Predicción de la mediana del precio de NFT's con LSTM

1st David Nuñez
dept. Matemáticas
Universidad Nacional de Colombia
Bogotá, Colombia
dnunezq@unal.edu.co

2nd Fabián Andres Ruiz Tortello
dept. Matemáticas
Universidad Nacional de Colombia
Bogotá, Colombia
fruiz@unal.edu.co

3rd María Sol Botello León
dept. Matemáticas
Universidad Nacional de Colombia
Bogotá, Colombia
mbotello@unal.edu.co

4th Sofia Salinas Rico
dept. Matemáticas
Universidad Nacional de Colombia
Bogotá, Colombia
ssalinas@unal.edu.co

***Abstract*—An LSTM neural network was trained to predict the median next day price of unique tokens belonging to a smart contract on blockchain with data taken from open sea and then processed to remove noise and smooth the volatility of this market, using a clustering algorithm and a decision forest we were able to give an estimate of the value of unseen unique tokens and obtained a minimum accuracy of 95 % and a maximum of 98% in different metrics.**

Translated with www.DeepL.com/Translator (free version)

***Index Terms*—NFT, neural, price.**

I. Introducción

El surgimiento de la tecnología blockchain, los contratos inteligentes y en general el mundo de las criptomonedas ha generado un nuevo mercado de distintos objetos como imágenes, música o terrenos virtuales. Estos objetos llamados tokens suelen estar caracterizados por ser únicos, es decir, que solo tienen un propietario y por pertenecer a un contrato inteligente. Debido a la naturaleza de los tokens y el creciente interés por la adquisición de estos, en este proyecto se propone generar un modelo que permita calcular la mediana de los tokens de un contrato con el fin de obtener información sobre la variabilidad de los precios de estos tokens.

El objetivo consiste en entrenar una red neuronal con información de todas las transacciones del contrato hasta el día anterior, para predecir la mediana de las transacciones del día actual. Esto se hace, ya que si se sabe qué mediana tendrán las transacciones del día de hoy, los potenciales compradores tendrán cierta información del precio general de los tokens, si este está en aumento o no y podrán tomar una decisión más acertada, sobre el precio de los tokens.

En este jupyter notebook pueden ver todo el proceso que se explica en este documento, además de los scripts en Github.

II. Materiales y Métodos

A. Definiciones

- Blockchain: Libro mayor compartido e inmutable que facilita el proceso de registro de transacciones y de seguimiento de activos en una red de negocios. Un activo puede ser tangible (una casa, un auto, dinero en efectivo, terrenos) o intangible (propiedad intelectual, patentes, derechos de autor, marcas). Prácticamente, cualquier cosa de valor puede ser rastreada y comercializada en una red de blockchain, reduciendo el riesgo y los costos para todos los involucrados. [1]
- NFT: Las siglas NFT significa Non-Fungible Token, un token no fungible, esto es, un token no sustituible de forma idéntica. Un ejemplo de token fungible es el dinero, ya que todo billete de 20 dólares vale 20 dólares. Los token no fungibles son, por ejemplo, las obras de arte, en el sentido que una obra de arte no es equivalente a otra. [4].
- Token: Es la representación digital de una "unidad de posesión" que puede ser intercambiada entre las partes.
- Contrato inteligente (Smart Contracts): Algoritmo de software que, cuando se activa, ejecuta automáticamente instrucciones para transferir tokens. Básicamente, es un contrato que establece que cuando se reciba un bien o activo, se acepte y confirme, se enviará el pago.

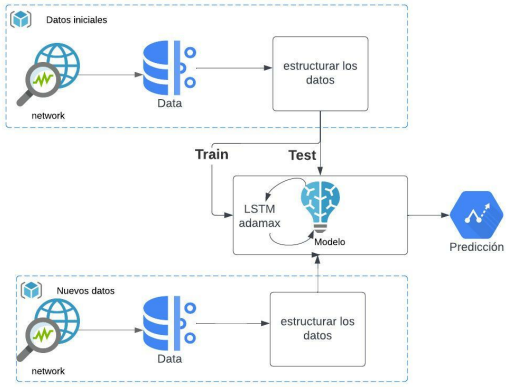
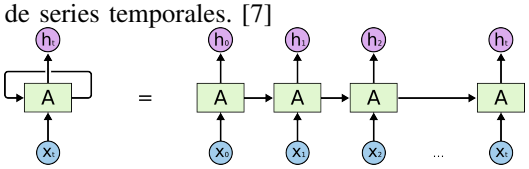


Fig. 1. Diagrama de la solución planteada imagen en mejor calidad

Utilizaremos las palabras contrato y conjunto de tokens alternadamente.

B. *Métodos*

- Red Neuronal LSTM: El LSTM es una red neuronal recurrente artificial (RNN), utilizada frecuentemente en el campo del Deep Learning. A diferencia de otras redes neuronales, el LSTM tiene conexiones de retroalimentación, lo que permite ser muy bueno para clasificar, procesar y hacer predicciones basadas en datos de series temporales. [7]



¿Por qué se decidió usarla? Teniendo en cuenta que en el mercado NFT los compradores son estratégicos, toman en cuenta los precios y tendencias de días pasados para actuar en el futuro, en corto, mediano y largo plazo, a diferencia de las RNN, LSTM tiene un mejor rendimiento con memoria a mediano y largo plazo.

- Ventaneo: Es un (hiper) parámetro que trata de escoger el menor intervalo de tiempo en el que se repita algún patrón entre ventana actual y la siguiente en el orden de la serie temporal. Un ejemplo de como el ventaneo puede servir cuando se usa en predicciones, es al intentar predecir palabras de un libro, si se le da como entrada a la red todas las palabras del libro solas, será difícil predecir la siguiente porque no tiene contexto, con el ventaneo se le como entrada las palabras acompañadas de la frase inmediatamente anterior a la palabra, así si se escoge un ventaneo de 5 la entrada será la palabra junto con las 4 palabras anteriores, lo cual le dará a la red el contexto y hará que las predicciones sean más sencillas.

La arquitectura de nuestra solución **(ver Fig 2)** está resumida en obtener los datos, procesarlos y entrenar el modelo de red neuronal.

El proceso inicia con la obtención de todas las transacciones del conjunto de tokens para los que queremos predecir la mediana, en este caso el conjunto es Fluf World. A través de Etherscan descargamos toda la información relacionada con las transacciones de compra de tokens de este contrato, Etherscan es una web que se encarga de extraer y organizar información de la blockchain de Etherium. Los datos obtenidos van desde 07-08-2021 hasta 26-11-2022, que equivalen 460 días con un total de 17090 transacciones. Terminado el proceso de obtención de datos, se calculan los parámetros que serán las entradas de la red neuronal.

Al tener las transacciones, es posible calcular el promedio, la mediana y la desviación estándar por día. Esto es fácil de hacer, ya que las transacciones están ordenadas por fecha, luego solo es calcular estas estadísticas de manera habitual. La tabla 1 **(ver Table 1)** es un ejemplo donde se muestra parte de la información de las transacciones del día 2021-08-07. La tabla 2 **(ver Table 2)** muestra el promedio, la mediana y la desviación estándar para los primeros 5 días del contrato.

Date Time (UTC)	NFT	Token ID	Price
2021-08-07 23:15:42	FLUF	8834	0.48 ETH
2021-08-07 23:53:20	FLUF	8893	0.42 ETH

TABLE I
PARTE DE LA INFORMACIÓN DE LAS TRANSACCIONES DEL DÍA 2021-08-07

date	average	median	deviation
2021.08.07	0.415	0.415	0.091923882
2021.08.08	0.437840835	0.4	0.345588205
2021.08.09	1.840529383	1.49	1.621085109
2021.08.10	1.413946572	1	2.167664753
2021.08.11	1.064519956	0.77	1.230842091

TABLE II
DATOS ESTADÍSTICOS DEL DÍA 7 AL 11 DEL MES 08 DEL AÑO 2021, LOS PRIMEROS DÍAS DEL CONTRATO

Una vez calculados estos datos, tenemos que decidir que vamos a predecir, los principales candidatos serían la mediana y el promedio, en principio se pensaría en predecir el promedio, pero al graficar la mediana y el promedio es claro que el promedio es muy ruidoso **(ver Fig 2)**, lo cual puede conducir a que sea más difícil de aprender para la red neuronal. Esta es la razón para preferir predecir la mediana.

Basta ver la gráfica **(ver Fig 2)** para darse cuenta de que aunque la mediana es menos ruidosa que el promedio tiene bastante ruido, en este caso decidimos suavizar la mediana en un intento de capturar los patrones importantes en los datos mientras que se elimina ruido. Este proceso utiliza una fórmula sencilla, para calcular la median suavizada de un día tomaremos las medianas suavizadas de los dos días inmediatamente anteriores y la mediana de ese día. En el caso de querer calcular la mediana suavizada de los datos de la tabla 2 **(ver Table 2)**, para los primeros dos días la mediana suavizada será la misma mediana, luego la mediana suavizada del tercer día es el promedio entre las dos medianas suavizadas anteriores y la mediana de ese día, así la mediana suavizada del 09-08-2021 es 0.768:

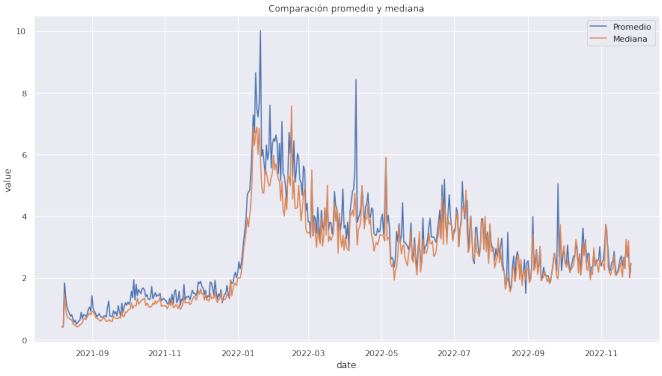


Fig. 2. Comparación entre la mediana y el promedio (imagen en mejor calidad)

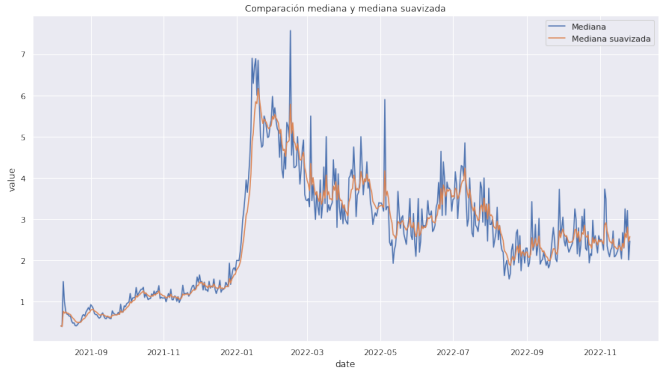


Fig. 3. Comparación entre la mediana y mediana suavizada imagen en mejor calidad

$$m_{s_9} = \frac{0.415 + 0.4 + 1.490}{3} = 0.768$$

Luego podemos ver la diferencia entre ambas **(ver Fig 3)**. Terminado este paso tenemos 4 métricas para la entrada y sabemos qué métrica se usará para la salida. Para entrenar el modelo vamos a agregar una métrica más a las entradas y a esta la llamamos retorno, este consiste en mostrar que tanto más grande o más pequeño es un valor con respecto a otro de la mediana suavizada en términos porcentuales. Así pues, el retorno del primer día será 0 y para el resto de días se hará la razón de la mediana suavizada entre el día anterior y el día actual y se le restará 1. Así entonces, el retorno del 08-08-2021 es -0.036144578:

$$r = \frac{0.4}{0.415} - 1 = -0.036144578$$

Con todas los datos de entrada ya calculados, lo último que hacemos es normalizar los datos con el fin de que todos ellos queden entre -1 y 1, esto permite que todos los parámetros estén en una escala parecida, lo que acelera el entrenamiento y mejorar la generalización de las redes neuronales [8].

Finalmente, realizamos una partición de nuestros datos, 80% para entrenamiento de la red neuronal y 20% para probar el modelo, al ser una serie de tiempo se tomaron los primeros registros que conformaban el 80% y el resto fue para prueba. Luego entrenamos la red neuronal, utilizamos un modelo de red neuronal LSTM de una única capa con 64 neuronas, tomando de entrada promedio, mediana, desviación estándar, mediana suavizada y retorno, con los datos normalizados y para dar más estabilidad a la entrada se utiliza un ventaneo de 5 días, esto para que el modelo tenga más información, tener en cuenta la temporalidad, utilizando un ventaneo de 5 días, esto se hace con la intención

y una capa de salida con una única neurona. Se utiliza el error cuadrático medio como función de pérdida y un optimizador adamax. Se escogió adamax como optimizador, ya que es computacionalmente eficiente, no requiere demasiada memoria y además es adecuado en problemas no estacionarios y donde se presenta mucho ruido como es el caso de nuestro problema.

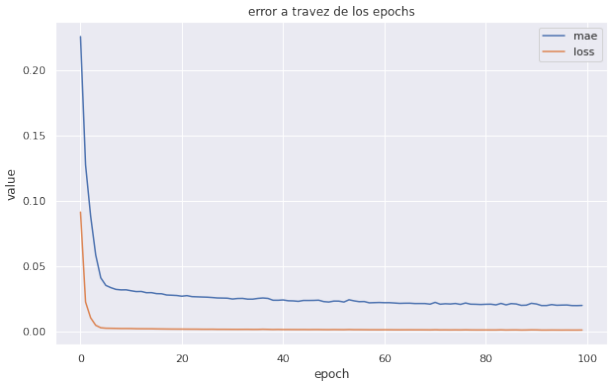


Fig. 4. Comportamiento del mse y mae en el entrenamiento del modelo imagen en mejor calidad

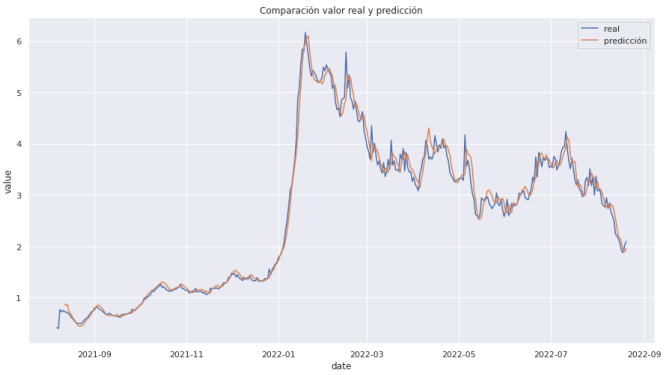


Fig. 5. Predicción en datos de entrenamiento imagen en mejor calidad

III. Resultados

date	average	median	deviation	soften m	return
2021-08-07	0.415	0.415	0.091	0.415	0.000
2021-08-08	0.437	0.400	0.345	0.400	0.037
2021-08-09	1.840	1.490	1.621	0.768	-0.479
2021-08-10	1.409	1.000	2.167	0.722	0.063
2021-08-11	1.064	0.770	1.230	0.753	-0.041

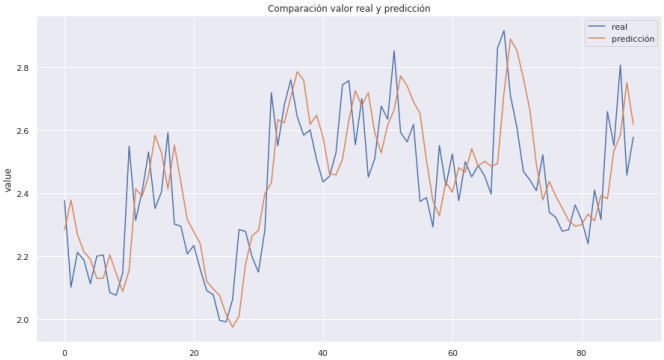


Fig. 6. Predicción en datos de prueba imagen en mejor calidad

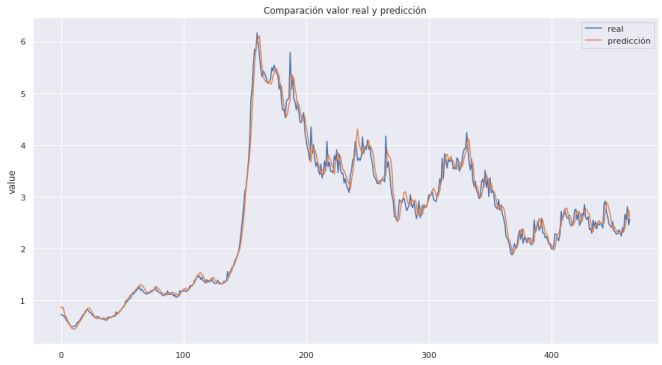


Fig. 7. Predicción sobre todos los datos. imagen en mejor calidad

A. *Configuración experimental*

Para verificar que la función a predecir tiene sentido utilizamos distintas métricas de rendimiento para el modelo. Además, probamos el mismo experimento con otros datos para predecir la mediana de otro modelo.

Para evaluar el modelo usamos las siguientes métricas:

- (MSE) Error cuadratico Medio $MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i)$
- (MAE) $MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_i|$
- (RMSE) $RMSE = \sqrt{(\frac{1}{n}\sum_{i=0}^{1}(y_i - y_i)^2)}$
- (MAPE) $MAPE = 100\%\frac{1}{n}\sum_{i=0}^{n}|\frac{y_i - y_i}{y_i}|$

[3]

Datos	Entrenamiento	Prueba
MSE	0.030	0.022
MAE	0.120	0.122
RMSE	0.175	0.150
MAPE	4.489	4.970

Después de entrenar el modelo se observa el comportamiento de la función de error (mse) y el error (mae) **(ver Fig. 4)** donde se evidencia que en los primeros epochs los dos errores caen rapidamente, y después el mae decae en una menor porporción mientras que el mse se comporta de una manera casi constante.Luego de esto se realizó la predicción de los datos;para la predicción de los datos de entrenamiento se obtuvo un buen rendimiento del modelo **(ver Fig. 5)** para las métricas MSE, MAE y el RMSE. Además, el MAPE nos indica que en promedio el modelo aproxima el valor real con un 96% de precisión.

Las métricas para datos de entrenamiento y datos de prueba obtuvieron resultados similares, lo cual es un buen indicio de que el modelo generalizó y no realizó overfitting. Podemos observar que el MAPE nos indíca que el modelo aproxima en promedio el valor real con un 95% de precisión **(ver Fig. 6)**.

Además, al realizar la predicción completa de los datos se calculó él R^2 y se obtuvo 0.98. Es decir, el modelo con las variables propuestas puede representar el 98% de la variabilidad de los datos **(ver Fig. 7)**.

IV. Conclusiones

Como se dijo en la introducción, el cálculo de la mediana es el primer paso en hacer una predicción al precio de los tokens, al estos ser únicos es difícil para los compradores en saber un precio justo, lo que se haría a continuación es, a través de analizar las características de los tokens, crear grupos para aquellos que tengan características parecidas usando un algoritmo de clusterizacion como $k-means$ y luego utilizando árboles de decisión para cada grupo, en los que cada uno de los nodos del árbol es una característica, luego se pasan en esos árboles los tokens y dependiendo la diferencia que había entre el precio y la mediana suavizada del día que se vendió llenar los nodos de los árboles, así luego se puede pasar tokens que nunca han sido vendidos se les da un valor al llegar a alguna hoja del arbol y con ese valor se calcula que tanto por encima o por debajo de la mediana de ese día debe estar.

References

[1] ¿Qué es la tecnología de blockchain? - IBM Blockchain (no date) IBM. Available at: https://www.ibm.com/mx-es/topics/what-is-blockchain (Accessed: December 2, 2022).

[2] Russell, S. and Norvig, P. (2022) “Long short-term memory RNNs,” in Artificial Intelligence: A modern approach. Harlow: Pearson Education Limited.

[3] Pascual, C. (2021) Tutorial: Understanding regression error metrics in python, Dataquest. Dataquest Labs, Inc. Available at: https://www.dataquest.io/blog/understanding-regression-error-metrics/ (Accessed: December 2, 2022).

[4] Binance Academy (no date) Non-fungible token (NFT), Binance Academy. Binance Academy. Available at: https://academy.binance.com/es/glossary/non-fungible-token-nft (Accessed: December 3, 2022).

[5] Time Time Series - LSTM model (no date) Tutorials Point. Available at: https://www.tutorialspoint.com/time_series/time_series_lstm_model.htm (Accessed: December 3, 2022).

[6] Enterprise Blockchains Fundamentals. (2022, septiembre). 101 blockchains academy.

[7] AprendizajeProfundo (no date) Aprendizajeprofundo/Diplomado: Este es un repositorio creado para el diplomado en inteligencia artificial y Aprendizaje Profundo, GitHub. Available at: https://github.com/AprendizajeProfundo/Diplomado (Accessed: December 3, 2022).

[8] Huang, L. Qin, J. Zhou, Y. (2020) ”Normalization Techniques in Training DNNs: Methodology, Analysis and Application”