

Predicción de señales financieras usando una red neuronal convolucional de memoria de corto plazo

Jesus-Alejandro Lechuga-Ortega, Volodymyr Ponomaryov,
Rogelio Reyes-Reyes

Instituto Politécnico Nacional,
Sección de Estudios de Posgrado e Investigación,
Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacán,
México

jlechugao1500@alumno.ipn.mx,
{vponomar, rreyesre}@ipn.mx

Resumen. Las señales financieras tienen como objetivo mantener un registro histórico del valor de las acciones en forma de series de tiempo. Para la predicción de valores en una serie de tiempo se han propuesto diferentes enfoques, centrados principalmente en el análisis estadístico clásico, sin embargo, cada vez existen más sistemas que emplean el aprendizaje automático, demostrando resultados optimistas para la predicción de señales financieras con un rango de error admisible. En el presente trabajo se emplea un método de aprendizaje automático basado en una red neuronal convolucional de memoria de corto plazo (ConvLSTM2D), donde a partir de la información obtenida de diferentes índices bursátiles de alta importancia a nivel mundial, permite predecir el valor de cierre para el día siguiente; dando especial énfasis al índice Standard & Poor's 500 (S&P500), debido a su gran longevidad e importancia a nivel mundial. Inicialmente se realiza un preprocesamiento de los datos obtenidos de los índices bursátiles, para interpolar aquellos que falten; posteriormente se realiza una extracción de características a partir de indicadores técnicos creando ventanas de tiempo, que servirán para el entrenamiento y validación del sistema propuesto. El sistema obtiene un MSE promedio de $1.22e-4$, demostrando resultados favorables para la predicción de señales financieras.

Palabras clave: Series de tiempo, extracción de características, procesamiento de señales, LSTM, ConvLSTM2D, aprendizaje automático, machine learning.

Prediction of Financial Signals Using a Short-Term Memory Convolutional Neural Network

Abstract. Financial signals aim to keep a historical record of stock value in the form of time series. Different approaches have been proposed for the prediction

of values in a time series, mainly focused on classical statistical analysis, however, there are more and more systems that use machine learning, showing optimistic results for the prediction of financial signals with a range of permissible error. In the present work, an automatic learning method based on a short-term memory convolutional neural network (ConvLSTM2D) is used, where, based on the information obtained from different stock market indices of high importance worldwide, it allows predicting the closing value for the next day; giving special emphasis to the Standard & Poor's 500 (S&P500) index, due to its great longevity and importance worldwide. Initially, a preprocessing of the data obtained from the stock indices is carried out, to interpolate those that are missing; Later, an extraction of characteristics is carried out from technical indicators, creating time windows, which will serve for the training and validation of the proposed system. The system obtains an average MSE of $1.22e-4$, showing favorable results for the prediction of financial signals.

Keywords: Time series, feature extraction, signal processing, LSTM, ConvLSTM2D, machine learning.

1. Introducción

Existe un gran registro histórico de los mercados financieros en forma de series de tiempo, que desde hace varios años se busca modelar y predecir; sin embargo, no es una tarea fácil, debido a su alta volatilidad y complejidad, sin olvidar las múltiples variables estocásticas y no estacionarias que comprenden las señales de este tipo [1]. Diariamente, se genera una gran cantidad de datos para las señales financieras, es por esto, que se ha despertado un gran interés en el área por parte de los investigadores para predecir su comportamiento [2]. El aprendizaje automático es una herramienta moderna para la predicción y extracción de características, debido a esto, demuestran una gran eficiencia para la predicción de las señales financieras [3].

Las redes neuronales artificiales (ANN), han demostrado un buen rendimiento para la predicción de señales financieras, debido a su capacidad de extraer características primordiales, además de reconocer patrones de información en espacios de alta dimensión [4-5].

Para predecir las señales financieras el modelo usado son las redes neuronales de memoria de corto plazo (LSTM), las cuales son un tipo de redes neuronales recurrentes [6]. Las LSTM tienen la capacidad de trabajar con intervalos de tiempo para predecir las señales financieras, para este fin, es necesario tomar en consideración no solamente los datos actuales, sino también los datos previos de la señal [7].

Un modelo más que demuestra tener resultados favorables para la predicción de señales financieras, son las redes convolucionales de memoria de corto plazo (ConvLSTM2D), siendo una variación de las redes LSTM [8]. Debido a la alta complejidad de las señales financieras, las redes ConvLSTM2D pueden encontrar patrones espaciotemporales a partir de la información recolectada [9].

En este artículo se presenta un sistema para predecir las señales financieras mediante el uso de ConvLSTM2D, para predecir el precio de cierre del índice Standard & Poor's 500 (S&P500) a partir de la información proporcionada por los índices mundiales más importantes. La principal contribución de este trabajo es la siguiente: una novedosa forma para el entrenamiento a partir de validaciones para las ConvLSTM2D usando ventanas de tiempo, con la intención de obtener una mejor precisión en la predicción de las señales financieras.

2. Trabajos relacionados

Hoseinzade et al. [10], proponen 2 sistemas basados en redes neuronales convolucionales (CNN), estos trabajan de forma bidimensional y tridimensional usando diferentes índices bursátiles americanos: Nasdaq Composite (NSQ), Dow Jones Industrial Average (DJI) y S&P500 para predecir el movimiento de los mercados. Inicialmente, extraen 80 indicadores técnicos de cada índice. Posteriormente el 60% de la información recopilada, es utilizada para el entrenamiento, 20% para validación y 20% para pruebas de la red.

Sin embargo, al utilizar pocas características, no generan suficiente información para el entrenamiento de las CNN, lo que no permite una correcta generalización del problema planteado. Adicionalmente, el uso de las técnicas de reducción de dimensiones en las características extraídas no demuestra un menor error de predicción en el sistema propuesto.

Ye et al. [11], presentan un algoritmo basado en CNN para predecir señales financieras. Para obtener información de los mercados, utilizan los índices estadounidenses: Johnson Outdoors Inc (JOUT), GlaxoSmithKline plc (GSK), DJI y S&P500. Primeramente, para la extracción de características relevantes usan la deformación dinámica del tiempo (DTW) y la divergencia de Jensen-Shannon (JS) con la intención de extraer características similares a las DTW. Seguidamente, efectúan una división de los datos para el entrenamiento y prueba del sistema.

Adicionalmente, aplican la técnica Transfer-Learning (TL), la cual a partir de la información obtenida de un índice se entrena otra red CNN con la intención de predecir el índice objetivo. No obstante, el aplicar la DTW a los índices, es un enfoque inapropiado, debido al alto costo computacional que conlleva el extraer la cantidad adecuada de características para un mismo índice.

Jerez et al [12], implementan varios sistemas basados en redes neuronales perceptron multicapa (MLP), redes neuronales recurrentes (RNN) y CNN para pronosticar el movimiento de diferentes mercados financieros a nivel global.

Utilizan los datos de Apertura, Máximo, Mínimo y Cierre (OHLC) de las señales financieras. Emplean los índices DJI, NSQ y S&P500. En primer lugar, realizan una extracción de características mediante el algoritmo K-means para generar de 2 a 5 clases, dando énfasis a la volatilidad de los mercados.

Seguidamente, seccionan los datos en 80% para entrenamiento y 20% para pruebas, además de generar un subconjunto de validación, conformado por el último 20% de los

datos del conjunto de entrenamiento; es importante mencionar, que el subconjunto de validación ayuda a mitigar el sobreajuste (*overfitting*), o el desajuste (*underfitting*) de las redes.

Pese al inconveniente de seleccionar la cantidad de clases adecuadas como características, es necesario conocer la clase esperada para una predicción adecuada. Lo cual es un proceso complicado en caso de aumentar las características.

Ghosh et al [13], proponen un esquema centrado en el uso de redes neuronales de memoria de corto plazo (LSTM) para la compra y venta de las acciones, usando todas las acciones que comprenden el índice S&P500. Inicialmente dividen los datos en un 25% para extracción de características, 50% para entrenamiento, 25% para validación y prueba del sistema.

Seguidamente, realizan una extracción automática de 3 características utilizando el precio de apertura, cierre y el retorno diario para todas las acciones que comprenden el índice. Posteriormente, seleccionan 93 características empleando bosques aleatorios (RF). Consecutivamente, realizan una clasificación binaria mediante el retorno diario para obtener el movimiento positivo o negativo del índice S&P500. Sucesivamente, crean un nuevo subconjunto a partir de los conjuntos de entrenamiento, validación y prueba obtenidos anteriormente, además de usar como etiqueta la clasificación binaria previamente realizada.

Finalmente, seccionan el nuevo conjunto en un 50% para entrenamiento, 25% para prueba y 25% para validación del esquema. Sin embargo, el esquema de partición de datos causa un sesgo de información a causa de la manipulación realizada con los conjuntos creados y dando como resultado un mal entrenamiento de la LSTM.

Rezaei et al [14], desarrollaron un sistema basado en CNN y LSTM para predecir el precio de cierre de los índices Nikkei 255 (NIK), German Stock 30 (DAX), S&P500, DJI. En primer lugar, realizan una descomposición de modo empírico (EMD) para obtener las componentes significativas.

Luego, obtienen la frecuencia de las señales usando funciones de modo intrínseco (IMF) y efectúan la extracción de características mediante CNN. Posteriormente, se crea un conjunto de ventanas de tiempo a partir de las características. Finalmente, seccionan los datos recopilados donde el 80% los datos son para el entrenamiento y el 20% son para prueba de la LSTM.

Los autores no mencionan cuantas características son necesarias para el entrenamiento de la CNN, otro inconveniente, es que por cada IMF se entrena una LSTM, por lo que se necesita una gran capacidad de cómputo.

Cao et al [15], implementan un algoritmo orientado a predecir el precio de divisas USD/CNY usando redes LSTM. Primeramente, emplean diversas características a nivel de mercado (Mercado de materias primas, mercado de acciones) y macroeconómico (Política monetaria, rango de inversión, inflación, balanza comercial e incertidumbre económica). Posteriormente, se dividen los datos en 2 conjuntos, donde el 70% es para entrenamiento y el 30% es para pruebas. Seguidamente, se entrena una LSTM para cada característica obtenida. Finalmente, se entrena una LSTM con la información obtenida para predecir el movimiento del par USD/CNY.

No obstante, para el entrenamiento de múltiples redes se necesita una gran cantidad de cómputo, otra desventaja, es que el enfocarse solo en un índice, no aporta la información suficiente para predecir otros mercados enfocados a las divisas.

Para hacer frente a los retos anteriores, se propone un sistema para predecir señales financieras basado en redes ConvLSTM2D, usando la información de los índices mundiales más importantes. Su objetivo consta de extraer características a partir de indicadores técnicos para cada índice seleccionado, además de utilizar las señales previamente recopiladas para predecir el precio de cierre del índice S&P500.

3. Sistema propuesto

El sistema consta de cuatro etapas principales: procesamiento de los datos, extracción de características, arquitectura y entrenamiento de la ConvLSTM2D, y postprocesamiento de los datos. La Fig. 1 muestra de forma detallada el diagrama a bloques del sistema propuesto para la predicción de señales financieras.

En primer lugar, se eligen los índices bursátiles más importantes a nivel global: Hang Seng (HSI), Financial Times Stock Exchange (FTSE), Nyse Composite (NYSE), German Stock Index (DAX), Nikkei 255 (NIK), Dow Jones Industries (DJI) y S&P500. Seguidamente, en la etapa de preprocesamiento, se usa la interpolación lineal, con la intención de que las fechas de los índices sean homogéneas; si no es suficiente, se opta por eliminar esos datos.

Posteriormente, en la etapa de procesamiento de datos, se extraen múltiples características, a través de diferentes indicadores técnicos, las cuales son normalizadas; a partir de estas señales, se crean múltiples ventanas de tiempo, que se utilizan para el entrenamiento, validación y prueba del modelo propuesto.

Finalmente, se obtienen las predicciones para cada índice de manera independiente, con la finalidad de obtener información suficiente para predecir el índice S&P500. Cada una de estas etapas se detallarán a continuación.

3.1. Preprocesamiento

Debido a la diferencia horaria de cada uno de los índices, puede existir una discordancia entre las fechas, por lo que se busca agregar información faltante, de tal manera que los índices contengan la misma cantidad de información en la mayoría de las fechas.

Esto se lleva a cabo mediante una interpolación lineal, a partir del intervalo de tiempo empleado para el S&P500 a fin de que los índices trabajen con fechas semejantes durante el lapso estudiado. Finalmente, se considera que en ocasiones los mercados no operan, razón por la que la interpolación no es suficiente y se opta por eliminar esos datos.

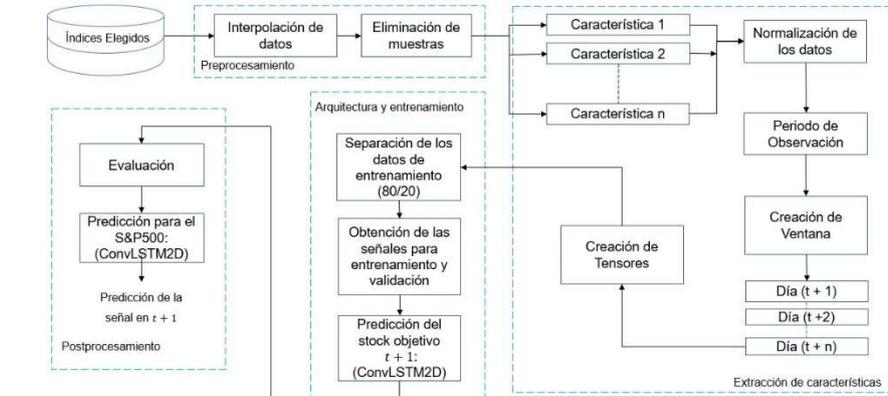


Fig. 1. Diagrama a bloques del sistema propuesto.

3.2. Extracción de características

Inicialmente, se realiza la extracción de características [16], a partir de las señales OHLC, seleccionando un conjunto de indicadores técnicos divididos en 6 subconjuntos: indicadores de tendencia (indican la dirección del movimiento del precio del índice), osciladores de tendencia (miden la desviación media del precio del índice), reconocimiento de patrones (buscan patrones similares en las señales OHLC), transformación de precio (denotan los movimientos abruptos de los índices), indicadores de volatilidad (calculan la rapidez con la que cambia el precio del índice) y estadísticas. Se tiene un total de 31 características extraídas las cuales se muestran en la Tabla 1.

Posteriormente, se realiza una normalización *MinMax* para cada característica extraída, empleando (1):

$$\bar{x}_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}, \quad (1)$$

donde x_{min} y x_{max} son los valores mínimos y máximos de cada valor en la serie de tiempo, x_i es el valor actual y \bar{x}_i es el valor normalizado.

Finalmente, se crean ventanas de tiempo de 1, 5, 10 y 30 días, las cuales corresponden a los datos de entrada para el entrenamiento del sistema propuesto usando una red ConvLSTM2D.

3.3. Arquitectura propuesta

Se hace uso de las redes ConvLSTM2D [17], debido a que estas arquitecturas trabajan con datos espacio temporales. La estructura interna de cada capa ConvLSTM se detalla en (2 - 6):

Indicadores de tendencia	Osciladores de tendencia		Reconocimiento de patrones		Transformación de precio	Indicadores de volatilidad	Estadísticas	
	SMA	STCK	STCD	TWO			AVG	ATR
EMA	RSI	WR	TREOUT		MED	NATR	LNR	LNR
MOM	ADO	CCI	TREIN		TYP	TRANGE	LNR	STD
MACD	RET	TRE	DRG	WCL			TIME	VAR

SMA: Simple Moving Average, EMA: Exponential Moving Average, MOM: Momentum, MACD: Moving Average Convergence/Divergence, STCK: Stochastic Oscillator %K, STCD: Stochastic Oscillator %D, RSI: Relative Strength Index, WR: Williams' %R, ADO: Chaikin A/D Oscillator, CCI: Commodity Channel Index, RET: Return, TWO: Two Crows, TREOUT: Three Outside Up/Down, TREIN: Three Inside Up/Down, TRE: Three Black Crows, DRG: Dragonfly Doji, AVG: Average Price, MED: Median Price, TYP: Typical Price, WCL: Weighted Close Price, ATR: Average True Range, NATR: Normalized Average True Range, TRANGE: True Range, BETA: Beta, PRS: Pearson's Correlation Coefficient, LNR: Linear Regression, LNRA: Linear Regression Angle, LNRS: Linear Regression Slope, STD: Standard Deviation, TIME: Time Series Forecast, Var: Variance.

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * X_{t-1} + W_{ci} \circ C_{t-1} + b_i), \tag{2}$$

$$f_t = \sigma(W_{xf} * X_t + W_{hf} * X_{t-1} + W_{cf} \circ C_{t-1} + b_f), \tag{3}$$

$$o_t = \sigma(W_{xo} * X_t + W_{ho} * X_{t-1} + W_{co} \circ C_{t-1} + b_o), \tag{4}$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c), \tag{5}$$

$$H_t = o_t \circ \tanh(C_t), \tag{6}$$

donde: X_1, \dots, X_n son las entradas de la red, C_1, \dots, C_n son las salidas de cada celda, H_1, \dots, H_n son los estados, H_n, i_t, f_t, o_t son las compuertas de entrada, olvido y salida respectivamente en un tiempo determinado t , cada capa cuenta con su respectiva matriz de pesos W y bias b .

La arquitectura propuesta se encuentra conformada por: una capa ConvLSTM2D, una función de activación *ReLU*, una capa dense: 1, un optimizador: descenso de gradiente estocástico (*SGD*), una función de pérdida: valor medio cuadrático (*MSE*), número de épocas: 300, parada sin cambios: 100 épocas, se usa el gradiente acelerado de *Nesterov*, estos parámetros fueron seleccionados de forma experimental, tomando en consideración [10, 16]. La Tabla 2 muestra los hiperparámetros de la red propuesta.

Tabla 2. Hiperparámetros para ConvLSTM2D.

Capas	Kernel	Recurrent Dropout	Taza de aprendizaje	Momento	Exponential Decay
1	3x3, 5x5	0.1	0.1, 0.001, 0.0001	0.8	0.1

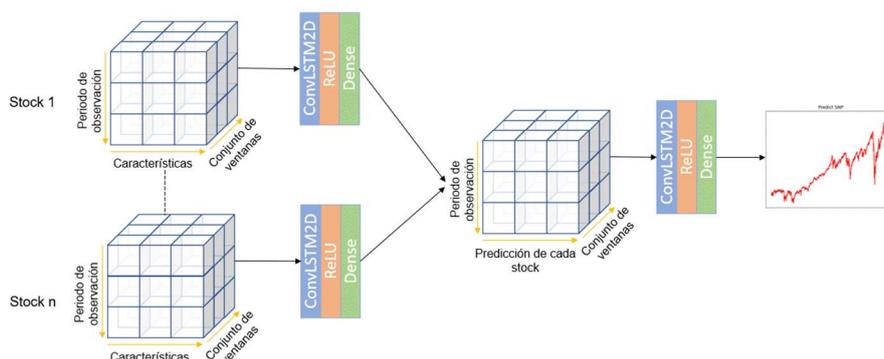


Fig. 2. Arquitectura ConvLSTM2D propuesta Fig. 2. Arquitectura ConvLSTM2D propuesta.

Seguidamente, para el entrenamiento, validación y prueba de las redes propuestas, se usan 5 años de los índices seleccionados, donde el periodo de tiempo analizado va del 01/01/2015 – 31/12/2020, obteniendo 1510 días. Los datos se dividen en 2 conjuntos donde el 80% de los datos son para el conjunto de entrenamiento y el 20% restante es para el conjunto de pruebas, adicionalmente, se crea un subconjunto de validación, el cual consta de un 20% de los datos de entrenamiento para el esquema de validación secuencial [13].

Por último, se entrena una ConvLSTM2D diferente para cada índice seleccionado, con la misma arquitectura, los datos de entrada están conformados por tensores que incluyen la siguiente información: periodo de observación, características seleccionadas y las ventanas de tiempo, como se muestra en la Fig. 2.

3.4. Postprocesamiento

Finalmente, se entrena una red ConvLSTM2D con la misma arquitectura mencionada en la sección 3.3, donde los datos de entrada serán las señales obtenidas por cada una de las redes entrenadas anteriormente, las cuales aportan la información necesaria para predecir el valor de cierre del índice S&P500; en la Fig. 3 se muestran las señales obtenidas de cada modelo entrenado.

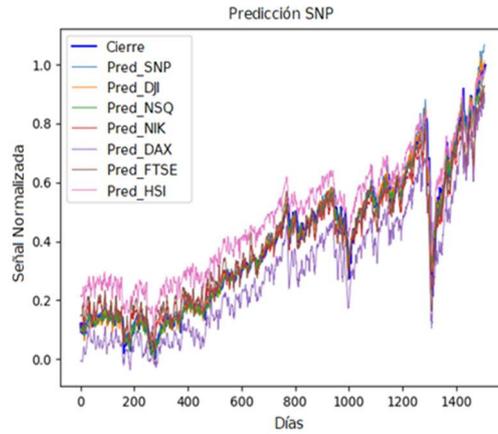


Fig. 3. Predicciones obtenidas para el S&P500 con los datos de los índices seleccionados.

Tabla 3. Resultados obtenidos para los modelos propuestos.

Algoritmo	Características	MSE	RMSE	MAPE	Tiempo de procesamiento (Hrs)
Ventana: 30 días					
LSTM	30	5.83e-3	5.42e-2	2.19e-1	18.54
Bidireccional LSTM	30	1.4e-3	3.73e-2	3.79e-1	16.77
ConvLSTM1D	30	2.3e-3	4.8e-2	6.96e-1	6.14
ConvLSTM2D	30	1.9e-3	4.38e-2	1.75e-1	5.92
Ventana: 10 días					
LSTM	30	7.15e-3	2.67e-2	2.50e-1	9.45
Bidireccional LSTM	30	7.79e-4	2.79e-2	2.15e-1	8.13
ConvLSTM1D	30	9.19e-4	3.03e-2	4.83e-1	6.26
ConvLSTM2D	30	9.3e-3	9.64e-2	1.75e-1	6.10
Ventana: 5 días					
LSTM	30	4.96e-4	2.23e-2	1.46e-1	6.52
Bidireccional LSTM	30	4.4e-4	2.1e-2	1.85e-1	4.28
ConvLSTM1D	30	6.40e-4	2.53e-2	3.05e-1	2.31
ConvLSTM2D	30	1.22e-4	1.11e-2	1.24e-1	1.71
Ventana: 1 día					
LSTM	30	1.95e-4	1.41e-2	2.15e-1	2.44
Bidireccional LSTM	30	1.50e-4	1.23e-2	2.61e-1	1.31
ConvLSTM1D	30	1.71e-4	1.31e-2	3.01e-1	0.68
ConvLSTM2D	30	2.45e-4	2.05e-2	1.46e-1	0.55

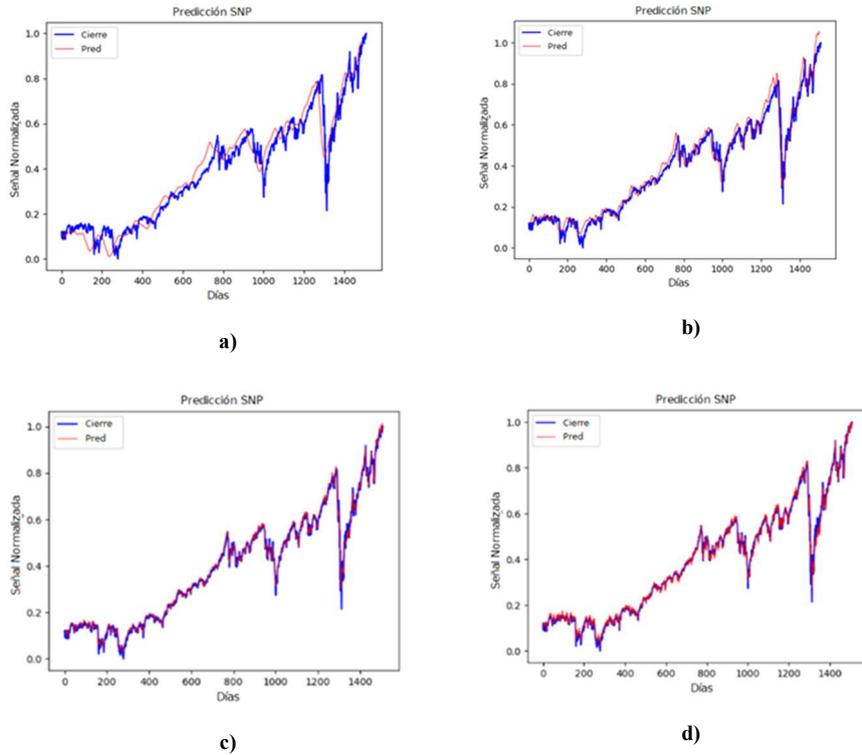


Fig. 4. Predicciones obtenidas por el modelo ConvLSTM2D: a) Ventana 30 días, b) Ventana 10 días, c) Ventana 5 días, d) Ventana 1 día.

Tabla 5. Comparación con [11].

Algoritmo	RMSE
Tr-GRU	2.31e-2
D-LSTM	1.08e-1
D-CNN	1.27e-2
Propuesta (ConvLSTM2D)	1.20e-2

4. Resultados experimentales

Se realizaron experimentos con los modelos LSTM, bidirectional-LSTM, ConvLSTM1D, los cuales usan hiperparámetros por defecto de la librería Tensorflow 2, así como el sistema propuesto basado en una red ConvLSTM2D. Se emplearon las 31 características mostradas en la Tabla 1 y se utilizaron las métricas *MSE*, *RMSE*, *MAPE* (7-9) así como el tiempo de ejecución para la evaluación y comparación de los

Tabla 6. Comparación con [12].

Algoritmo	MSE
MLP	1.64e-4
LSTM	1.6e-4
Conv	1.59e-4
Propuesta (ConvLSTM2D)	1.45e-4

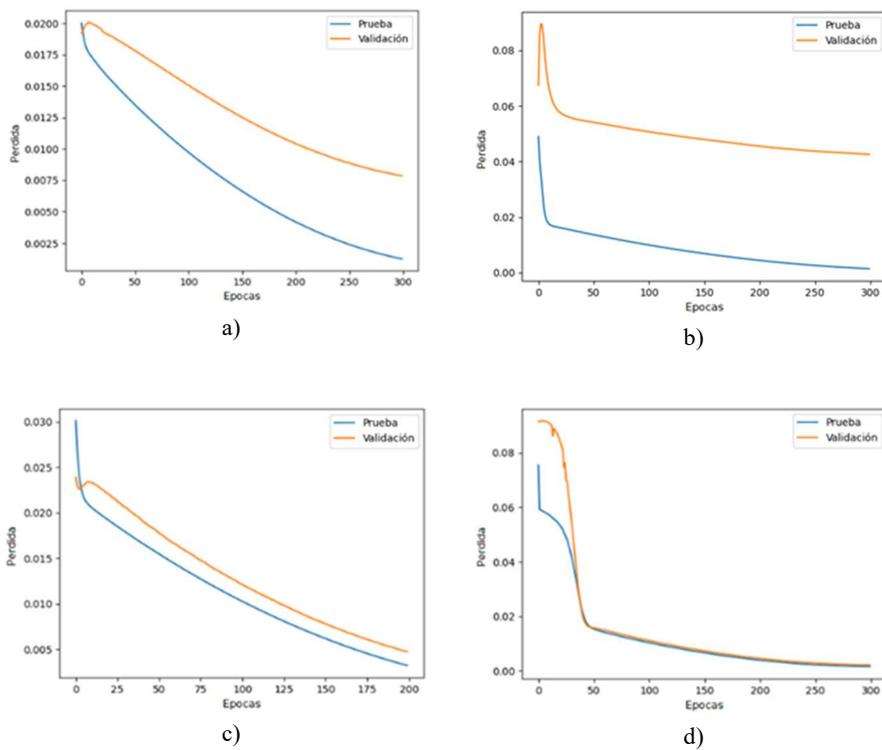


Fig. 5. Curvas de validación y prueba obtenidas para el modelo ConvLSTM2D: a) Ventana 30 días, b) Ventana 10 días, c) Ventana 5 días, d) Ventana 1 día.

diferentes sistemas. Se empleó hardware específico con las siguientes especificaciones: CPU Ryzen 5 5600X 3.7 GHz, GPU Nvidia GeForce RTX 3060 y 16 Gb RAM:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (7)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}, \quad (8)$$

$$MAPE = \frac{100}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|, \quad (9)$$

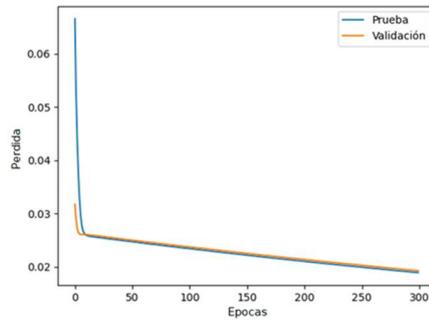


Fig. 6. Curvas de validación y prueba obtenidas del sistema propuesto, para el mismo intervalo de tiempo empleado en [11].

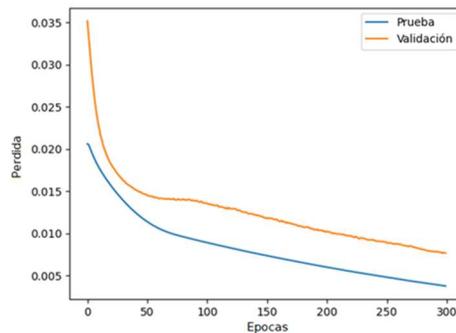


Fig. 7. Curvas de validación y prueba obtenidas del sistema propuesto, para el mismo intervalo de tiempo empleado en [12].

donde: y_i es el valor actual, \hat{y}_i es la predicción y n es el número total de muestras.

Los resultados obtenidos se muestran en la Tabla 3, observándose que el sistema propuesto basado en la red ConvLSTM2D tiene mejores resultados de acuerdo con las métricas elegidas. Se observa que las predicciones obtenidas con ventanas de 10 y 30 días presentan un retraso, mientras que para las ventanas de 1 y 5 días la predicción se encuentra más cercana al precio real de cierre, en especial la ventana de 5 días tiene el error más bajo conforme al MSE ($1.22e-4$), véase la Fig. 4.

Por otra parte, la curva de validación es similar a la de pérdida, donde ambas decrecen suavemente de manera simultánea y proporcional, evitando el *underfitting* y el *overfitting* [18] como se muestra en la Fig. 5.

A continuación, se muestra la comparativa del sistema propuesto con otros autores [11, 12]. En [11], se utilizan 10 años de datos, los cuales van de 16/11/2009-14/11/2019.

El criterio empleado para esta comparación es el *RMSE*, y se tomaron como referencia los 3 modelos propuestos con los mejores resultados obtenidos por los

autores, los resultados comparativos se muestran en la Tabla 5. De acuerdo con los resultados experimentales, el sistema tiene un menor error en la predicción respecto a la métrica RMSE a comparación de [11], esto se debe a la extracción de características primordiales al usar una red ConvLSTM2D. En la Fig. 6 se puede observar que las curvas de validación y pérdida decaen de forma constante a partir de la época 20, por lo cual, el sistema propuesto no presenta *overfitting* ni *underfitting* en comparación con [11] durante el mismo intervalo de tiempo analizado en ambas propuestas.

En relación con [12], el intervalo de tiempo empleado es de 48 años, el cual comprende de 02/01/1970-01/06/2018. El criterio considerado para esta comparación es el *MSE*, y se ocuparon como referencia los 3 algoritmos propuestos con los mejores resultados obtenidos por los autores, los resultados comparativos se muestran en la Tabla 6.

De acuerdo con los resultados experimentales, el sistema tiene un menor error respecto a la métrica MSE a comparación de [12], ya que al utilizar las señales obtenidas de diferentes índices como entrada de datos para la red ConvLSTM2D, se logra un buen entrenamiento para la misma. En la Fig. 7 se puede apreciar que las curvas de validación y pérdida decrecen suavemente a partir de la época 50, debido a esto, el sistema propuesto no presenta *overfitting* ni *underfitting* en comparación con [12] durante el lapso analizado en ambas propuestas.

De acuerdo con los resultados experimentales obtenidos a partir de diferentes intervalos de tiempo, se demuestra que el sistema propuesto tiene mejor desempeño (menor error) en la predicción del precio de cierre para el índice S&P500 debido al uso de diferentes índices técnicos; ya que estos aportan información relevante para un correcto entrenamiento de las redes ConvLSTM2D independientes.

5. Conclusiones y trabajo a futuro

En el presente trabajo se diseñó un sistema para predecir señales financieras basado en redes ConvLSTM2D, en el cual se utiliza la información de los índices más representativos a nivel mundial, entre los que se encuentran: NIK, HSI, DAX, FTSE, NYSE, DJI y S&P500, de los cuales se extraen 31 características relevantes que consisten en: indicadores de tendencia, osciladores de tendencia, reconocimiento de patrones, transformación de precio, indicadores de volatilidad y estadísticas. Posteriormente, se crean ventanas de tiempo de 1, 5, 10 y 30 días para el entrenamiento y validación del sistema propuesto.

Uno de los puntos clave del sistema es el uso de validaciones secuenciales, donde el sistema busca patrones en un cierto intervalo de tiempo sin mostrar todo el periodo estudiado. Ayudando a la red a encontrar patrones relevantes para una predicción eficaz del precio de cierre del índice S&P500.

Los resultados obtenidos para las métricas MSE, RMSE y MAPE fueron $1.22e-4$, $1.11e-2$ y $1.24e-1$ respectivamente, usando una ventana de tiempo de 5 días, demostrando así un desempeño superior a los métodos propuestos en el estado del arte.

El diseño de este tipo de sistemas puede ayudar a los financieros a tomar decisiones efectivas al momento de comprar o vender un activo. Sin embargo, se debe tener en

consideración el riesgo asociado al mercado de acciones, debido a múltiples factores externos que influyen directamente a los mercados. Como trabajo futuro se pretende implementar módulos de atención para mejorar el performance del sistema propuesto, además de crear portafolios de inversión con la intención de tener un retorno aceptable con el menor riesgo posible.

Referencias

1. Marszałek, A., Burczyński, T.: Modeling and forecasting financial time series with ordered fuzzy candlesticks. *Information sciences*, vol. 273, pp. 144–155 (2014)
2. Lau, K. W., Wu, Q. H.: Local prediction of non-linear time series using support vector regression. *Pattern recognition*, vol. 41, pp. 1539–1547 (2008)
3. Arévalo, A., Niño, J., Hernández, G., Sandoval, J.: High-frequency trading strategy based on deep neural networks. *Intelligent computing methodologies*, vol. 9773, pp. 424–436 (2016)
4. Rosenblatt, F.: The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, vol. 65 (1958)
5. Moghaddam, A. H., Moghaddam, M. H., Esfandyari, M.: Stock market index prediction using artificial neural network. *Journal of Economics, Finance and Administrative Science*, vol. 21, pp. 89–93 (2016)
6. Hochreiter, S., Jürgen S.: Long short-term memory. *Neural computation*, vol. 9, pp. 1735–1780 (1997)
7. Cao, J., Li, Z., Li, J.: Financial time series forecasting model based on CEEMDAN and LSTM. *Physica A: Statistical Mechanics and its Applications*, vol. 519, pp. 127–139 (2019)
8. Shi, X., Chen, Z., Wang, H., Yeung, D. Y.: Wong, W. K., Woo, W. C.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, vol. 28 (2015)
9. Gunduz, H., Yaslan, Y., Cataltepe, Z.: Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations. *Knowledge-Based Systems*, vol. 137, pp. 138–148 (2017)
10. Hoseinzade, E., Haratizadeh, S.: CNNpred: CNN-based stock market prediction using a diverse set of variables. *Expert Systems with Applications*, vol. 129, pp. 273–285 (2019)
11. Ye, R., Dai, Q.: Implementing transfer learning across different datasets for time series forecasting. *Pattern Recognition*, vol. 109 (2021)
12. Jerez T., Kristjanpoller W.: Effects of the validation set on stock returns forecasting. *Expert Systems with Applications*, vol. 150, pp 113271 (2020)
13. Ghosh, P., Nufeld, A., Keshari Sahoo, J.: Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. *Finance Research Letters*, pp 102280 (2021)
14. Rezaei, H., Faaljou, H., Mansourfar, G.: Stock price prediction using deep learning and frequency decomposition in *Expert Systems with Applications*, vol. 169 (2021)
15. Cao, W., Zhu, W., Wang, W., Demazeau, Y., Zhang, C.: A deep coupled LSTM approach for USD/CNY exchange rate forecasting. *IEEE Intelligent Systems*, vol. 35, pp. 43–53 (2020)

16. Henrique, B. M., Sobreiro, V. A., Kimura, H.: Literature review: Machine learning techniques applied to financial market prediction. *Expert Systems with Applications*, vol. 124, pp. 226–251 (2019)
17. Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., Woo, W. C.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, vol. 28 (2015)
18. Li, F., Wu, J., Gao, R.: *Neural Networks for Visual Recognition in class notes for CS231*. (2021)