN = 1), indicando un margen mínimo de error al identificar la clase positiva.

Esta distribución demuestra que el modelo posee una capacidad sobresaliente para distinguir entre datos válidos e inválidos, manteniendo un equilibrio adecuado entre sensibilidad y especificidad. La eliminación total de falsos positivos es especialmente relevante en sistemas de validación de datos, de forma complementaria, la reducida presencia de falsos negativos sugiere una alta sensibilidad, garantizando que la mayoría de los datos válidos sean identificados como tales. Estos resultados se reflejan en una precisión global de 1.00, evidenciando un rendimiento del clasificador en este escenario.

La Figura 4 muestra la curva ROC del modelo junto con su respectivo valor AUC. La curva asciende casi verticalmente hasta alcanzar la esquina superior izquierda del gráfico, describiendo el comportamiento característico de un clasificador altamente discriminativo. El área bajo la curva (AUC = 1.00) confirma que el modelo logra una separación entre las instancias positivas y negativas. Un AUC igual a 1.00 implica que existe al menos un umbral de decisión donde la tasa de verdaderos positivos (TPR) alcanza el valor máximo posible mientras que la tasa de falsos positivos (FPR) permanece en cero. Esto significa que el modelo logra asignar probabilidades más altas a los casos positivos que a los negativos en el 100% de las comparaciones posibles. Dicho comportamiento evidencia una capacidad discriminativa ideal, lo que confirma que las fronteras de decisión generadas por el clasificador separan completamente ambas clases sin superposición entendiendo los datos de manera adecuada.

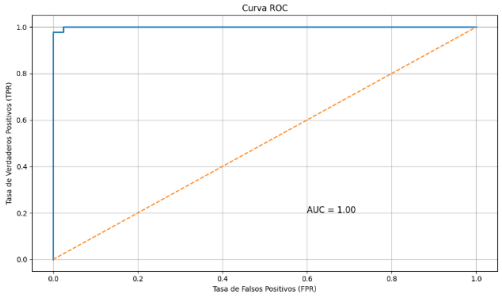

Fig. 4 *Tasa de verdaderos positivos del modelo*
Fuente: *elaboración propia*

4.6 Despliegue

Finalmente, se avanzó en el despliegue del sistema en un ambiente de producción (6.1), probando la solución con datos reales en una red local de blockchain. Este paso permitió validar la escalabilidad y adaptabilidad del sistema a nuevos contextos, asegurando que la arquitectura pueda ser reutilizada en otros dominios contractuales que requieran validación automatizada de datos externos.

5. RESULTADOS

Para validar la efectividad y estabilidad del enfoque propuesto, se diseñó y ejecutó una arquitectura experimental que simula un entorno de operación de contratos inteligentes donde se aplican procesos de validación de envió de datos mediante oráculos distribuidos. Esta arquitectura fue sometida a un protocolo de pruebas basado en 60 ejecuciones secuenciales, controladas y monitorizadas, con el fin de verificar su rendimiento y capacidad de respuesta ante posibles fallos, como se observa en la siguiente imagen.

Fig. 5 *Proceso de evaluación.* ***Fuente:*** *elaboración propia*

5.1 Configuración de la arquitectura

La arquitectura propuesta está compuesta por los siguientes componentes funcionales:

- Contrato inteligente principal, encargado de recibir y almacenar datos provenientes de oráculos.
- Oráculos externos, los cuales proporcionan entradas a partir de datos simulados o reales sobre condiciones agrícolas (como temperatura, humedad y precipitación).
- Módulo de votación por mayoría (hard voting), que consolida las decisiones de múltiples oráculos sobre la validez de un conjunto de datos.
- Subsistema de logging y monitoreo, que registra el resultado de cada ejecución en tiempo real, etiquetando los eventos como "Exitoso" o "Fallido".

5.2 Secuencia de ejecuciones

Cada ejecución consiste en una simulación de ingreso de datos desde múltiples oráculos hacia el contrato inteligente. La secuencia completa se compone de 60 ejecuciones consecutivas numeradas del 1 al 60. El resultado de cada ejecución se clasifica binariamente como:

- Éxito (1): Cuando los datos son correctamente validados y el contrato inteligente responde con la aceptación y registro del evento.
- Fallo (0): Cuando ocurre un error en la validación, sincronización, o en la transmisión de datos.
- Evaluación latencia en 60 ejecuciones
- Evaluación tiempo promedio de modelos

5.3 Visualización del comportamiento

Para ilustrar el comportamiento del sistema durante las 60 ejecuciones, se construyó un gráfico de dispersión, donde el eje horizontal representa el número de ejecución y el eje vertical representa el estado del resultado (0 para fallos y 1 para éxitos). A nivel visual se implementaron dos zonas claves:

- Zona de calibración inicial: Corresponde a las ejecuciones 1 a 10, donde se esperaba una posible inestabilidad del sistema debido a ajustes y carga inicial. Esta zona fue resaltada en amarillo para facilitar su identificación.
- Zona de error: Representa el área de fallos (resultado igual a 0) y está sombreada en color rosa para indicar visualmente la ocurrencia de eventos no exitosos.

5.4 Análisis de comportamiento

Durante la fase de calibración, se registraron dos fallos en las ejecuciones 10 y 12. Estos fallos tempranos fueron analizados y asociados a condiciones de sincronización en la red de oráculos y al tiempo de respuesta en la validación del hash de integridad. A partir de la ejecución número 13 hasta la 60, no se volvió a presentar ningún fallo, lo cual evidencia que las condiciones del entorno y de la arquitectura se estabilizaron correctamente luego de los primeros ajustes.

5.5 Descripción de resultados

Este procedimiento permitió evaluar de manera controlada la confiabilidad del enfoque propuesto, destacando lo siguiente, como se observa en la siguiente figura 6:

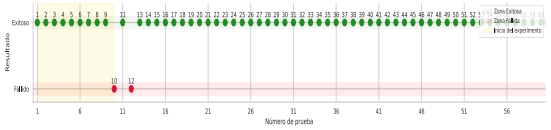

Fig. 6 *Ejecución de pruebas utilizando la arquitectura propuesta*

El procedimiento de evaluación del enfoque propuesto se llevó a cabo mediante la ejecución de 60 pruebas consecutivas sobre la arquitectura diseñada, utilizando Ganache como entorno de simulación local, Truffle para la gestión y despliegue de contratos inteligentes, y Web3.py para la interacción programática con la red blockchain. En cada intento, se registraron tanto el número de prueba como su resultado (éxito o fallo), con el propósito de validar el comportamiento del sistema frente a eventos simulados de integridad de datos.

El sistema alcanzó una tasa de éxito del 96,66% (58 de 60), con solo dos fallos detectados en las pruebas 10 y 12, ambos ocurridos en la fase inicial del experimento. La segmentación visual facilitó la interpretación del desempeño del sistema, demostrando que la incorporación de fases de calibración y monitoreo permanente fue clave para detectar y corregir errores tempranos. En conjunto, los resultados evidenciaron que la arquitectura propuesta logró mantener un comportamiento estable y confiable, reafirmando su aplicabilidad en contextos que requieren validación automatizada de datos sobre infraestructura blockchain

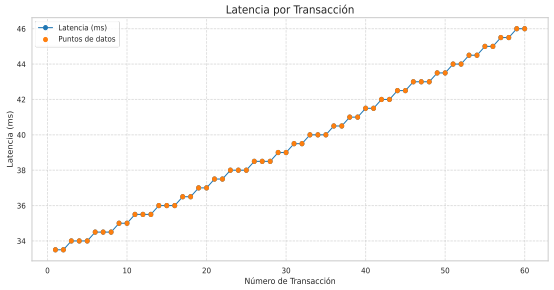

Fig. 7 *Análisis de latencia por transacciones*
Fuente: *elaboración propia*

En la Figura 7 se observa el comportamiento de la latencia durante la ejecución de 60 transacciones consecutivas. La prueba se ejecutó en un entorno controlado sin procesos paralelos de alto consumo, lo cual garantiza una carga mínima del sistema durante la evaluación.

La gráfica ilustra claramente una tendencia ascendente en la latencia medida en milisegundos

(ms), con valores que oscilan inicialmente entre 33.5 ms y alcanzan hasta 46.0 ms al final del conjunto de transacciones. La curva muestra una evolución escalonada y progresiva, lo que sugiere que los tiempos de respuesta del sistema van aumentando de forma acumulativa conforme avanza la carga transaccional.

Este patrón puede deberse a diversos factores internos del entorno de pruebas, tales como la saturación progresiva de buffers de red, creciente consumo de recursos del sistema, o la gestión de colas internas en la lógica del oráculo o contrato inteligente asociado al experimento.

Además, los datos representados como puntos anaranjados refuerzan la consistencia del experimento, indicando que no hubo pérdidas, valores atípicos ni caídas abruptas de rendimiento, lo cual es positivo para escenarios que requieren estabilidad, como contratos inteligentes en aplicaciones financieras.

Finalmente, el comportamiento observado puede considerarse aceptable dentro de un entorno de simulación local, pero también invita a realizar pruebas adicionales en redes distribuidas (testnet/mainnet) para validar el impacto de la infraestructura de red descentralizada y el número de nodos sobre la latencia global.

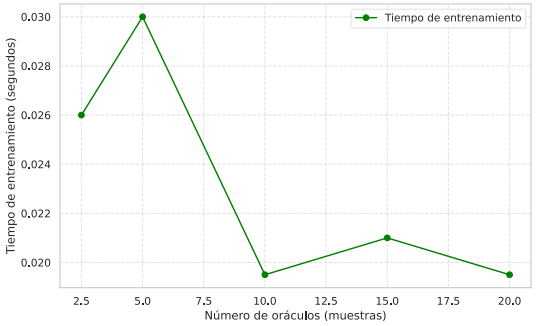

Fig. 8 *Tiempo de entrenamiento promedio de modelos*
Fuente: *elaboración propia*

Finalmente Figura 8 se observa la relación entre el número de oráculos utilizados como muestras de entrada y el tiempo promedio de entrenamiento del modelo IMWV, diseñado y utilizado para validar la integridad de datos enviados a la red en el contexto de contratos inteligentes [20].

El experimento considera cinco configuraciones de entrada, con tamaños de muestra equivalentes a 2, 5, 10, 15 y 20 oráculos, y evidencia cómo varía el tiempo computacional necesario para ajustar el modelo a medida que se incrementa la cantidad de datos recibidos desde los oráculos.

Los valores obtenidos muestran una fluctuación inicial, alcanzando un pico de 0.030 segundos cuando se emplean 5 oráculos, y luego disminuyendo progresivamente hasta estabilizarse alrededor de 0.0195 segundos para configuraciones de 10 a 20 oráculos.

Este comportamiento sugiere que el modelo presenta una adaptabilidad eficiente a partir de cierto umbral de entrada. Inicialmente, el costo computacional puede verse afectado por el sobreajuste en muestras pequeñas o la necesidad de reajustar los pesos de votación. Sin embargo, al aumentar el número de oráculos, el entrenamiento se vuelve más estable, con tiempos cercanos a los 20 milisegundos, lo que representa una ventaja significativa para sistemas en tiempo real que requieren validación de datos sin sacrificar rendimiento.

Estos resultados confirman la viabilidad del enfoque propuesto en escenarios distribuidos, donde el modelo puede ser entrenado dinámicamente con múltiples fuentes de entrada sin representar una carga computacional considerable.

6. CONCLUSIONES

Los resultados obtenidos en el proceso experimental permiten concluir que la propuesta metodológica diseñada para validar datos provenientes de oráculos muestra un comportamiento altamente eficiente desde el punto de vista computacional. En particular, se evidenció que, a pesar del incremento progresivo en el número de oráculos utilizados para el envío de datos a la red, tanto la latencia como el tiempo promedio de entrenamiento se mantuvieron dentro de rangos aceptables y estables.

La prueba de latencia sobre 60 transacciones reveló un patrón creciente pero controlado, con tiempos que oscilaron entre los 33 y los 46 milisegundos, lo cual indica que el sistema puede sostener operaciones continuas sin afectar significativamente el rendimiento. Esta tendencia sugiere que el flujo de datos externos hacia los contratos inteligentes es procesado de manera consistente, incluso bajo escenarios de carga moderada.

Por otro lado, el análisis del tiempo de entrenamiento en función del número de oráculos demostró que la propuesta puede escalar sin incurrir en penalizaciones computacionales críticas. En

configuraciones que iban desde 2.5 hasta 20 oráculos, el tiempo de procesamiento no superó los 30 milisegundos, manteniéndose estable en torno a los 20 milisegundos a partir de cierto umbral. Esto confirma que el sistema es adecuado para ser implementado en entornos que requieren procesamiento en tiempo casi real, como aplicaciones agrícolas, cadenas de suministro o sistemas de alerta temprana.

Contribución de los autores:

CO: Conceptualización, Metodología, Validación, Redacción–borrador original, Investigación, Recursos, Visualización. **HO:** Adquisición de financiación, Investigación, Metodología, Administración del proyecto, Recursos, Validación, Redacción–revisión y edición. **JSGS:** Investigación, Metodología, Validación, Redacción – revisión y edición

Financiación: Este trabajo se desarrolla en el marco del proyecto de investigación "Incremento de la Oferta de Prototipos Tecnológicos en Estado Pre-Comercial Derivados de Resultados de I+D para el Fortalecimiento del Sector Agropecuario en el Departamento del Cauca" (código BPIN 2020000100098), financiado por el Sistema General de Regalías (SGR) de Colombia.

Agradecimientos: Los autores expresan su agradecimiento al Grupo de investigación GTI y al Grupo de Investigación GREDIS, por el tiempo, acompañamiento y apoyo brindado durante el desarrollo de esta investigación.

REFERENCIAS

[1] C. Contini, F. Boncinelli, G. Piracci, G. Scozzafava, and L. Casini, "Can blockchain technology strengthen consumer preferences for credence attributes?," *Agric. Food Econ.*, vol. 11, no. 1, 2023, doi: 10.1186/s40100-023-00270-x.

[2] M. Enayati *et al.*, "Blockchain-Based Location Sharing in 5G Open RAN Infrastructure for Sustainable Communities," *Lect. Notes Networks Syst.*, vol. 333, pp. 571 – 585, 2022, doi: 10.1007/978-981-16-6309-3_54.

[3] C. C. Ordoñez, M. M. Organero, G. Ramirez-Gonzalez, and J. C. Corrales, "Smart Contracts as a Tool to Support the Challenges of Buying and Selling Coffee Futures Contracts in Colombia," *Agric.*, vol. 14, no. 6, pp. 1–19, 2024, doi: 10.3390/agriculture14060845.

[4] S. P. Tan *et al.*, "A review on post-COVID-19 impacts and opportunities of agri-food supply chain in Malaysia," *PeerJ*, vol. 11, 2023, doi: 10.7717/peerj.15228.

[5] K. Almi'ani, Y. C. Lee, T. Alrawashdeh, and A. Pasdar, "Graph-Based Profiling of Blockchain Oracles," *IEEE Access*, vol. 11, no. March, pp. 24995–25007, 2023, doi: 10.1109/ACCESS.2023.3254535.

[6] G. Caldarelli, "Before Ethereum. The Origin and Evolution of Blockchain Oracles," *IEEE Access*, vol. 11, no. April, pp. 50899–50917, 2023, doi: 10.1109/ACCESS.2023.3279106.

[7] A. Beniiche, "A Study of Blockchain Oracles," pp. 1–9, 2020, [Online]. Available: http://arxiv.org/abs/2004.07140.

[8] S. Ellis, A. Juels, and S. Nazarov, "ChainLink: A Decentralized Oracle Network," 2017. [Online]. Available: https://link.smartcontract.com/whitepaper.

[9] Valencia-Payan, "Smart Contract to Traceability of Food Social Selling," *Comput. Mater. Contin.*, vol. 74, no. 3, pp. 4703 – 4728, 2023, doi: 10.32604/cmc.2023.031554.

[10] M. Sharifi, T. Khatibi, M. H. Emamian, S. Sadat, H. Hashemi, and A. Fotouhi, "Development of glaucoma predictive model and risk factors assessment based on supervised models," *BioData Min.*, vol. 14, no. 1, 2021, doi: 10.1186/s13040-021-00281-8.

[11] N. Peppes, E. Daskalakis, T. Alexakis, E. Adamopoulou, and K. Demestichas, "Performance of Machine Learning-Based Multi-Model Voting Ensemble Methods for Network Threat Detection in Agriculture 4.0," *Sensors*, vol. 21, no. 22, 2021, doi: 10.3390/s21227475.

[12] H. Zhang, S. Li, H. Bao, S. Wu, and J. Li, "A Trust-Aware and Cost-Optimized Blockchain Oracle Selection Model with Deep Reinforcement Learning." 2025, doi: 10.48550/arXiv.2502.16133.

[13] M. Di Gennaro, L. Italiano, G. Meroni, and G. Quattrocchi, "DeepThought: A Reputation and Voting-Based Blockchain Oracle," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 13740 LNCS, pp. 369–383, 2022, doi: 10.1007/978-3-031-20984-0_26.

[14] P. Liu, Y. Xian, C. Yao, P. Wang, L. Wang, and X. Li, "A Trustworthy and Consistent Blockchain Oracle Scheme for Industrial Internet of Things," *IEEE Trans. Netw. Serv. Manag.*, vol. 21, no. 5, pp. 5135–5148, 2024, doi: 10.1109/TNSM.2024.3399837.

[15] G. Palaiokrassas, S. Bouraga, and L. Tassiulas, "Machine Learning on Blockchain Data: A

Systematic Mapping Study," *Elsevier BV*, 2024, [Online]. Available: http://arxiv.org/abs/2403.17081.

[16] S. Woo, J. Song, and S. Park, "A distributed oracle using intel SGX for blockchain-based IoT applications," *Sensors (Switzerland)*, vol. 20, no. 9, 2020, doi: 10.3390/s20092725.

[17] H. Taherdoost, "Blockchain and Machine Learning: A Critical Review on Security," *Information*, vol. 14, no. 5, 2023, doi: 10.3390/info14050295.

[18] C. Connors and D. Sarkar, "Survey of prominent blockchain development platforms," *J. Netw. Comput. Appl.*, vol. 216, p. 103650, 2023, doi: https://doi.org/10.1016/j.jnca.2023.103650.

[19] H. Xiong, M. Chen, C. Wu, Y. Zhao, and W. Yi, "Research on Progress of Blockchain Consensus Algorithm: A Review on Recent Progress of Blockchain Consensus Algorithms," *Futur. Internet*, vol. 14, no. 2, 2022, doi: 10.3390/fi14020047.

[20] C. C. Ordoñez, G. Ramirez-Gonzalez, and J. C. Corrales, "Enhancing Data Integrity in Blockchain Oracles Through Multi-Label Analysis," *Appl. Sci.*, vol. 15, no. 5, 2025, doi: 10.3390/app15052379.

[21] C. De and G. Barrios-puente, "Cobertura de precios para el café, utilizando el mercado de futuro," *Rev. Mex. Ciencias Agrícolas*, vol. 13, no. 6, pp. 1147–1154, 2022.

[22] B. H. Shekar and G. Dagnew, "Grid Search-Based Hyperparameter Tuning and Classification of Microarray Cancer Data," in *2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP)*, 2019, pp. 1–8, doi: 10.1109/ICACCP.2019.8882943.