ISBN 978 -987-633-574-4

Estimación de Temperatura en Servidores mediante Herramientas de Deep Learning

Federico G. D'Angiolo, Ignacio Mas, Juan Ignacio Giribet

Universidad Nacional de Avellaneda, Buenos Aires, Argentina.
Instituto Tecnológico de Buenos Aires (ITBA) y CONICET, Buenos Aires, Argentina.
Universidad de Buenos Aires e Instituto Argentino de Matemática "Alberto Calderón" (IAM) CONICET, Buenos Aires, Argentina.
fdangiolo@undav.edu.ar
imas@itba.edu.ar
jgiribet@fi.uba.ar

Resumen En este trabajo se propone el estudio de estimación de temperatura sobre un servidor, con el objetivo de poder predecir el funcionamiento del mismo bajo condiciones ambientales controladas. Para esto se propone la utilización de herramientas de Deep Learning como por ejemplo, MLP (Multi Layer Perceptron) y LSTM (Long Short-Term Memory). La utilización de éstas persigue el objetivo de poder compararlas y sacar conclusiones sobre su funcionamiento en el ámbito de un datacenter, donde se encuentra el servidor bajo estudio.

Palabras claves: Deep Learning, Redes Neuronales, MLP, LSTM, Datacenter, Temperatura.

1 Introducción

Actualmente, debido al gran avance en el desarrollo de sensores para el muestreo de datos, resulta importante analizar cómo se puede integrar esta tecnología con algoritmos de Inteligencia Artificial (IA). En el caso particular de los Datacenters, es una necesidad controlar la temperatura para que los equipos que se encuentran dentro, puedan trabajar correctamente. Por esta razón, resulta conveniente tener una estimación de la temperatura a la que se encuentra cada servidor, de manera que luego se pueda modificar la ventilación y obtener resultados que permitan cierta estabilidad en las condiciones climáticas del recinto.

En este trabajo se propone usar herramientas de Machine Learning (ML) para predecir la temperatura a la que se encuentra un servidor en particular dentro de un Datacenter. Para poder llevar a cabo este trabajo, se propone sensar la temperatura de un servidor y observar su evolución en función del tiempo para luego, mediante herramientas como MLP (Multi Layer Perceptron) y LSTM (Long Short Term Memory), predecir cómo evoluciona dicha variable un instante después, es decir, en t+1 (una unidad de tiempo después). Este análisis permite pensar en distintas acciones de ventilación sobre el ambiente dado que se puede conocer cómo varía la temperatura una unidad de tiempo

ISBN 978 -987-633-574-4

posterior, la cual, en este trabajo, es de 15 segundos. Las estimaciones obtenidas con cada una de estas redes neuronales se pondrán en comparación para obtener conclusiones sobre su funcionamiento y observar así, cuál de ellas resulta más adecuada para el caso.

La motivación de este trabajo radica en la proliferación de algoritmos de ML relacionados con el procesamiento de series de tiempo ya que permiten obtener resultados aproximados en tiempos relativamente cortos. En particular, en este trabajo, se describe el procedimiento de sensado de temperatura sobre un servidor, el cual va a conformar un dataset o conjunto de datos, que tendrá la forma de una serie temporal pues cada valor de temperatura se encuentra en función del tiempo. Este sensado tiene en cuenta herramientas de IoT (Internet of Things), para el envío de datos a un servidor remoto, el cual almacena la información. Luego, con estos datos, se procede a trabajar con las redes neuronales propuestas (MLP y LSTM) para lograr la estimación.

La distribución de este trabajo se describe de la siguiente forma: en la sección 2 se realiza una introducción a la estimación de temperatura mediante redes neuronales. Luego, en la sección 3 se describe la aplicación al caso de estudio donde se comenta cómo se toman los datos de temperatura del servidor para luego, en la sección 4 mostrar los resultados y concluir con la sección 5 donde se dan las conclusiones.

1.1 Trabajos relacionados

Actualmente la investigación en estimación de variables mediante herramientas de ML se encuentra en gran consideración. Por ejemplo, existen trabajos donde se describe el funcionamiento de una red neuronal para estimar la temperatura en ambientes, es decir, mediante el aprendizaje de esta red se puede conocer cómo puede evolucionar la temperatura dentro de un recinto[1]. Siguiendo en este misma línea, se puede citar trabajos similares donde además de usar redes reuronales para la estimación de temperatura, se añaden algoritmos genéticos [2]. Dentro de esta literatura se pueden encontrar trabajos donde se estudian distintos modelos de ML para la estimación de temperatura, teniendo en cuenta la precisión con que se realiza la misma [3]. Por su parte, si bien obtener predicciones sobre la temperatura mediante herramientas de ML resulta importante, en algunos casos en base a la predicción lograda, es vital tomar decisiones que permitan reducir el consumo de potencia de equipos que generan elevaciones térmicas [4]. A su vez, en lo respectivo a la utilización y comparación de MLP y LSTM, podemos observar trabajos donde se usan estas redes para estimar temperaturas dentro de edificios y conocer cuál de estas tiene mejor rendimiento [5], [6]. Siguiendo con esta comparación, existen trabajos donde se estudia cómo varía el flujo de caja utilizando MLP y LSTM, observando sobre todo los métodos de comparación [7]. Dado que muchos de estos estudios se basan en el análisis de series de tiempo, es importante estudiar cómo se pueden utilizar las LSTM y observar su beneficio [8].

ISBN 978 -987-633-574-4

2 Estimación de temperatura mediante Redes Neuronales.

Los datos de temperatura obtenidos tienen además la información del momento de su captura, es decir, se conoce el día, hora, minutos y segundos de cada valor sensado. Concentrando estos datos en un dataset se puede obtener lo que se llama una Serie Temporal o Serie de Tiempo la cual nos brinda información sobre la evolución de la temperatura a lo largo de un período de tiempo.

Para procesar esta información y poder lograr estimaciones, se utilizan dos redes neuronales con el fin de obtener una comparación y estudiar cuál de ellas resulta mejor en cuanto al objetivo de predicción. Estas redes neuronales son: MLP (Multi Layer Perceptron) y LSTM (Long Short-Term Memory), las cuales se comentan a continuación.

2.1 Multi Layer Perceptron

Un Perceptrón Multi Capa (MLP, Multi Layer Perceptron), resulta ser una Red compuesta por perceptrones conectados entre sí. Estas conexiones forman tres capas, denominadas: capa de entrada, capa oculta (ésta puede estar conformada por varias capas) y capa de salida, las cuales se pueden visualizar en la Fig. 1

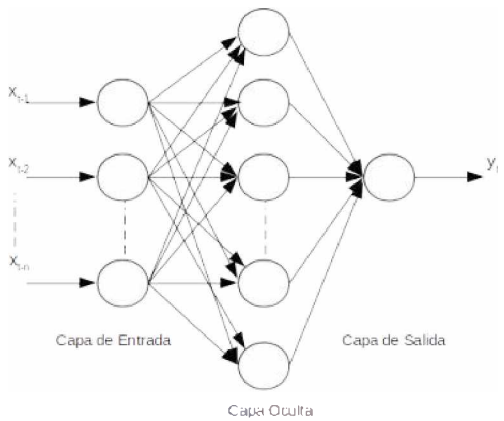


Fig. 1: Perceptrón Multi Capa

La capa de entrada toma como vector de entrada a las muestras de la serie temporal, es decir, si tenemos en cuenta que a la muestra de temperatura actual la denotamos como x_t, a las muestras anteriores las podemos llamar: $[x_{t-1}, x_{t-2}, ..., x_{t-n}]$. Con esta información, la MLP realizará el procesamiento para obtener en su salida, la estimación. Cabe aclarar que la salida de cada perceptrón se puede obtener mediante:

ISBN 978 -987-633-574-4

$$g_i = h(W_i.x + b_i) \tag{1}$$

Siendo:
h = Función de Activación
W_i = Pesos
x = Entradas
b_i = Bias

En el caso de generar una sola estimación, como se describe en este trabajo donde se busca estimar el valor de la temperatura en t+1, la capa de salida consta de un solo perceptrón, cuya salida es $\hat{y}$.

2.2 LSTM

En el caso aquí tratado sobre series de tiempo, cada uno de los valores de esta serie se encuentra relacionado con el anterior, por esta razón, sería conveniente trabajar con redes neuronales que tengan "memoria", es decir, que puedan tener en cuenta el resultado de un estado anterior para luego procesar. Esto resulta importante cuando se trata de sobre estimaciones pues el resultado a predecir está ligado a los valores anteriores. Este tipo de redes se denomina Redes Neuronales Recurrentes (RNN) y se muestran en la siguiente imagen:

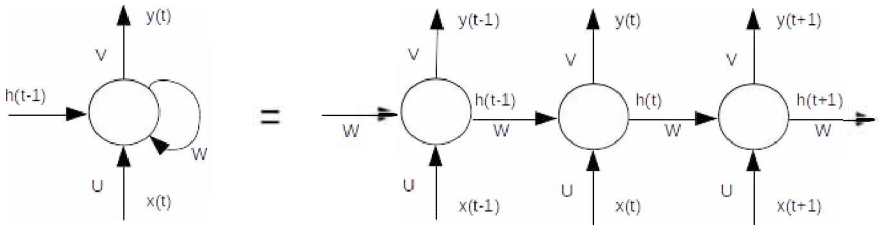


Fig. 2: Red Neuronal Recurrente

En la Fig.2 se observa la entrada a la Red como $x(t)$, la salida mediante $y(t)$ y el estado anterior $h(t-1)$. Cada uno de los nodos que conforma la red, recibe como entrada a cada una de las muestras de x(t).

En base a este sistema, se puede definir entonces el procesamiento, mediante:

$$h_t = \Phi(h_{t-1}, x_t) \tag{2}$$

Siendo:
Φ = Función de Activación.
h_{t-1} = Estado anterior.
x_t = Entradas.

ISBN 978 -987-633-574-4

El cálculo para el entrenamiento de los pesos de este tipo de red neuronal suele ser algo dificultoso, sobre todo cuando las series de tiempo tienen una longitud considerable. Por esta razón se suelen usar las LSTM (Long Short Term Memory), las cuales tienen un funcionamiento similar a las RNN pero su arquitectura se encuentra optimizada para poder solventar el problema de entrenamiento. Esto proporciona cierta robustez frente a las RNN, razón por la cual, suelen usarse este tipo de Redes frente a las Recurrentes, para ciertos problemas.

3 Aplicación al caso de estudio

Previamente al estudio en cuanto a la estimación de temperaturas sobre los servidores, resulta conveniente describir cómo se realiza el sensado de temperatura sobre dicho servidor. Para esto, en la Fig. 3, se muestra una vista superior del Datacenter y cómo se distribuyen los distintos servidores.

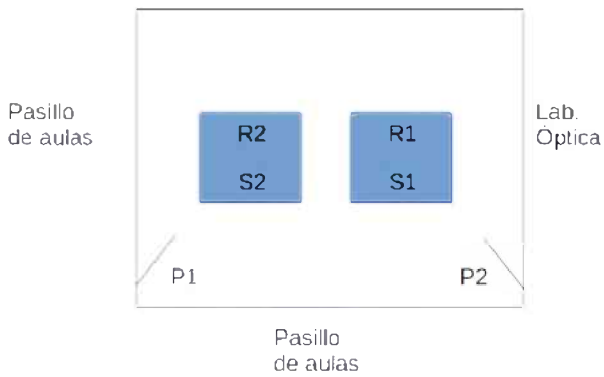


Fig. 3: Layout del Datacenter

La distribución espacial que se propone en la Fig. 3, tiene en cuenta a los Racks denotados mediante R_1 y R_2 los cuales alojan un servidor cada uno (S_1 y S_2). Para sensar la temperatura sobre cada servidor, se propone un sistema basado en un módulo WiFi ESP8266 y sensores de temperatura. Este sistema se encarga de tomar la información de temperatura a la que se encuentra el servidor bajo estudio (S_1) y la envía, mediante WiFi, a un servidor externo el cual almacena todos los datos. Como resguardo, los datos de temperatura sensados se envían a los propios servidores del Datacenter (S_1 y S_2).

En esta etapa del estudio, resulta importante tener en cuenta los datos tomados dado que el muestreo no tiene un efecto continuo en todo momento, es decir, muchas veces se producen caídas en la red de WiFi o cortes de luz momentáneos los cuales producen un reseteo del módulo. Al producirse esto último, los sensores emiten valores que no se condicen con el comportamiento que tiene el servidor hasta el momento. Este análisis es muy importante dado que con estos

ISBN 978 -987-633-574-4

datos se conforma el dataset o conjunto de datos el cual resulta ser la entrada a las redes reuronales que se estudian, si este dataset no se encuentra revisado adecuadamente, las redes podrían estimar de forma incorrecta.

3.1 Datos obtenidos

En base a los datos de temperatura tomados de los sensores, se puede obtener el gráfico de la Fig.4, que muestra cómo evoluciona la temperatura del servidor S_1, en función del tiempo.

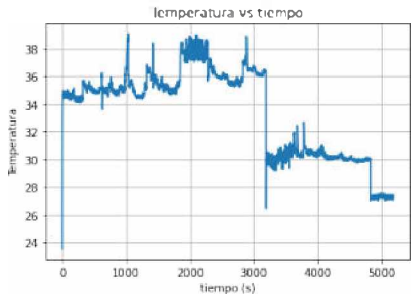


Fig. 4: Temperatura de cada servidor

En la Fig.4 se observan las variaciones de temperatura en ciertos rangos, lo cual demuestra los períodos de mayor y menor actividad del servidor. Por esta razón, resulta interesante utilizar herramientas como redes neuronales para estimar estas variaciones y tener un mejor cuidado de los servidores.

Por último, en base a la Fig.4, cabe aclarar que el muestreo de datos se realiza cada 15 segundos dadas las limitaciones del servidor donde se almacenan los datos.

4 Resultados

En esta sección se muestran los distintos resultados obtenidos a partir del entrenamiento y predicción de cada una de las redes neuronales propuestas. Los resultados obtenidos parten de experimentos realizados sobre la cantidad de muestras pasadas. Con estas y el procesamiento respectivo de cada red, se obtiene la estimación en el momento t+1. Para estos ensayos, se propone entonces configurar a las redes con cantidades similares de parámetros entrenables, iterando con distintas cantidades de muestras pasadas como por ejemplo, 5, 10, 20, 30, etc. Luego, tomando una métrica denominada Error Medio Absoluto, (MAE, Mean Absolute Error), se podrá cuantificar cómo estima cada red neuronal con cada una de las muestras mencionadas.

ISBN 978 -987-633-574-4

A continuación se describe cada detalle de la configuración del hardware, del software, del dataset y las redes neuronales, para obtener y comparar los resultados correspondientes.

4.1 Hardware, Software y Dataset.

En cuanto al Hardware, se utilizó un procesador Intel Core I7 de 2.40GHz con 4 GB de Memoria RAM. Luego, en cuanto al Software, se desarrolló en Python mediante bibliotecas de Keras y Scikit learn.

En base a los datos obtenidos mediante la toma de datos con sensores, se dividió al conjunto en dos partes: una para entrenamiento y otra para validación, siendo esta relación de 70/30, respectivamente.

En cuanto a la configuración de las redes neuronales, se utilizó una cantidad de parámetros entrenables similares en cada una para lograr una comparación. Hay que tener en cuenta que un sistema LSTM contiene mayor complejidad en cuanto a sus parámetros entrenables con respecto a una MLP, dadas las capas internas (ver fig.2). Dada esta situación, se contempla entonces una cantidad de 8749 parámetros entrenables para MLP contra 8506 de LSTM. Luego, en cuanto a la cantidad de iteraciones (epochs), para ambas redes se utilizó un valor de 100.

Por último, para evaluar el desempeño de las redes, se utilizó como métrica al Error Medio Absoluto, el cual viene dado por la siguiente expresión:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{3}$$

Siendo:

N: Cantidad de muestras para la evaluación.
y = Valor de la temperatura.
$\hat{y}$ = Estimación de temperatura.

4.2 Comparación entre Redes Neuronales.

Teniendo en cuenta el hardware, el software y las métricas utilizadas, a continuación se muestran los gráficos de estimación de cada Red Neuronal, esto es: MLP y LSTM. En la Fig. 5 se puede ver la respuesta de la MLP.

ISBN 978 -987-633-574-4

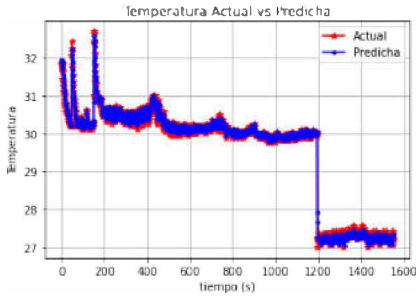
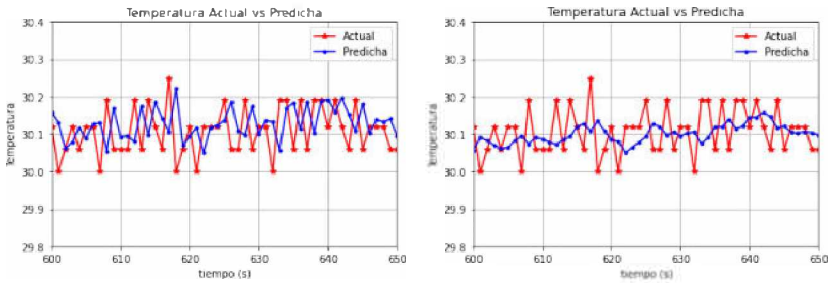


Fig. 5: Temperatura actual vs predicha con MLP

Mediante LSTM se obtiene una estimación similar a la de la Fig.5 sin embargo, dado que no se llega a visualizar cómo se realiza la estimación, a continuación se muestra un mayor detalle de la estimación de cada una de las redes:



(a) MLP (b) LSTM

Fig. 6: Temperatura actual y predicha para MLP y LSTM

Lo que se puede observar de las figuras anteriores es que con LSTM se obtiene una mejor estimación. Si bien las figuras anteriores describen cómo se produce la estimación, en el cuadro que se muestra a continuación (Tabla 1) se expone el valor de MAE para cada red neuronal teniendo en cuenta la cantidad de muestras pasadas. Con esto se podrá determinar cuál de las dos resulta conveniente a la hora de obtener estimaciones de temperatura en este Datacenter.

ISBN 978 -987-633-574-4

MAE		
Cant.Muestras pasadas	MLP	LSTM
5	0.0890	0.0939
10	0.0920	0.0909
20	0.0915	0.0915
30	0.0914	0.0915
40	0.0919	0.0920
50	0.0947	0.0907
60	0.0954	0.0890
70	0.0942	0.0855
80	0.0925	0.0851
90	0.0948	0.0884
100	0.0943	0.0883
120	0.0918	0.0876

Table 1: Tabla de Comparación entre Métricas

En la Tabla 1 se puede observar que, a medida que se incrementa la cantidad de muestras pasadas para la predicción, LSTM tiene un mejor comportamiento que MLP, es decir, el Error Medio Absoluto comienza a decrecer. Para visualizar esto, se propone la siguiente imagen, Fig. 7.

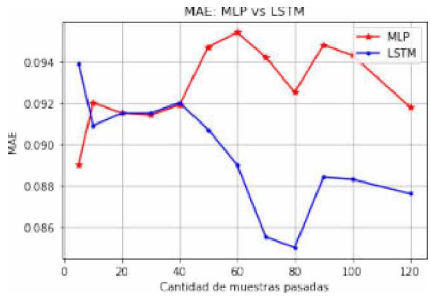


Fig. 7: MAE: MLP vs LSTM

La Fig. 7 muestra cómo evoluciona el MAE para MLP y LSTM a medida que se toma una mayor cantidad de muestras. Esto avala entonces el funcionamiento de LSTM la cual tiene en cuenta el estado previo de cada neurona en el cómputo de estimación.

5 Conclusiones

A partir de las distintas simulaciones teniendo en cuenta la cantidad de muestras pasadas, se puede observar que la mejor alternativa es LSTM. Sin embargo,

ISBN 978 -987-633-574-4

dado que ésta reviste mayor complejidad de cómputo frente a MLP, la misma tiene mayor demora en el tiempo de estimación lo cual concuerda con el modelo planteado. Si bien se plantean estas dos herramientas, existen otras como CNN y GRU que también se pueden comprobar con las mismas u otras pruebas para corroborar su efectividad. Además, existen otras herramientas de estimación como Filtro de Kalman, las cuales resultan de gran interés para su análisis.

Dado que los datos se muestrean cada 15 segundos, la acción de estimar qué sucede en t+1 mediante estas redes, especifica una idea de lo que puede suceder en los próximos 15 segundos. Por esta razón resulta importante estudiar cuál de estas redes tiene una mejor efectividad a la hora de estimar. Sobre estas acciones tomadas se puede ver que a futuro se puede realizar otro tipo de muestreo, tal vez cada 1 minuto, y observar cómo afecta a la predicción, sobre todo teniendo en cuenta que la variación de temperatura en un ambiente no suele ser demasiado rápida.

Por otro lado, se pueden seguir haciendo pruebas en cuanto a la aceleración en Hardware como por ejemplo, la utilización de una GPU, para atenuar estos tiempos de respuesta.

Referencias

1. Qiu Fang., Zhe Li.,Yaonan Wang.,Mengxuan Song., Jun Wang.: A neural-network enhanced modeling method for real-time evaluation of the temperature distribution in a data center. Springer-Verlag London Ltd., part of Springer Nature 2019.
2. Weiping Yu., Zhaoguo Wang., Yibo Xue., Lingxu Guo., Liyuan Xu.: A Combined Neural and Genetic Algorithm Model for Data Center Temperature Control. Published in CIMA@ICTAI 2018
3. Shashikant Ilager., Kotagiri Ramamohanarao.: Thermal Prediction for Efficient Energy Management of Clouds using Machine Learning. SIEEE Transactions on Parallel and Distributed Systems. Volume: 32, Issue: 5, May 1 2021. ISSN: 1045-9219.
4. Yuya Tarutani., Kazuyuki Hashimoto., Go Hasegawa., Yutaka Nakamura., Takumi Tamura., Kazuhiro Matsuda., Morito Matsuoka.: Reducing Power Consumption in Data Center by Predicting Temperature Distribution and Air Conditioner Efficiency with Machine Learning. 2016 IEEE International Conference on Cloud Engineering (IC2E). ISBN:978-1-5090-1961-8
5. Miguel Martínez Comesaña., Lara Febrero-Garrido., Francisco Troncoso-Pastoriza., Javier Martínez-Torresa.: Prediction of Building's Thermal Performance Using LSTM and MLP Neural Networks. 2020 Appl. Sci. 2020, 10(21), 7439; https://doi.org/10.3390/app10217439
6. Kim, T.-Y., Cho, S.: Predicting residential energy consumption using CNN-LSTM neural networks. Energy 2019, 182, 72–81
7. Hans Weytjens., Enrico Lohmann., Martin Kleinsteuber.: Cash Flow Prediction: MLP and LSTM compared to ARIMA and Prophet. 2019. Springer.DOI:10.1007/s10660-019-09362-7
8. Hansika Hewamalage., Christoph Bergmeir., Kasun Bandara.: Recurrent Neural Networks for Time Series Forecasting: Current Status and Future Directions. 2020. Faculty of Information Technology, Monash University, Melbourne, Australia.