

Arquitectura Neuronal Apoyada en Señales Reconstruidas con Wavelets para Predicción de Series de Tiempo

Ángel García Pedrero, Pilar Gómez-Gil
Coordinación de Ciencias Computacionales
Instituto Nacional de Astrofísica, Óptica y Electrónica
agarciapedrero@ccc.inaoep.mx, pgomez@inaoep.mx

RESUMEN

En el presente trabajo se describe una nueva arquitectura neuronal inspirada en la red recurrente conocida como *red híbrida compleja* (HCNN por sus siglas en inglés) para predicción a mediano plazo de series de tiempo. El modelo propuesto mostró un mejor desempeño que una red neuronal totalmente recurrente y una *feedforward* en la predicción de 56 valores de 11 series de un conjunto utilizado para concursos de predicción de series económicas y financieras, conocido como NN5.

I. INTRODUCCIÓN

La predicción de series de tiempo consiste en la estimación de valores futuros de una serie x_{t+1}, x_{t+2}, \dots , basándose en las observaciones de su pasado, x_1, x_2, \dots, x_t [4]. La predicción es un problema con un amplio rango de áreas de aplicación tales como control, procesamiento de señales, meteorología y economía. Aunque se han desarrollado gran cantidad de métodos para tratar de obtener una estimación precisa, éste continúa siendo un problema abierto al día de hoy.

Debido a su capacidad de abstracción de información esencial a través de aprendizaje basado en ejemplos, las redes neuronales han mostrado ser de los mejores métodos para modelar sistemas temporales. En especial, los mejores desempeños en algunas aplicaciones se han obtenido con redes recurrentes [6]. Dentro de los trabajos que emplean redes neuronales recurrentes para predicción de series de tiempo se encuentran los siguientes: Cai y colaboradores [1] proponen un algoritmo de aprendizaje, que combina optimización de enjambre de partículas y algoritmos evolutivos (PSO-EA). El algoritmo es empleado para entrenar una red recurrente de tipo Elman para predecir series de tiempo. La red es probada con la serie CATS Benchmark, una serie de tiempo artificial de 5,000 datos, con 100 valores faltantes distribuidos en 5 bloques. La predicción de los datos se realiza

mediante 5 redes, cada una de las cuales predice 20 valores. El resultado de la predicción de los 100 valores faltantes tiene un error cuadrático medio (MSE) de 351. El modelo neuronal recurrente, HCNN, propuesto por Gómez-Gil [3] hace uso del análisis de Fourier para entrenar múltiples generadores armónicos y con ello ayudar a la predicción. Un generador armónico es una red totalmente conexa, la cual es capaz de oscilar autónomamente generando series sinusoidales. Cada generador se conecta a otras neuronas recurrentes mediante conexiones de alimentación hacia adelante. Esta red utiliza como algoritmo de entrenamiento retropropagación a través del tiempo. La HCNN es probada en la predicción del comportamiento de electrocardiogramas empleando 512 puntos para entrenar. Los resultados mostrados señalan un MSE de $2.5e^{-3}$. En este trabajo se propone una arquitectura neuronal basada en componentes recurrentes inspirada en el trabajo de Gómez-Gil para predecir series de tiempo. Este artículo está organizado como sigue: la sección II aborda algunos conceptos acerca de la arquitectura propuesta. La sección III presenta una descripción de las series de tiempo utilizadas. Algunos resultados son presentados en la sección IV. Finalmente, la sección V concluye el artículo.

II. MODELO DE PREDICCIÓN

A. Redes Neuronales Recurrentes

Una red neuronal recurrente (RNN) puede ser definida como:

$$y_j(t+1) = \varphi \left(\sum_{i=1}^{m+n} w_{ji} z_i(t) \right) \quad (1)$$
$$z_i(t) = \begin{cases} y_i, & i \leq n \\ u_{i-n}, & i > n \end{cases}$$

donde y_j denota la salida de la j -ésima neurona, w_{ji} representa las conexiones del i -ésimo al j -

ésimo neurón y $\varphi(\cdot)$ es una función diferenciable arbitraria, comúnmente, una sigmoide. Las entradas externas u_i y las entradas recurrentes y_i son representadas como z_i por conveniencia. Existen múltiples modelos de redes neuronales recurrentes, entre los cuales se encuentran las redes de Elman y la red totalmente recurrente. Una red totalmente recurrente tiene una representación muy rica de información sobre el pasado [7] y por tanto, es empleada en este trabajo.

B. Descomposición Wavelet

La Transformada Discreta Wavelet (DWT) permite descomponer una señal en varios componentes de menor resolución empleando una función wavelet. La descomposición consiste en operaciones de filtrado y decimado. Durante la descomposición, filtros de descomposición pasa bajas (L) y pasa altas (H) son aplicados a la señal. Las salidas de los filtros L y H son llamadas coeficientes de aproximación (cA) y de detalle (cD), respectivamente. Este proceso es conocido como árbol de descomposición wavelet.

Un árbol de descomposición wavelet produce una señal de la que se puede obtener información importante. La descomposición puede ser iterada hasta que los detalles de la señal consistan de un solo un ejemplo. Dependiendo de la complejidad de la señal, el nivel de descomposición puede ser seleccionado. La Figura 1 ilustra un árbol de descomposición wavelet de tres niveles.

La reconstrucción de la señal original o Transformada Inversa Wavelet Discreta (iDWT) es realizada mediante los procesos de undecimado y filtrado. El undecimado consiste en insertar ceros entre los valores de la señal de entrada. Para realizar el filtrado se emplean filtros de síntesis. Los filtros de síntesis están basados en los filtros de descomposición y denotados por L' y H' . La señal original reconstruida es obtenida sumando las salidas de los filtros de síntesis. En la Figura 2 se muestra el proceso de reconstrucción, las entradas de los filtros L' y H' son los coeficientes de aproximación y detalle, respectivamente.

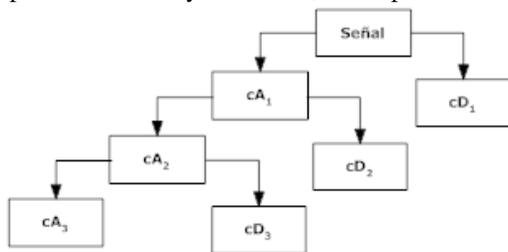


Figura 1. Árbol de descomposición wavelet de tres niveles.

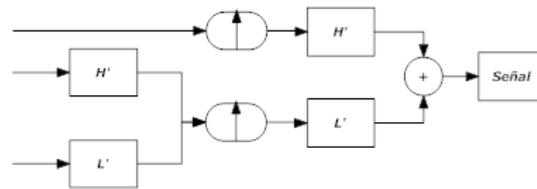


Figura 2. Proceso de reconstrucción.

Es posible reconstruir la señal original sin perder información de los coeficientes de aproximación y detalle. Además, es posible reconstruir señales de aproximación y de detalle de sus coeficientes. Por ejemplo, puede reconstruirse una señal de aproximación empleando los coeficientes cA_1 y sustituyendo los valores de cD_1 por ceros. Las señales de aproximación poseen la información de baja frecuencias y por tanto mayor información de cambios a largo plazo. Por su parte, las señales de detalle poseen información de altas frecuencias, que denota cambios a corto plazo.

Partiendo del árbol de descomposición wavelet de la Figura 1, para este trabajo, se reconstruyeron las señales correspondientes a los coeficientes cA_3 , cD_3 , cD_2 y cD_1 obteniendo series que son identificadas en el resto del artículo como A, B, C y D respectivamente. Las funciones wavelet empleadas para realizar la descomposición y reconstrucción de la señales pertenecen a la familia Daubechies (db). Para determinar específicamente la función de la familia db a emplear para cada serie analizada, las señales de aproximación reconstruidas (A) con cada una de las funciones wavelet db son empleadas para entrenar una red totalmente recurrente de 11 neuronas ocultas. Se evalúa la predicción de 56 valores de cada una de la redes y la de mejor ajuste determina la función wavelet a emplear.

C. Arquitectura Propuesta

La arquitectura propuesta en este trabajo posee una capa de entrada, dos capas ocultas y una capa oculta (ver Figura 3). La capa de entrada posee 5 nodos y se conecta directamente a ambas capas ocultas. La primera capa oculta está formada por tres redes totalmente recurrentes. El objetivo de las subredes es aprender las relaciones de tiempo y frecuencia que conforman la señal original. La segunda capa oculta es totalmente recurrente, posee 10 nodos y recibe las entradas de la primera capa oculta y de la capa de entrada. La capa de salida tiene sólo un nodo.

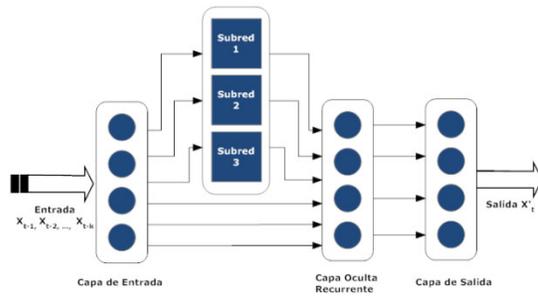


Figura 3. Arquitectura Propuesta.

Cada subred que integra la arquitectura es entrenada con señales reconstruidas mediante la descomposición wavelet. El número de nodos de cada subred es determinado experimentalmente evaluando su capacidad de generalización al predecir los siguientes 56 valores de las series reconstruidas. El entrenamiento de las subredes es realizado mediante el algoritmo de filtro de Kalman extendido basado en aprendizaje recurrente en tiempo real para el cálculo del gradiente (RTRL-EKF) [3]. El entrenamiento es realizado alrededor de 500 épocas.

Para disminuir el número total de neuronas de la red propuesta y con ello el tiempo de entrenamiento, sólo 3 señales reconstruidas son empleadas para entrenar las subredes del modelo. Para elegir las señales a emplear se crean combinaciones de 3 señales reconstruidas, las cuales son sumadas y posteriormente este resultado se resta a la señal original para obtener el MSE correspondiente a dicha combinación. La combinación con el menor MSE es entonces seleccionada.

Después de que las subredes son entrenadas, sus pesos son integrados a la arquitectura propuesta y establecidos como no modificables. Entonces, la arquitectura completa es entrenada alrededor de 50 épocas empleando el algoritmo de RTRL-EKF.

En la fase de predicción, las entradas de la arquitectura en un tiempo actual son las salidas de la red en tiempo previo. Durante este procedimiento no son necesarias entradas externas, excepto por los valores de las condiciones iniciales de la red, las cuales son establecidas durante el entrenamiento. Cabe mencionar que en esta fase, las subredes funcionan como generadores autónomos siendo alimentadas mediante sus propias salidas.

III. PREPROCESAMIENTO DE DATOS

El conjunto de series de tiempo empleado proviene del sitio web de la competencia de predicción NN5 [2]. Dado que las series NN5 poseen datos faltantes, dichos valores fueron sustituidos por la

media de los vecinos inmediatos más cercanos a éstos y cuyos valores son conocidos. Por otra parte, debido a que las funciones de activación de la red están definidas en el intervalo (-1,1), los datos de entrada de la red fueron escalados a este rango para facilitar el aprendizaje. El siguiente escalamiento lineal fue empleado:

$$Z_t = lb + \frac{X_t - \text{mín}(X)}{\text{máx}(X) - \text{mín}(X)} - (ub - lb) \quad (2)$$

donde lb y ub son los límites inferior y superior del intervalo en el que se desea escalar los datos, X_t es el punto de la serie de tiempo X a escalar.

IV. RESULTADOS

Los experimentos se realizaron empleando un subconjunto de 11 series del conjunto de datos NN5. Los primeros 679 datos se emplearon para entrenar las redes neuronales. Los siguientes 56 valores se utilizaron como conjunto de pruebas, para realizar las comparaciones entre modelos de predicción.

Los resultados son comparados empleando dos arquitecturas comunes en la literatura como son redes *feedforward* de tres capas y redes totalmente recurrentes. Ambas redes poseen 5 nodos en la capa de entrada, 26 en la capa oculta y una neurona de salida. Para realizar las comparaciones de error en las predicciones, 12 redes de cada una de las arquitecturas antes mencionadas son entrenadas con cada una de las 11 series empleadas. Para calcular los errores, las predicciones obtenidas por los modelos son re-escaladas linealmente al intervalo original de la serie.

En el mejor de los casos los resultados de la arquitectura propuesta señalan promedios de error y desviación estándar de $\text{MSE} = 31.554 \pm 3.517$. Mientras que las predicciones hechas por las redes *feedforward* y las redes totalmente recurrentes señalan, promedios de error de $\text{MSE} = 240.727 \pm 493.700$ y $\text{MSE} = 224.519 \pm 198.009$, respectivamente. La Figura 4 muestra la mejor predicción de cada arquitectura respecto a los valores reales del conjunto de prueba en el mejor de los casos.

En el peor de los casos los resultados de la arquitectura propuesta señalan promedios de error y desviación estándar de $\text{MSE} = 60.352 \pm 91.831$, mientras que las predicciones hechas por las redes totalmente recurrentes y las redes *feedforward* señalan, promedios de error de $\text{MSE} = 294.838 \pm 108.063$ y $\text{MSE} = 107.870 \pm 64.352$, respectivamente. La Figura 5 muestra la mejor predicción de cada arquitectura respecto a los valores reales del conjunto de prueba en el peor de los

casos. Puede observarse que en el peor de los casos las predicciones de la red propuesta son superiores en promedio a los modelos de redes *feedforward* y redes totalmente recurrentes.

Para ambos casos se empleó la función wavelet *db10* y las señales A, C y D para entrenar las subredes que poseen 11, 8 y 4 nodos respectivamente.

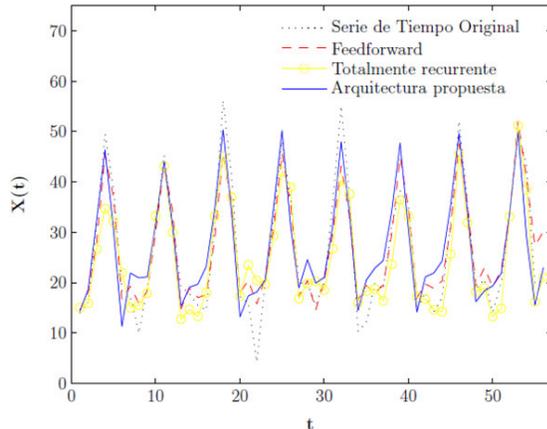


Figura 4. Predicciones de los modelos de predicción en el mejor de los casos.

Los promedios de error obtenidos por los modelos de predicción sobre las 11 series analizadas señalan los siguientes promedios y desviaciones estándar: $MSE = 250.122 \pm 226.046$ de las redes *feedforward*, $MSE = 198.693 \pm 131.121$ de las redes totalmente recurrentes y $MSE = 34.053 \pm 20.116$ de la arquitectura propuesta. Los resultados muestran que las predicciones hechas por la arquitectura propuesta superan ampliamente a los modelos de redes *feedforward* y redes totalmente recurrentes.

V. CONCLUSIONES

En el presente trabajo se mostró una arquitectura neuronal capaz de predecir valores futuros de series de tiempo, basada en subredes capaces de generar señales reconstruidas de coeficientes wavelets. La arquitectura propuesta se comparó con arquitecturas *feedforward* y totalmente recurrentes empleando 11 series de NN5. Los resultados obtenidos señalan que el método propuesto es superior a los modelos clásicos de redes neuronales alimentadas hacia adelante y totalmente recurrentes en la predicción de 56 valores de las series antes mencionadas.

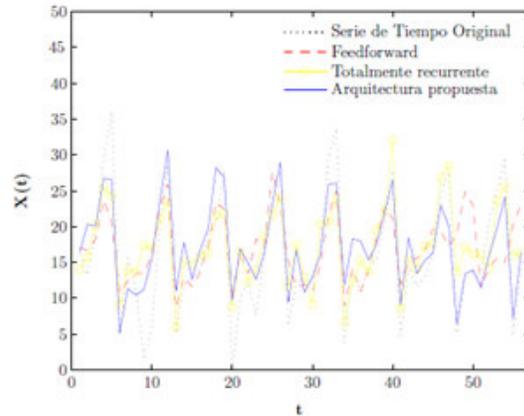


Figura 5. Predicciones de los modelos de predicción en el peor de los casos.

Agradecimientos

El autor le agradece al CONACyT el otorgado a través de la beca # 212427 para estudios de maestría, así como al INAOE por los apoyos recibidos durante la realización de los estudios.

Referencias

- [1] Cai, X., y otros. "Time series prediction with recurrent neural networks trained by a hybrid PSO-EA algorithm". *Neurocomputing*, Vol. 70(13-15), 2007, págs. 2342-2353.
- [2] Crone, S. NN5 forecasting competition for artificial neural networks & computational intelligence. <http://www.neural-forecasting-competition.com>. Última consulta realizada en Junio de 2008.
- [3] Gómez-Gil, P. "The effect of Non-linear Dynamic Invariants in Recurrent Neural Networks for Predictions of Electrocardiograms". Tesis doctoral, Texas Tech University, 1998.
- [4] Gómez-Gil, P. "Long Term Prediction, Chaos and Artificial Neural Networks. Where is the meeting point" *Engineering Letters*. Vo. 15, Number 1. August 2007.
- [5] Haykin, S. "Neural Networks: A Comprehensive Foundation (2nd Edition)". Canada: Prentice Hall, 1998.
- [6] Lendasse, A., Oja, E. y Verleysen, M. "Time series prediction competition: The CATS Benchmark". *Neurocomputing*, Vol. 70(13-15), 2007, págs. 2325-2329.
- [7] Mandic, D. P. and Chambers, J. "Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability". New York, NY, USA: John Wiley & Sons, Inc., 2001.