

# Modelo de redes neuronales CNN-LSTM para el pronóstico de la demanda eléctrica.

Alfredo Rodríguez García  
Gerencia de Análisis de Redes  
Instituto Nacional de Electricidad y Energías Limpias  
Cuernavaca, Morelos, México.  
alfrodga@ineel.mx

**Resumen**—Se presenta un modelo para realizar el pronóstico de la demanda eléctrica. El modelo CNN-LSTM está compuesto por redes neuronales del tipo convolucional (CNN) y por redes neuronales con celdas de Memoria de Largo y Corto Plazo (LSTM). Para mejorar el comportamiento del modelo se optimizan la estructura y algunos hiperparámetros de estas redes. Se presentan resultados para el área Noroeste, y los sistemas Mexicano, Brasileño y Turco. Se consideran periodos de pronóstico de una semana y un mes. Los periodos incluyen días festivos como Semana Santa y Navidad. Se incrementa la dificultad para el modelo al realizar pronósticos en tiempos de emergencia sanitaria debido al COVID-19. Los índices de error obtenidos en los pronósticos indican un comportamiento competitivo del modelo CNN-LSTM.

**Palabras Clave**—Pronóstico de demanda eléctrica; Redes Neuronales Artificiales (RNA); Redes Neuronales con Funciones de Base Radial (RBF); Redes Neuronales Convolucionales (CNN); Celdas de Memoria de Largo y Corto Plazo (LSTM); Optimización de hiperparámetros;

## I. INTRODUCCIÓN

### A. El modelo perceptrón multicapa.

Las Redes Neuronales Artificiales (RNA) son sistemas de reconocimiento que pueden generalizar aprendiendo de ejemplos. Una RNA es esencialmente un grupo de elementos (neuronas) interconectados. Típicamente una neurona calcula la suma de sus entradas (las cuales pueden ser las salidas de otras neuronas o entradas externas al sistema) y pasa esta suma a través de una función no lineal, tal como un sigmoide o un límite. La salida de cada neurona es multiplicada por un factor de ponderación (peso) si esta salida se usa como entrada a otra neurona. El resultado es que hay un peso ajustable para cada conexión entre neuronas. Las RNA se basan en el aprendizaje de las características a través de muestras del problema, extraen las características del mismo y las almacenan en su estructura.

Se utilizan tres conjuntos de datos en diferentes etapas de la creación del modelo ANN [1]: entrenamiento, validación y prueba. Inicialmente el modelo se ajusta utilizando un conjunto de datos de entrenamiento, que es un conjunto de ejemplos que se utilizan para ajustar los pesos de las conexiones del modelo. El modelo se entrena mediante métodos de optimización como un algoritmo de retropropagación (BP), que calcula las derivadas del error con respecto a los pesos de la red. En la práctica, el conjunto de datos de entrenamiento a menudo consta de pares de un vector

de entrada y el vector de salida correspondiente, este se indica comúnmente como el objetivo (o etiqueta). El modelo actual se ejecuta con el conjunto de datos de entrenamiento y produce un resultado. El error de resultado se define como la suma de las diferencias al cuadrado entre el resultado del modelo y el objetivo, para cada vector de entrada. Con base en el resultado de la comparación, se ajustan los parámetros del modelo. Sucesivamente, el modelo ajustado se utiliza para predecir las respuestas de las observaciones en un segundo conjunto de datos denominado conjunto de datos de validación. El conjunto de datos de validación proporciona una evaluación imparcial de un modelo que se ajusta al conjunto de datos de entrenamiento mientras se ajusta el modelo. Se detecta un signo de sobreajuste cuando aumenta el error en el conjunto de datos de validación. Por último, el conjunto de datos de prueba es un conjunto de datos que se utiliza para proporcionar una evaluación imparcial de un ajuste final del modelo en el conjunto de datos de entrenamiento.

Es importante mencionar que para la optimización de los pesos de las conexiones se tiene una superficie de error multidimensional de error; debido a que la RNA está constituida por unidades de procesamiento no lineales, la superficie de error tiene un mínimo global y mínimos locales. El objetivo del BP es comenzar en un punto arbitrario de la superficie de error (determinado por los valores iniciales de pesos) y entonces moverse hacia un mínimo global. Este objetivo no siempre se alcanza, ya que es posible que se quede atrapado en un mínimo local de la superficie de error. En nuestra experiencia [2], el algoritmo BP se puede substituir por un método más efectivo denominado gradiente conjugado escalado (SGC) [3]. Este método elige más cuidadosamente la dirección de búsqueda y el tamaño de paso utilizando información de una aproximación de segundo orden de la función de error en la vecindad del punto actual, esto produce que el método SGC sea un orden de magnitud más rápido que el BP.

### B. El modelo de Redes neuronales artificiales con Funciones de Base Radial (RBF).

Las redes con RBF son un tipo de RNA y se constituyen sólo de 3 capas: una capa de entrada, una capa oculta, que es la que

se encarga de procesar los datos de acuerdo al tipo de función de base radial que tienen las neuronas de esta capa; y una última capa de salida. Para la especificación de los parámetros (centroides, parámetros de dispersión y pesos) de una red RBF se usa un proceso de dos fases. En la primera fase se determinan los primeros dos parámetros, los cuales gobiernan las funciones de base radial que corresponden a cada neurona. Esta fase se realiza mediante la optimización en el espacio de los datos de entrada y se busca que cada neurona de la capa oculta represente una zona diferente de dicho espacio. Para esta selección de centroides se pueden utilizar el método K-medias o un método más efectivo denominado mínimos cuadrados ortogonales (OLS) [4]. El método OLS implica la selección secuencial de los centroides que asegura que tales nuevos centroides elegidos son ortogonales a la selección anterior. En la elección de las mejores funciones de base radial, se mide la contribución de cada función de base radial a la disminución del error. Cada centroide elegido disminuye el error de la salida de la red, y el método se detiene cuando el error alcanza un nivel aceptable de error o cuando el número deseado de centroides han sido elegidos. Para la estimación de los parámetros de dispersión se usa la distancia máxima entre los centroides.

Mientras que en la segunda fase los parámetros de las RBF se mantienen fijos y se determinan los pesos de la capa final. Esto requiere la solución de un problema lineal, utilizándose técnicas de álgebra lineal para obtener esta solución. Existe la posibilidad de tener problemas numéricos debido a que la matriz inversa sea singular o cercana a singular (no invertible). En la práctica, estos problemas se resuelven usando la técnica de descomposición en sus valores singulares (SVD) [5].

## II. MEJORAS A LOS MODELOS DE PRONÓSTICO.

Los dos modelos de pronóstico anteriores, Perceptrón multicapa y la red neuronal con RBF, fueron implementados en un sistema de pronóstico que contiene otros modelos de pronóstico de corto plazo. Este sistema de pronóstico tiene problemas en el manejo de pronóstico a mediano plazo y la consideración de días festivos. Para mejorar el comportamiento de los modelos se buscaron nuevas alternativas, especialmente en el preprocesamiento de los datos de entrada.

Zhao et al [6] proponen un modelo de pronóstico que plantea analizar y procesar grandes cantidades de datos de manera eficiente y superar el problema de la excesiva dimensionalidad. La complejidad del modelo de pronóstico se analiza usando la reducción de la dimensionalidad de los datos históricos de entrada a través del Análisis de Componentes Principales utilizando Kernels (KPCA). Los datos históricos de dimensiones reducidas se ingresan a una red neuronal artificial (RNA), que se optimiza mediante la metodología de enjambre de partículas (PSO). Para ganar experiencia en estos métodos se implementó el método PSO en el proceso de solución de una red neuronal para un problema de clasificación y se probó el método KPCA para la clasificación de 3 anillos concéntricos [1].

En el trabajo de Yan et al [7] utiliza la técnica de máquinas de vectores de soporte (SVM) en sus modelos de pronóstico. Se realizaron ejemplos de la aplicación de las SVM para clasificación (SVC) y regresión (SVR). En la referencia [2] se presenta un ejemplo de optimización de los hiperparámetros de la SVM utilizando la metodología de búsqueda en cuadrícula (GridSearch).

Finalmente se encontró que con el objetivo de poder extraer las entidades de un conjunto de datos se pueden utilizar las redes neuronales convolucionales (CNN). Esta red contiene una o varias capas convolucionales. La convolución permite preservar la relación entre las diferentes partes de una serie de tiempo. La capa convolucional se apoya en otro tipo de capas y funciones, entre ellas están la capa de agrupamiento (pooling), la función ReLu, la capa de aplanado (flatten) y la función acolchonado (padding). En la referencia [2] se presentan un caso de ejemplo de la forma de implementar una CNN para la clasificación de la base de datos MNIST de dígitos manuscritos.

También se encontró que las RNA con demasiadas capas ocultas se tiene la problemática del gradiente que se desaparece o explota [11]. Esta problemática se puede resolver utilizando las unidades de memoria de largo y corto plazo (LSTM). Las LSTM son bastante útiles en las tareas de pronóstico de series temporales que implican autocorrelación, la presencia de correlación entre las series temporales y las versiones rezagadas de sí misma, debido a su capacidad para mantener el estado y reconocer patrones a lo largo de la serie temporal. La arquitectura recurrente permite que los estados persistan o se comuniquen entre las actualizaciones de los pesos a medida que progresa cada periodo. Además, la arquitectura de células LSTM mejoran a las RNA al permitir la persistencia a largo plazo además de corto plazo

## III. RED NEURONAL CONVOLUCIONAL (CNN).

La diferencia fundamental entre una capa densamente conectada de las RNA y una capa especializada en la operación de convolución, que se denomina capa convolucional, es que la capa densa aprende patrones globales de su espacio de entrada, mientras que las capas convolucionales aprenden patrones locales en pequeñas ventanas de dos dimensiones [8].

El propósito principal de una capa convolucional es detectar características en las series de tiempo. Esta es una propiedad muy interesante, porque una vez aprendida una característica en un punto concreto de la serie de tiempo, la puede reconocer después, en cualquier parte de la misma. En cambio, en una RNA densamente conectada tiene que aprender el patrón nuevamente si este aparece en una nueva localización. Otra característica importante es que las capas convolucionales pueden aprender jerarquías espaciales de patrones, permitiendo aprender patrones muy complejos.

Existen series de tiempo donde parece haber poca correlación o relación entre dos datos individuales, a menos que ellos estén cercanos uno del otro. En nuestro caso no son tan grandes las variaciones de demanda en horas continuas y eso hace que exista correlación entre los valores continuos de demanda. Además, existe gran correlación con datos de demanda de la misma hora del día, datos de fines de semana, festivos y estacionales, entre otros. Esto lleva a la idea de usar las *Capas convolucionales* y *Capas de agrupamiento*.

#### A. Capa convolucional.

La capa convolucional es la primera capa donde se extraen las características en nuestros conjuntos de datos. Debido al hecho de que los datos de demanda están relacionados con los datos adyacentes y cercanos, la convolución permite preservar la relación entre las diferentes partes de una serie de tiempo. La convolución consiste básicamente en filtrar los datos con un filtro de datos más pequeño para disminuir el tamaño de los datos sin perder la relación. Se considera como ejemplo una matriz de datos (5x5) y un filtro de (3x3) (tamaño del Kernel igual a 3), estas matrices se muestran en la Figura 1. La convolución de estas dos matrices da como resultado la matriz denominada "mapa característico". El proceso es recorrer la matriz filtro sobre la matriz de datos. Se empieza en la esquina superior izquierda. Posteriormente se desliza a la derecha con un paso preestablecido y llegando al final de la derecha, se desliza hacia abajo. Al posicionar inicialmente la matriz filtro en la parte superior izquierda de la matriz de datos se procede a la multiplicación de los elementos coincidentes y posteriormente la suma de los resultados. Esto se realiza para cada posición resultando el mapa característico.

$$\begin{bmatrix} 7 & 2 & 3 & 3 & 8 \\ 4 & 5 & 3 & 8 & 4 \\ 3 & 3 & 2 & 8 & 4 \\ 2 & 8 & 7 & 2 & 7 \\ 5 & 4 & 4 & 5 & 4 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 6 & -9 & -8 \\ -3 & -2 & -3 \\ -3 & 0 & -2 \end{bmatrix}$$

Figura 1 Operación de convolución. Se convolucionan la matriz de datos (5x5) con el filtro (kernel) (3x3) dando como resultado el mapa característico.

#### B. Agrupamiento de valores máximos (Max pooling).

Al construir una CNN, es común insertar capas de agrupación después de cada capa de convolución para reducir el tamaño espacial de la representación y reducir el recuento de parámetros, lo que reduce la complejidad computacional. Además, agrupar capas también ayuda con el problema de sobreajuste. Básicamente, se selecciona un tamaño de agrupación para reducir la cantidad de parámetros seleccionando los valores máximos, promedio o de suma dentro de estos datos. La Figura 2 muestra el resultado de realizar una agrupación de valores máximos (*Max Pooling*), utilizando una ventana de 4X4 y un paso de recorrido igual a uno. Se busca el valor máximo en la ventana y se coloca como resultado en la posición correspondiente.

$$\begin{bmatrix} -1 & -2 & -3 & 4 & 5 & 6 \\ -9 & -10 & -11 & 12 & 13 & 14 \\ -17 & -18 & -19 & -20 & -21 & -22 \\ -25 & 26 & 27 & 28 & 29 & 30 \end{bmatrix} \Rightarrow \begin{bmatrix} 28 & 29 & 30 \\ 36 & 37 & 38 \\ 44 & 45 & 46 \\ 51 & 53 & 54 \end{bmatrix}$$

-33	34	35	36	37	38
-41	42	43	44	45	46
-49	50	51	51	53	54
-57	58	59	60	61	62

60	61	62
----	----	----

Figura 2 Agrupación de valores máximos usando una ventana de 4X4 y un paso igual a 1.

#### IV. CELDA DE MEMORIA DE LARGO Y CORTO PLAZO (LSTM).

Una celda LSTM se compone de 5 componentes no lineales diferentes, interactuando entre sí de una manera particular [9]. El estado interno de una celda LSTM es modificado sólo a través de interacciones lineales [10]. Esto permite que la información retroceda sin problemas a lo largo del tiempo, con una mejora consecuente de la capacidad de memoria de la celda. La LSTM protege y controla la información en la celda a través de tres compuertas, que se implementan mediante una sigmoide y una multiplicación puntual. Para controlar el comportamiento de cada compuerta, se entrena un conjunto de parámetros con gradiente descendiente, con el fin de resolver una tarea con un valor deseado.

Un esquema de la arquitectura de la celda LSTM se representa en la Figura 3 [11].

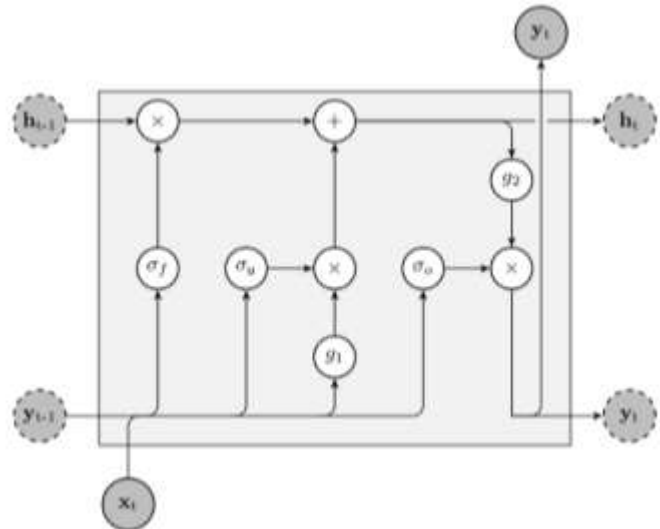


Figura 3 Ilustración de la arquitectura de una celda LSTM.

En esta figura los círculos grises oscuros con una línea sólida son las variables cuyo contenido se intercambia con la entrada y salida de la celda. Los círculos grises oscuros con una línea discontinua representan las variables de estado internas, cuyo contenido se intercambia entre las celdas de la capa oculta. Los operadores  $g_1$  y  $g_2$  son la transformación no lineal, generalmente implementada como una tangente hiperbólica. Círculos blancos con + y  $\times$  representan operaciones lineales, mientras que  $\sigma_f$ ,  $\sigma_u$  y  $\sigma_o$  son los sigmoideos utilizados en las compuertas de olvido, actualización y salida respectivamente. Las ecuaciones que definen los componentes de la celda LSTM se presentan a continuación [10]:

Compuerta de olvido:  $\sigma_f[t] = \sigma(W_f x[t] + R_f y[t-1] + b_f)$ ,

Estado candidato:  $\tilde{h}[t] = g_1(W_h x[t] + R_h y[t-1] + b_h)$ ,

Comp. de actualización:  $\sigma_u[t] = \sigma(W_u x[t] + R_u y[t-1] + b_u)$ ,

Comp. de estado:  $h[t] = \sigma_u[t] \otimes \tilde{h}[t] \oplus \sigma_f[t] \otimes h[t-1]$ ,

Compuerta de salida:  $\sigma_o[t] = \sigma(W_o x[t] + R_o y[t-1] + b_o)$ ,

Salida:  $y[t] = \sigma_o[t] \otimes g_2(h[t])$

Donde  $x[t]$  es el vector de entrada en la etapa  $t$ .  $W_f$ ,  $W_h$ ,

$W_u$ , y  $W_o$  son las matrices rectangulares de los pesos, que se aplican a la entrada de la celda LSTM.  $R_f$ ,  $R_h$ ,  $R_u$  y  $R_o$

son matrices cuadradas que definen los pesos de las conexiones recurrentes. Mientras que  $b_f$ ,  $b_h$ ,  $b_u$  y  $b_o$  son los vectores sesgos. La función  $\sigma(\cdot)$  es una función sigmoide.

Mientras que  $g_1(\cdot)$  y  $g_2(\cdot)$  son funciones de activación no lineales puntuales, generalmente implementadas como tangentes hiperbólicas que aplastan los valores entre (-1, 1).

## V. SISTEMAS DE ESTUDIO.

Para realizar las pruebas de pronóstico se consideran los datos históricos de demanda de cuatro sistemas:

### A. Caso de estudio: Sistema Noroeste.

Se considera al área del Noroeste del sistema eléctrico nacional, utilizando sus datos históricos del 1 de enero de 2010 al 21 de marzo de 2013. La serie consiste en 28,224 intervalos horarios de tiempo. El perfil de la demanda eléctrica del sistema Noroeste se muestra en la Figura 4.

### B. Caso de estudio: Sistema Nacional.

El sistema eléctrico nacional se considera en este caso de estudio. Los datos horarios históricos considerados abarcan del 1 de septiembre de 2000 al 31 de diciembre de 2008. La serie consiste en 73,056 intervalos horarios. El perfil de la demanda del sistema nacional se muestra en la Figura 5.

### C. Caso de estudio: Sistema Brasileño.

El sistema eléctrico nacional de Brasil se considera en este caso de estudio. Los datos para este caso se obtienen de la página del Operador Nacional del Sistema Eléctrico Brasileño [12]. Los datos horarios históricos considerados abarcan del 1 de enero de 2016 al 31 de enero de 2021. La serie consiste en 44592 intervalos horarios. El perfil de la demanda del sistema brasileño se muestra en la Figura 6.

### D. Caso de estudio: Sistema Turco.

El sistema eléctrico nacional de Turquía se considera en este caso de estudio. Los datos para este caso se obtienen de la página de Kaggle [12]. Los datos horarios históricos considerados abarcan del 1 de enero de 2016 al 31 de agosto de 2020. La serie consiste en 40920 intervalos horarios. El

perfil de la demanda del sistema Turco se muestra en la Figura 7.

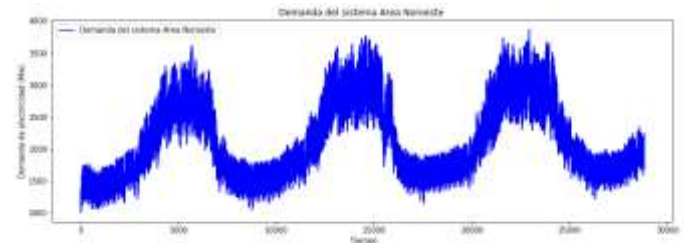


Figura 4 Perfil de la demanda del área Noroeste.

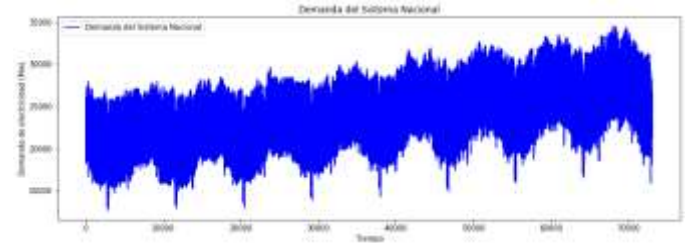


Figura 5 Perfil de la demanda del sistema Nacional.

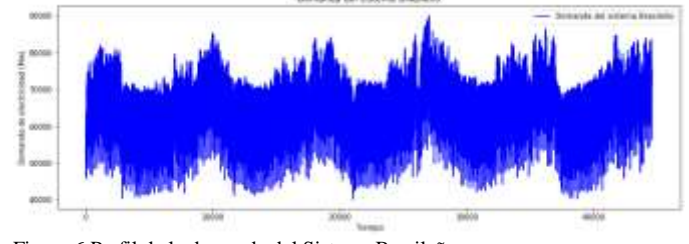


Figura 6 Perfil de la demanda del Sistema Brasileño.

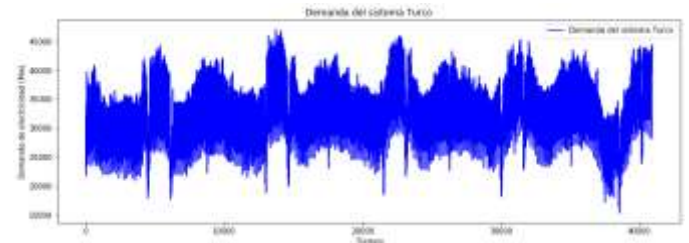


Figura 7 Perfil de la demanda del Sistema Turco.

## VI. RESULTADOS OBTENIDOS.

### A. Índices de error considerados.

Se presentan cinco índices estadísticos para establecer la exactitud general de los pronósticos realizados. Los índices y su formulación matemática son los siguientes:

- Error cuadrático medio (RMSE, Root Mean Square Error).

$$RMSE = \left[ \frac{1}{N} \sum_{i=1}^N (V_{real_i} - V_{pronos_i})^2 \right]^{\frac{1}{2}}$$

- Error cuadrático medio normalizado (NRMSE, Normalized Root Mean Square Error).

$$NRMSE = \frac{RMSE}{(Vreal_{MAX} - Vreal_{MIN})}$$

- Error porcentual absoluto medio (MAPE, Mean Absolute Percentage Error).

$$MAPE = \left( \frac{1}{N} \sum_{i=1}^N \left| \frac{Vreal_i - Vpronos_i}{Vreal_i} \right| \right) * 100 \quad (4)$$

- Desviación media absoluta (MAD, Mean Absolute Deviation).

$$MAD = \frac{1}{N} \sum_{i=1}^N |Vreal_i - Vpronos_i| \quad (5)$$

- Máximo Porcentaje de Error (MPE).

$$MPE = \left( \left| \frac{Vreal_i - Vpronos_i}{Vreal_i} \right| \right) * 100 \quad \forall i$$

Donde  $Vreal_i$  y  $Vpronos_i$  son los valores real y pronosticado para el periodo  $i$ . Además,  $Vreal_{MAX}$  y  $Vreal_{MIN}$  son los valores máximo y mínimo de los valores reales.

### B. Características del modelo CNN-LSTM y optimización de hiperparámetros.

El modelo se implementa en lenguaje Python considerando sus bibliotecas de Keras para la definición de los bloques constructivos de las redes neuronales [14] y Optuna para la optimización de los hiperparámetros [14]. Se realizan pruebas considerando una semana y un mes como periodos de pronóstico. Inicialmente se construye el modelo y posteriormente se busca mejorarlo. En el proceso de mejorar el comportamiento del modelo CNN-LSTM se realiza una optimización de los hiperparámetros y de la estructura del modelo. Los hiperparámetros del modelo considerados y los valores sugeridos se muestran en la Tabla 1.

TABLA 1 VALORES SUGERIDOS PARA LOS HIPERPARÁMETROS DEL MODELO CNN-LSTM.

Kernel_size	= ("kernel_size", [3,5,8])
hp_n_units_LSTM	= ("hp_n_units_LSTM", [16,32,64,128,256])
hp_dense_units	= ("hp_dense_units", [100,200,400])
hp_optimizador	= ("hp_optimizador", ['adam'])
hp_batch_size	= ("hp_batch_size", [16,32,64,128])
hp_epochs	= ("hp_epochs", [10])

El tamaño del Kernel es un hiperparámetro de la red neuronal convolucional. El número de unidades LSTM y unidades de la red densa son de la estructura del modelo LSTM. El tamaño del batch es incluido en el proceso de optimización de las redes neuronales. Cabe señalar que Optuna permite utilizar graficas como las mostradas en la Figura 8 y Figura 9 para ayudar a establecer los valores de los hiperparámetros. Se hace el proceso de optimización para los dos sistemas de estudio y los dos periodos de análisis. Los resultados obtenidos se presentan en la Tabla 2.

Las características de los modelos CNN-LSTM para los cuatro sistemas con un mes de pronóstico se muestran la Tabla 3.

Cabe señalar que en el modelo más grande considerado en este trabajo se entrenan a más de 500 mil parámetros.

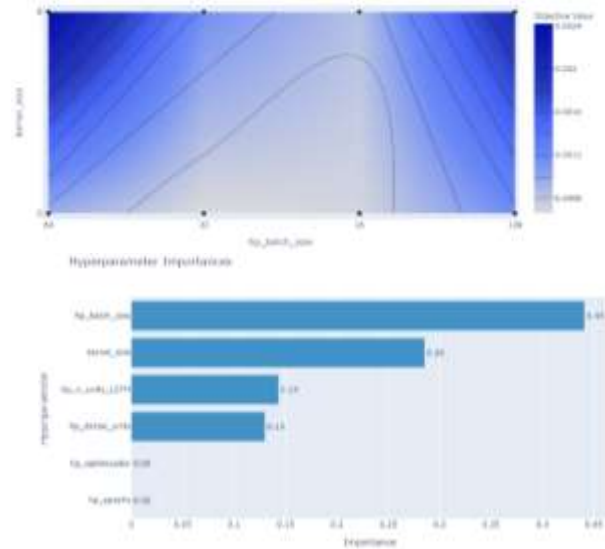


Figura 8 Valores de la importancia de los hiperparámetros para el sistema Nacional para un periodo de pronóstico de un mes.

Figura 9 Gráfica del contorno entre los hiperparámetros hp\_batch\_size y kernel\_size para el sistema Nacional para un mes de pronóstico.

### C. Resultados e índices del comportamiento del modelo de pronóstico.

En esta sección se presentan graficas de los resultados obtenidos. Las demandas reales en las gráficas son representadas por líneas de color rojo y los pronósticos de demanda obtenidos se representan en líneas de color azul.

#### 1) Periodo de una semana (168 horas) de pronóstico.

Inicialmente se considera un periodo de una semana de pronóstico. En la Figura 10 se presentan los pronósticos para el sistema Noroeste teniendo un MAPE de 0.75165% de y un MPE de 3.65854 %.

Para el Sistema Nacional se tiene incluido un día festivo (Navidad) en el periodo de pronóstico. Valores mínimos de demanda se tienen en las Navidades como puede observarse en la Figura 5. Los resultados del modelo CNN-LSTM se muestran en la

Figura 11. Los índices del comportamiento para este caso son MAPE 0.80594% y MPE de 3.15081%. Se puede comentar que estos resultados son muy buenos considerando que este

periodo contenía un día festivo, que usualmente disminuye de manera importante la demanda.

### 2) Periodo de un mes (744 horas) de pronóstico.

Pretendiendo aplicar el modelo CNN-LSTM a un periodo de pronóstico a mediano plazo se extiende a un mes el periodo de análisis. Se analiza el Sistema Noroeste y se consideran los días festivos de Semana Santa en el periodo de pronóstico. Los resultados obtenidos para este caso se muestran en la Figura 12. La Semana Santa está colocada en la parte central del periodo donde se muestra una disminución en la demanda eléctrica en esos días. Como resultado se tiene que el buen funcionamiento del modelo CNN-LSTM no se ve afectado por los días festivos ya que presenta un índice MAPE de 0.59494 % y un MPE de 2.93626%.

El pronóstico de demanda de electricidad para el sistema Nacional considerando un mes como periodo de pronóstico, incluyendo un día festivo (Navidad) se muestra en la Figura 13. El modelo no muestra cambio y presenta buenos resultados otra vez, ahora se tuvieron un MAPE de 0.55963% y un MPE de 2.22891%. Al realizar una comparación de estos índices mensuales con los semanales se visualiza una disminución de los valores obtenidos. Esto es quizás debido al mayor apalancamiento del modelo para un periodo mensual de pronóstico.

### 3) Pronosticos en tiempos de COVID-19 para un periodo de un mes (744 horas) de pronóstico.

En el perfil de la demanda del sistema brasileño se muestra en la Figura 6. Se puede distinguir el abatimiento en el patrón de demanda del año 2020 debido a la contingencia sanitaria del COVID-19. El modelo de pronóstico resultante tiene el número de parámetros que se muestran en la Tabla 3. Utilizando estos parámetros se obtiene el pronóstico de demanda de electricidad para el sistema Brasileño considerando un mes como periodo de pronóstico. El resultado se muestra en la Figura 14

En el perfil de la demanda del sistema Turco que se muestra en la Figura 7, además de poderse distinguir el abatimiento del patrón de demanda en los últimos meses del año 2020, se pueden notar dos variaciones fuertes anuales a la baja. Estas variaciones se deben a dos festejos que se tienen en Turquía:

- o Eid al-Fitr (Fiesta de la Ruptura del Ayuno)
- o Eid al-Adha (Festejo de la Celebración del Sacrificio)

Dado que el último mes de datos históricos es agosto, se tendrá el festejo Eid al-Adha dentro del periodo de un mes de pronóstico de la demanda. Inicialmente en el proceso se crea un modelo de pronóstico que tiene la estructura que se muestra en la Tabla 3. Este modelo de pronóstico con un total de 502,529 parámetros a entrenar. Los resultados de pronóstico de demanda para el sistema Turco, para un mes como periodo de pronóstico se muestran en la Figura 15. Para estos resultados se tiene un máximo porcentaje de error de 3.26%. El conjunto total de los índices del comportamiento del

modelo CNN-LSTM para los seis casos de estudio se muestra en la Tabla 4

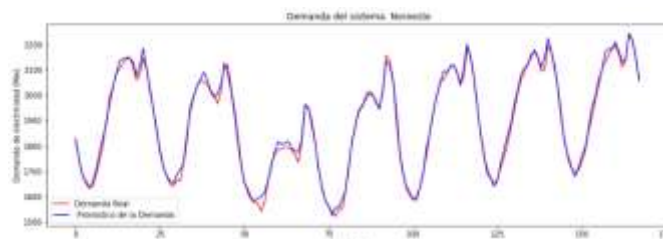


Figura 10 Pronóstico de demanda de electricidad para el Sistema Noroeste considerando un semana como periodo de pronóstico

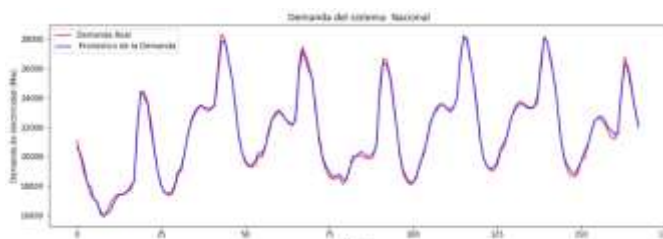


Figura 11 Pronóstico de demanda de electricidad para el Sistema Nacional considerando un semana como periodo de pronóstico con un día festivo.

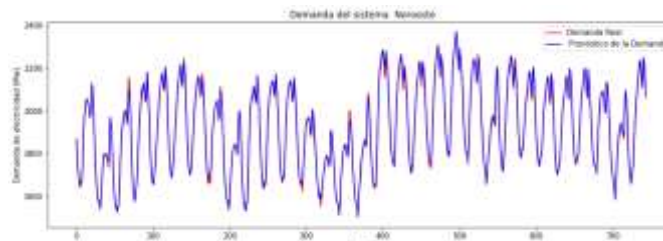


Figura 12 Pronóstico de demanda de electricidad para el sistema Noroeste considerando un mes (incluyendo los días festivos de Semana Santa) como periodo de pronóstico.

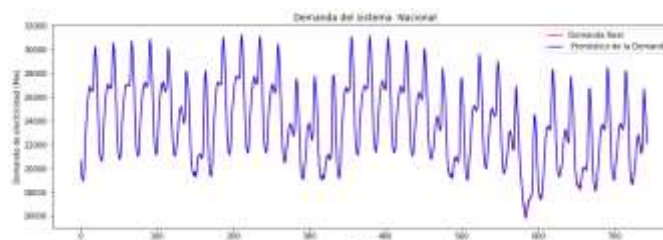


Figura 13 Pronóstico de demanda de electricidad para el sistema Nacional considerando un mes como periodo de pronóstico, incluyendo un día festivo (Navidad).

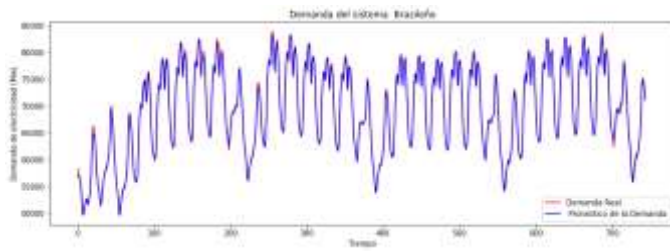


Figura 14 Pronóstico de demanda de electricidad para el sistema Brasileño considerando un mes como periodo de pronóstico, en tiempos de COVID.

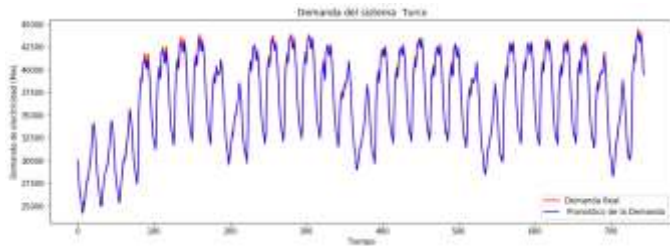


Figura 15 Pronóstico de demanda de electricidad para el sistema Turco considerando un mes como periodo de pronóstico, en tiempos de COVID. Además incluye los días festivos por el Eid al-Adha (Festejo de la Celebración del Sacrificio)

## VII. CONCLUSIONES.

Se ha presentado un modelo CNN-LSTM de pronóstico de demanda eléctrica basado en redes neuronales del tipo convolucionales (CNN) y redes neuronales con memorias de largo y corto plazo (LSTM). Para analizar el comportamiento del modelo se describen y se utilizan los datos históricos de demanda de cuatro sistemas de prueba: Sistema del área Noroeste y Sistemas Mexicano, Brasileño y Turco. Se utiliza una metodología de optimización de la estructura e hiperparámetros del modelo. Se muestran los resultados obtenidos en forma de gráficas y con índices estadísticos del comportamiento del modelo para pruebas de pronóstico de demanda, considerando periodos de análisis de una semana (168 horas) y un mes (744 horas). El modelo CNN-LSTM obtuvo resultados competitivos aún en periodos de pronóstico considerando días festivos: semana santa para el Sistema Noroeste y Navidad para el Sistema Nacional.

Además, el modelo muestra buenos resultados en la más difícil de las pruebas. Esta prueba es realizar pronósticos de demanda en tiempos de contingencia sanitaria debido al COVID-19. En los resultados presentados para un periodo de un mes, se tiene que los resultados del sistema brasileño son muy buenos; y los resultados para el sistema Turco son excelentes dado que el periodo considerado incluye uno de los festejos (Celebración del Sacrificio) del país de Turquía.

Finalmente, aunque el modelo puede utilizar redes entrenadas anteriormente, se sugiere dar pasos en la dirección de paralelizar el algoritmo para reducir los tiempos de cómputo de los procesos de optimización de hiperparámetros y entrenamiento de los parámetros del modelo CNN-LSTM.

## REFERENCES

- [1] [https://en.wikipedia.org/wiki/Training\\_validation\\_and\\_test\\_sets](https://en.wikipedia.org/wiki/Training_validation_and_test_sets).
- [2] Alfredo Rodríguez García, Metodologías de pronóstico a mediano plazo de la demanda eléctrica, Reporte Técnico, INEEL, GAR-HOME\_OFFICE-RPT-PRON\_MP\_DEM\_ELEC, septiembre 2020.
- [3] Martin F. Moller, "A scaled conjugate gradient algorithm for fast supervised learning", Report PB-339, University of Aarhus Denmark, November, 1990.
- [4] S. Chen and S. A. Billings, "The orthogonal least squares algorithm", Section 42 pp. 97-106, Advances in intelligent control, Edited Christopher John Harris, Taylor & Francis Ltd. 1994, ISBN 0-7484-0066-4
- [5] G. Golub and W. C. Reinsch, "Singular value decomposition and least squares solutions", Numerische Mathematik, Vol. 14, No. 5. (29 April 1970), pp. 403-420.
- [6] Zhao Liu, Xincheng Sun, Shuai Wang, Mengjiao Pan, Yue Zhang, and Zhendong Ji, Medium-term demand forecast model based on the analysis of the principal components of the Kernel and the inverse propagation neural network with optimization of particle swarm. Big Data. Volume: 7 Issue 2: June 2019 pp. 130-138. <http://doi.org/10.1089/big.2018.0118>
- [7] Yan X, Chowdhury NA. Mid-term electricity market clearing price forecasting using multiple least squares support vector machines. IET Gener Transm Dis. 2014;8:1572–1582.
- [8] [https://en.wikipedia.org/wiki/Convolutional\\_neural\\_network](https://en.wikipedia.org/wiki/Convolutional_neural_network)
- [9] [https://en.wikipedia.org/wiki/Long\\_short-term\\_memory](https://en.wikipedia.org/wiki/Long_short-term_memory)
- [10] <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- [11] Filippo Maria Bianchi, Enrico Maiorino, Michael C. Kampffmeyer, Antonello Rizzi and Robert Jenssen. An overview and comparative analysis of Recurrent Neural Networks for Short Term Load Forecasting. <https://arxiv.org/abs/1705.04378>
- [12] [http://www.ons.org.br/Paginas/resultados-da-operacao/historico-da-operacao/curva\\_carga\\_horaria.aspx](http://www.ons.org.br/Paginas/resultados-da-operacao/historico-da-operacao/curva_carga_horaria.aspx)
- [13] <https://www.kaggle.com/hgultekin/hourly-power-consumption-of-turkey-20162020>
- [14] <https://es.wikipedia.org/wiki/Keras>
- [15] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta and Masanori Koyama, Optuna: A Next-generation Hyperparameter Optimization Framework, Masanori, Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019

TABLA 2 MEJORES VALORES ENCONTRADOS PARA LOS HIPERPARÁMETROS DEL MODELO CNN-LSTM.

	Una semana de pronóstico		Un mes de pronóstico			
	Sistema Noroeste	Sistema Nacional	Sistema Noroeste	Sistema Nacional	Sistema Brasileño	Sistema Turco
Mejor valor de los intentos:	8.336e-05	1.626e-04	9.604e-05	2.949e-04	2.795e-04	8.292e-04
kernel_size:	5	5	5	5	5	5
hp_n_units_LSTM:	32	128	128	256	256	256
hp_dense_units	100	200	400	400	400	800
hp_optimizador:	adam	adam	adam	adam	adam	adam
hp_batch_size:	16	16	16	16	16	16
hp_epochs:	10	10	10	5	5	5

TABLA 3 RESUMEN DEL MODELO CNN-LSTM PARA UN MES DE PRONÓSTICO. SE MUESTRAN LOS NOMBRES DE LAS CAPAS DEL MODELO JUNTO CON EL NÚMERO DE VARIABLES DE SALIDA DE CADA CAPA. ADEMÁS, EL NÚMERO DE PARÁMETROS A ENTRENAR POR CAPA Y TOTAL.

Capa (tipo)	Sistema Noroeste		Sistema Nacional y Sistema Brasileño		Sistema Turco	
	Número parám.	Forma de salida	Número parám.	Forma de salida	Número parám.	Forma de salida
Input_1 (inputLayer)	0	(1488,1)	0	(1488,1)	(1488,1)	0
conv1d_1 (conv1D)	192	(1484,32)	192	(1484,32)	(1484,32)	192
max_pooling_1 (Max_pooling)	0	(371,32)	0	(371,32)	(371,32)	0
lstm_1 (LSTM)	82,432	(128)	295,936	(256)	(256)	295,936
dense_1 (Dense)	51,600	(400)	102,800	(400)	(800)	205,600
dense_2 (Dense)	401	(1)	401	(1)	(1)	801
Param. Totales:	134,625		399,329		502,529	
Param. Entren.:	134,625		399,329		502,529	
Par. No entren.:	0		0		0	

TABLA 4 ÍNDICES DEL COMPORTAMIENTO DEL MODELO CNN-LSTM.

	Una semana de pronóstico.		Un mes de pronóstico.		Un mes de pronóstico en tiempos COVID-19.	
	Sistema Noroeste	Sistema Nacional	Sistema Noroeste	Sistema Nacional	Sistema Brasileño	Sistema Turco
RMSE (MW):	18.02	217.65	14.56	163.90	447.03	302.72
NRMSE:	0.02492	0.01750	0.01712	0.01068	0.01301	0.01494
MAPE (%):	0.75165	0.80594	0.59494	0.55963	0.50274	0.64753
MAD (MW):	14.05608	171.4163	11.25981	130.53657	348.1833	238.30441
MPE (%):	3.65854	3.15081	2.93626	2.22891	2.55631	3.26018