

TRABAJO FIN DE MASTER

# Gestión de Bandas de Frecuencias en Entornos Celulares mediante técnicas Predictivas de Deep Learning

---

Autor: Andrés Parra Guirado

Tutores: Dr. José López Vicario y Dr. Xavier Vilajosana Guillén

# Contenidos

---

- Objetivos
- Repetidores de Salto de Frecuencia (FSR)
- Inteligencia Artificial
- Redes neuronales artificiales
- Red LSTM
- Herramientas TIC
- Implementación
- Pruebas
- Resultados
- Conclusiones

# Objetivos

---

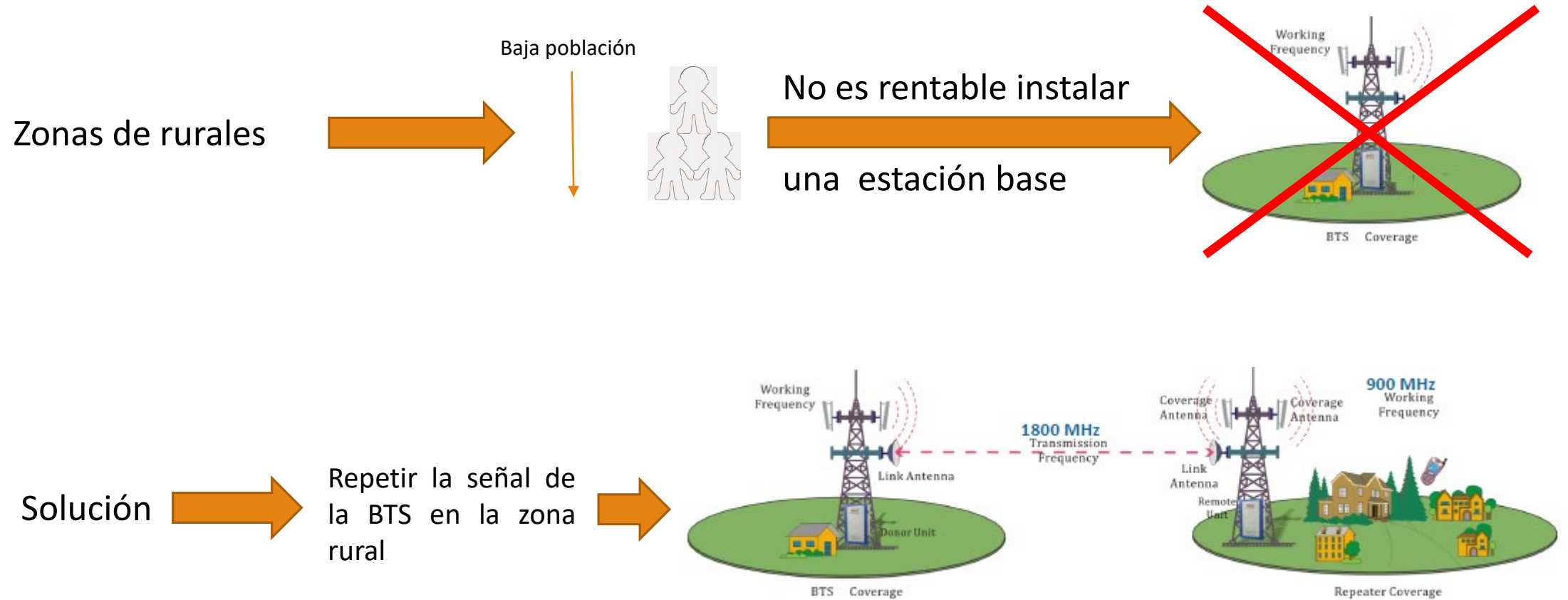
## Generales

- Implementar un modelo capaz de predecir el número de llamadas salientes en una región de la ciudad de Milán.
- Capacidad de elección sobre el tipo de red neural adecuado para la predicción de series temporales.

## Específicos

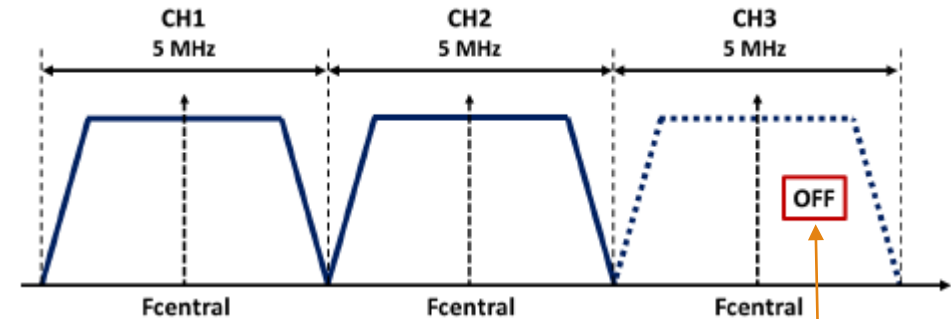
- Capacidad para tratar y extraer información de datos estructurados.
- Capacidad para resolver un problema de predicción de una serie temporal a través de técnicas de Deep Learning.
- **Capacidad para implementar un sistema con un horizonte de predicción amplio, aplicable a la gestión de bandas de frecuencias en entornos celulares.**
- Capacidad de interpretación sobre los resultados obtenidos detectando si los datos de entrenamiento provocan overfitting.

# Repetidores de Salto de Frecuencia (FSR)



# Repetidores de Salto de Frecuencia (FSR)

Por ser más eficientes, los FSRs solo tienen dos canales activos que cubren las llamadas habituales de la población



Conmutación manual

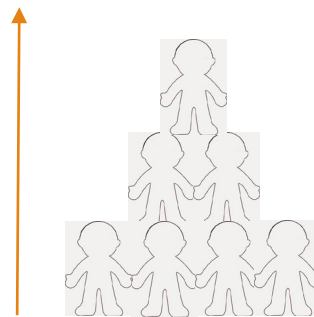
**Problema**

En algunas épocas

la población aumenta



20 – 40 %



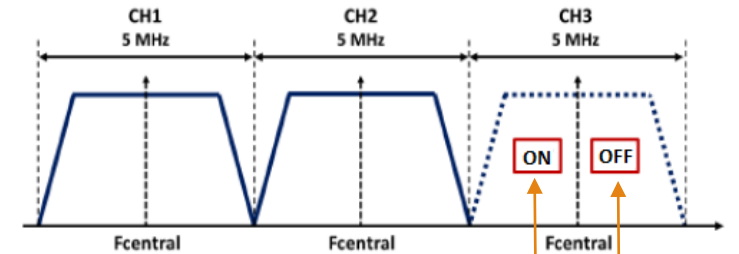
**Si el número de llamadas excede la capacidad de los dos canales, estas no serán cursadas**

# Repetidores de Salto de Frecuencia (FSR) (y III)

Solución



Predicción del número de llamadas mediante técnicas de Deep Learning

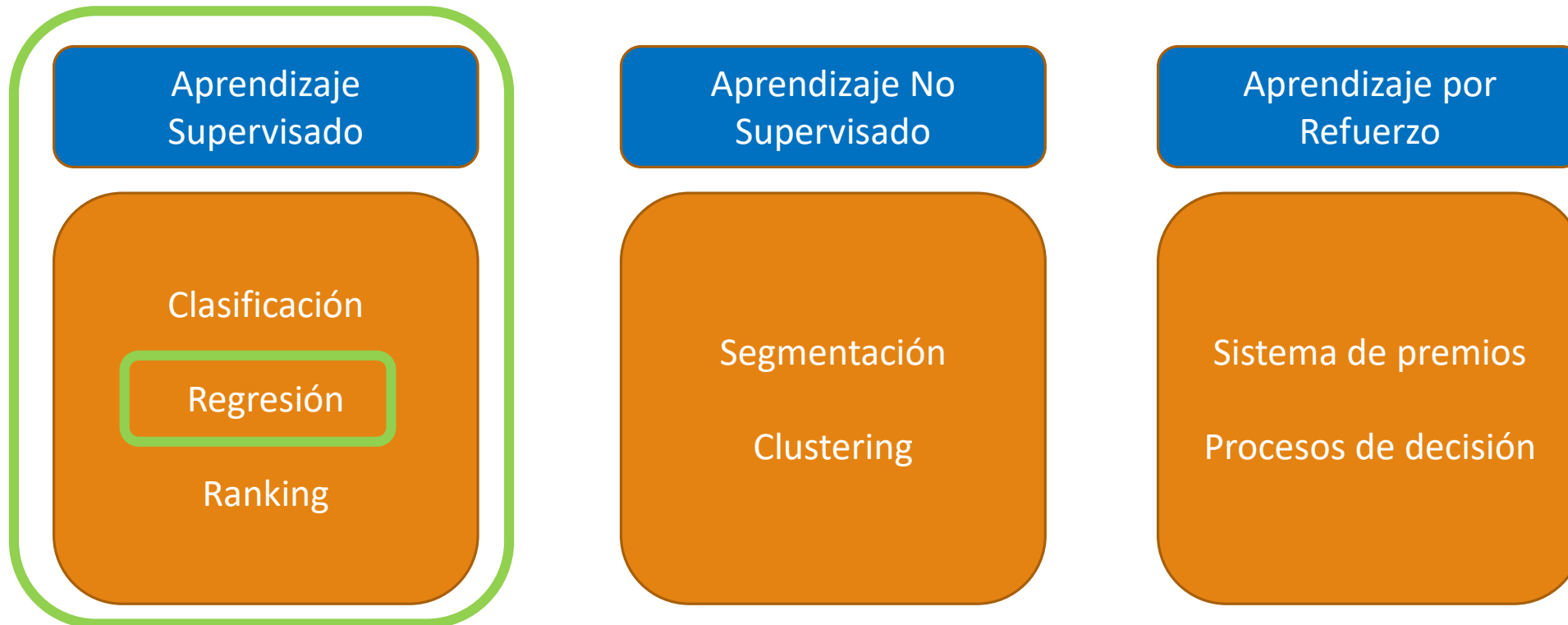


Conmutación automática



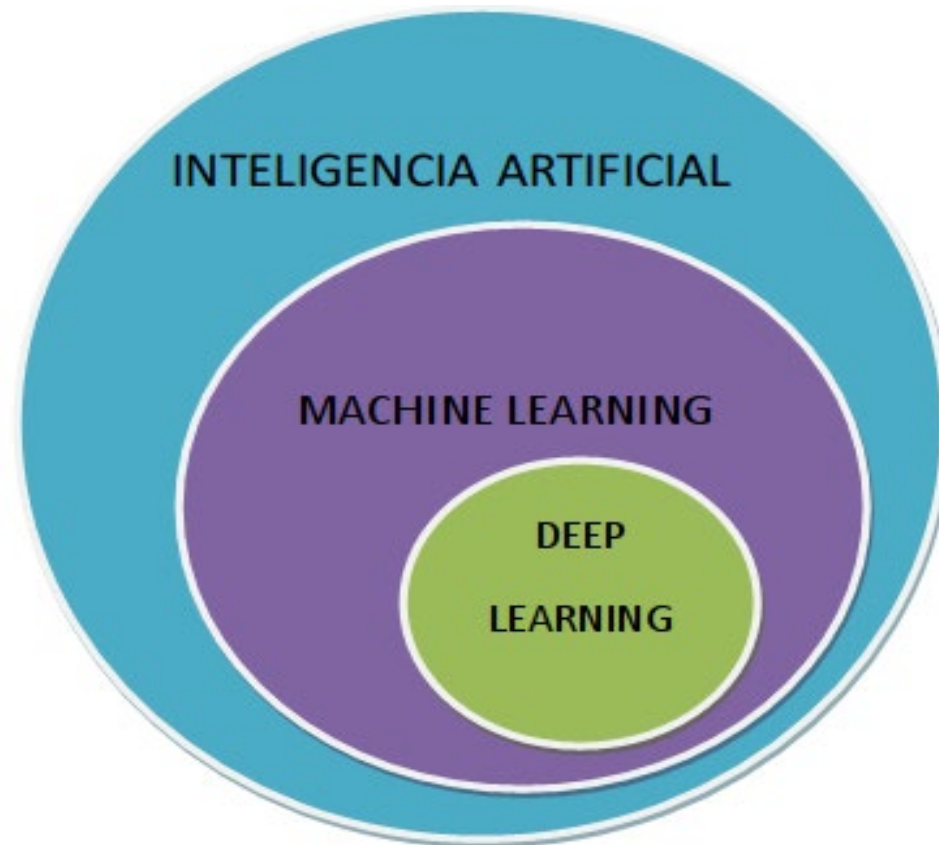
# Inteligencia Artificial. Tipos de aprendizaje

---



# Inteligencia Artificial. Deep Learning (y II)

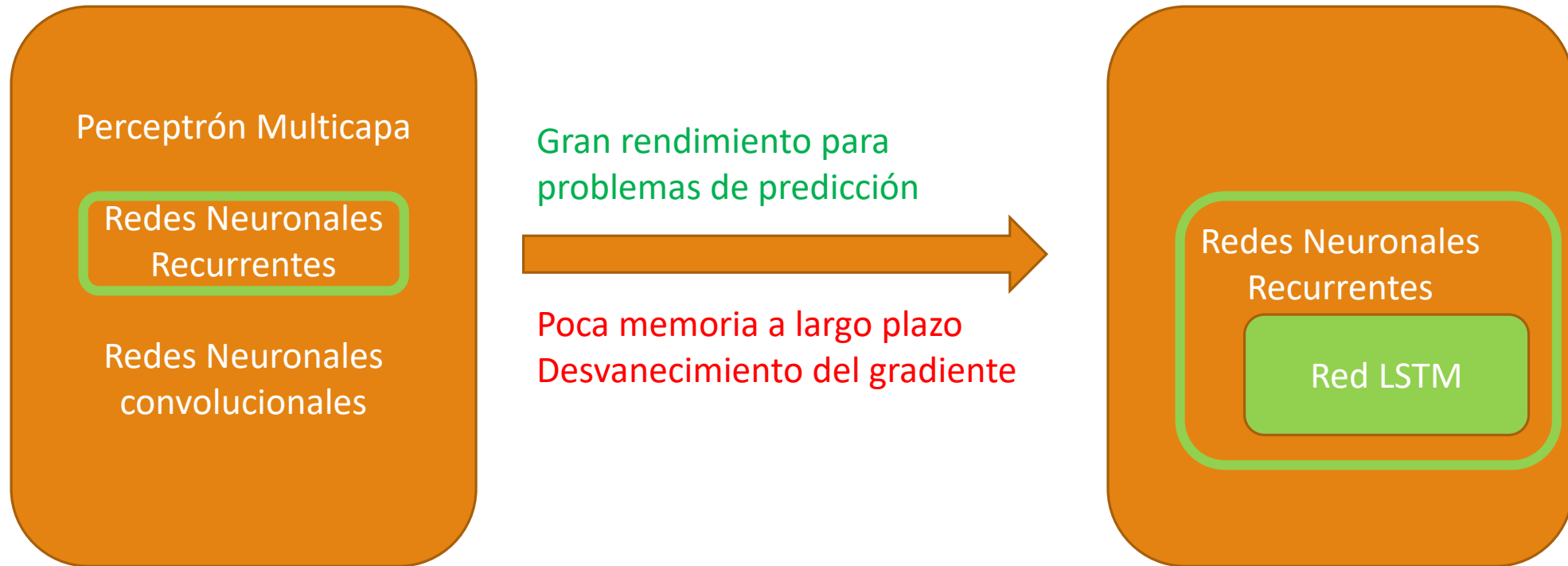
---



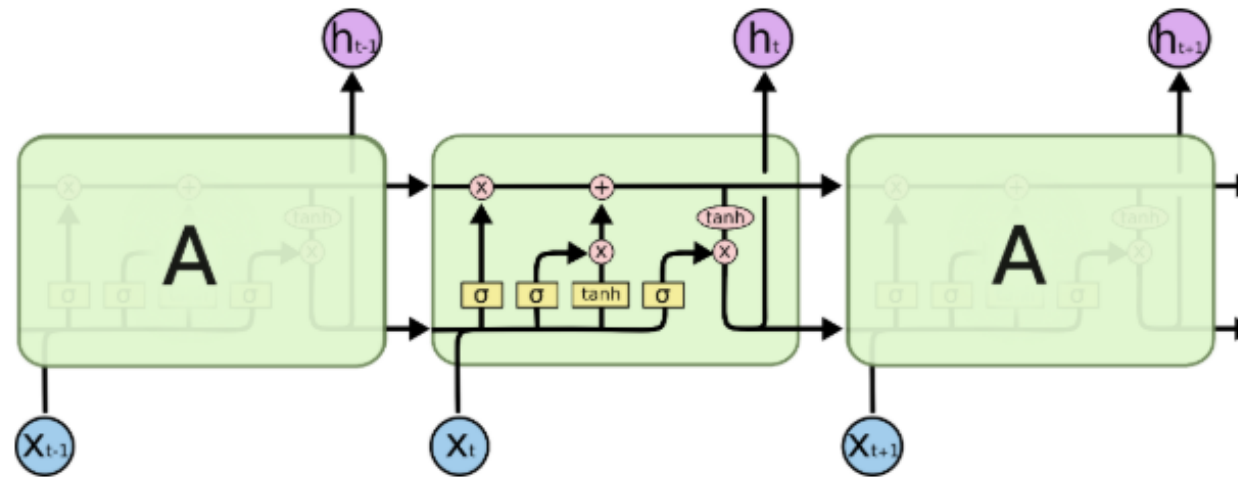


# Redes Neuronales Artificiales

---



# Red Neuronal LSTM



- Soluciona el problema del desvanecimiento del gradiente
- Tiene memoria a largo plazo

- Puerta de entrada  $\longrightarrow i_t = (W_i \cdot [h_{t-1}, x_t] + b_i)$
- Puerta de candidato  $\longrightarrow \tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$
- Celda de estado  $\longrightarrow C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$
- Puerta de salida  $\longrightarrow o_t = \tanh(W_o \cdot [h_{t-1}, x_t] + b_o)$
- Salida  $\longrightarrow h_t = o_t * \tanh(C_t)$

# Herramientas TIC

---



# Herramientas TIC. TensorFlow (y II)

---



- Framework de bajo nivel.
- Ejecución de múltiples arquitecturas.
- Operaciones numéricas a través de diagramas de flujo de datos.
- Las operaciones matemáticas son representadas por los nodos de los diagramas, las aristas muestran tensores.
- Fácil implementación.
- Soporte para GPU.
- Extensa documentación y gran número de tutoriales.

# Implementación. Tratamiento de los datos

9901	9902	...	9999	10000
9801	...	...	9899	9900
...	...	...	...	...
101	102	...	...	200
1	2	3	...	100

Descarga

```

@_msc.cdfviewer en 2017-11-01 0:00:00
Archivo: Archivo: Formato: No. Celdas
1 1813240000 0 0.4017620375125802 1 1813240000 39 0.1312824279242102 0.14079093930436 0.100071
0000 0 0.02713264473752426 1 1813240000 39 0.121131377721214 0.0819370272277 0.4242421
1 1813240000 39 0.4519984877701818 0.404848497471216 0.07974743979848 0.12611610430811 1813240000 0 0.071794
2000421 18132740000 39 0.108928188705611 0.07974743979848 0.12611610430811 18132740000 39 0.128777270154
1 1813240000 0 0.0017974594942675 0.02427
180341151726 0.0548271641387054 0.0514370891442728 0.000795164816254 0.44486951624121 1813240000 0 0.0011714243436662
307517079708 0.0119143147487181 1813240000 0 0.112133099702421 0.0112133099702421 1 1813240000 39 0.137754
200451000 0.1171212912748209 0.0316427617013481 0.104744648108195 11.00014872480121 1813240000 0 0.001318118186461
012164424840621 18132740000 39 0.06127713746615 0.04818110370776 0.1570851212010 0.70242779210421 18132740000 39
0.031274242424062 1 1813240000 0 0.0073192881670183 0.18812668704262 1 1813240000 39 0.140181121212129 0.143484
7024294 1 1813240000 39 1 1813240000 39 0.1747091827018 0.27184207708082 0.704342412427 0.76219709494542
1 18132740000 0 0.146424007704641 1 18132740000 39 0.10612439442154 0.1047848201919176 18132740000 39 0.15221
4007970137741 0.0017073084949176 1 1813240000 39 1.01810310214272 0.12770822442427 0.15617991081219 0.001070497421
12100000 01 0.1884271291970 1 1813240000 0 0.00112133099702421 0.12611610430811 0.12611610430811 0.07179420000 0
811210000 0 0.1884271291970 1 1813240000 42 0.00112133099702421 0.12611610430811 0.12611610430811 0.07179420000 0
20000 39 0.0017073084949176 0.1047848201919176 0.1047848201919176 0.1047848201919176 1813240000 0 0.1047848201919176
1813240000 39 0.0017073084949176 0.1047848201919176 0.1047848201919176 0.1047848201919176 1813240000 0 0.1047848201919176
0.0017073084949176 0.1047848201919176 0.1047848201919176 0.1047848201919176 1813240000 0 0.1047848201919176
5794761001 0.04151046476101 0.178710101010101 0.178710101010101 1813240000 40 0.00112133099702421 0.12611610430811
4127 0.1748049102091 0.120130910272104 0.14189710172915 9.1666470804771 1813240000 0 0.0512121212121212
101812 0.0017073084949176 0.1047848201919176 0.1047848201919176 1813240000 0 0.1047848201919176
1813240000 39 0.1047848201919176 0.1047848201919176 0.1047848201919176 0.1047848201919176 1813240000 0 0.1047848201919176
1800011 0.1047848201919176 0.1047848201919176 0.1047848201919176 0.1047848201919176 1813240000 0 0.1047848201919176
10479478128 18132740000 0 0.00112133099702421 0.12611610430811 0.12611610430811 0.07179420000 0 0.00112133099702421
2001000000 0 0.00112133099702421 0.12611610430811 0.12611610430811 0.07179420000 0 0.00112133099702421
0.05798178131222 1813240000 39 0.00112133099702421 0.12611610430811 0.12611610430811 0.07179420000 0 0.00112133099702421
1757421042 0.00112133099702421 0.12611610430811 0.12611610430811 1813240000 0 0.00112133099702421 0.12611610430811
9 0.12611610430811 0.12611610430811 1813240000 0 0.12611610430811 0.12611610430811
0000 0 0.12611610430811 0.12611610430811 1813240000 15 0.12611610430811 0.12611610430811
000079010077 0.01104110812014 0.01104110812014 1813240000 0 0.12611610430811 0.12611610430811
0 0.12611610430811 0.12611610430811 1813240000 0 0.12611610430811 0.12611610430811
2211481 1813240000 0 0.12611610430811 0.12611610430811 1813240000 39 0.12611610430811 0.12611610430811
0000000000118 1813240000 39 0.12611610430811 0.12611610430811 1813240000 150 0.12611610430811 0.12611610430811
    
```

Archivos con formato texto

Filtrado y agrupación

9901	9902	...	9999	10000
9801	...	...	9899	9900
...	...	...	...	...
101	102	...	...	200
1	2	3	...	100

Área redimensionada a 470 m<sup>2</sup> por celda, que es más acorde con el área de cobertura de una BTS.

Datos recolectados del área de Milán, está dividida en 1000 celdas de 235 m<sup>2</sup>, lo que facilita la comparación entre diferentes regiones

Scikit-learn

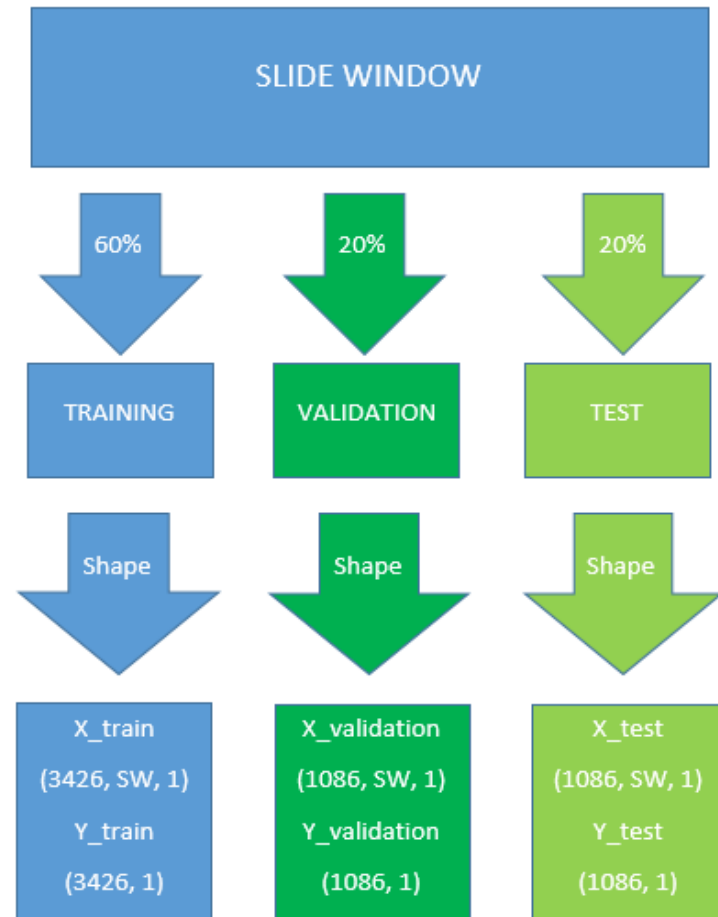
Normalización y escalado de datos

División de datos

Unpickle

Pickle (Cadena de bits)

# Implementación. División de los datos



Ejemplo con horizonte de predicción a 2 muestras vista. SLIDE WINDOW = 3  
Vector de entrada:  $X = [1, 2, 3, 4, 5, 6, 7, 8]$   
El dato predicho es cotejado con  $Y\_data$

X data 1	X data 2	X data 3	Y data
1	2	3	5
2	3	4	6
3	4	5	7
4	5	6	8

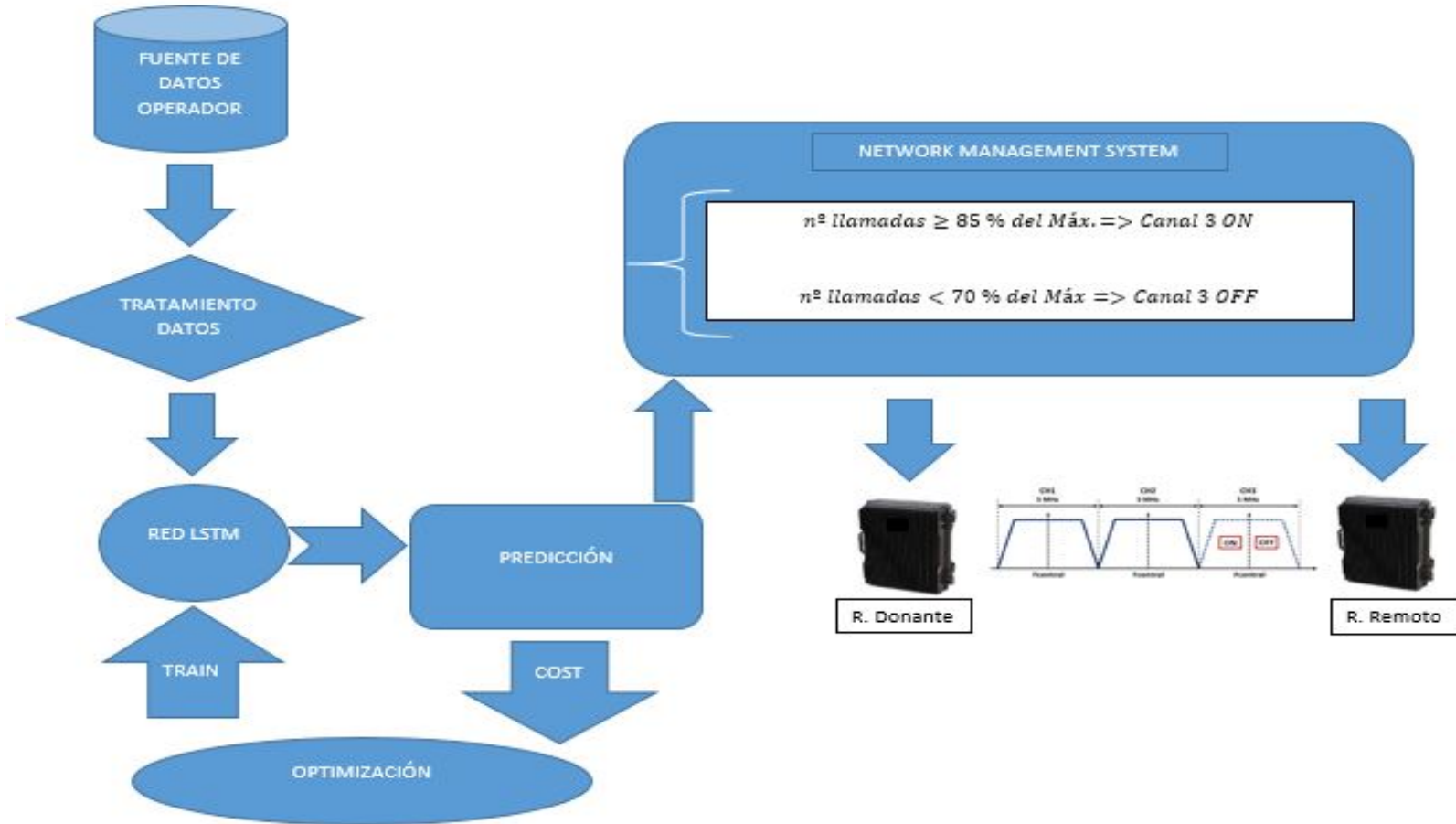
# Funcionamiento

Una vez tratados los datos, el modelo es entrenado hasta conseguir la mejor predicción dentro de un horizonte de predicción que sea aplicable a la aplicación de gestión de los FSRs.

- Si el número de llamadas predichas es mayor o igual al 85 % del número máximo de llamadas habituales cursadas por el repetidor, el canal 3 se activará.
- Si el número de llamadas predichas es menor al 70 % del número máximo de llamadas habituales cursadas por el repetidor, el canal 3 se desactivará.

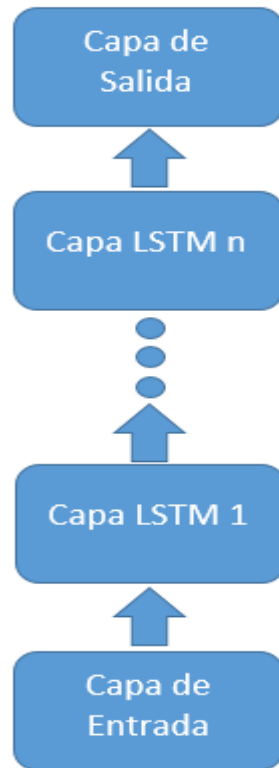
El sistema de monitorización NMS será el encargado de hacer la conmutación.

# Implementación. Modelo completo

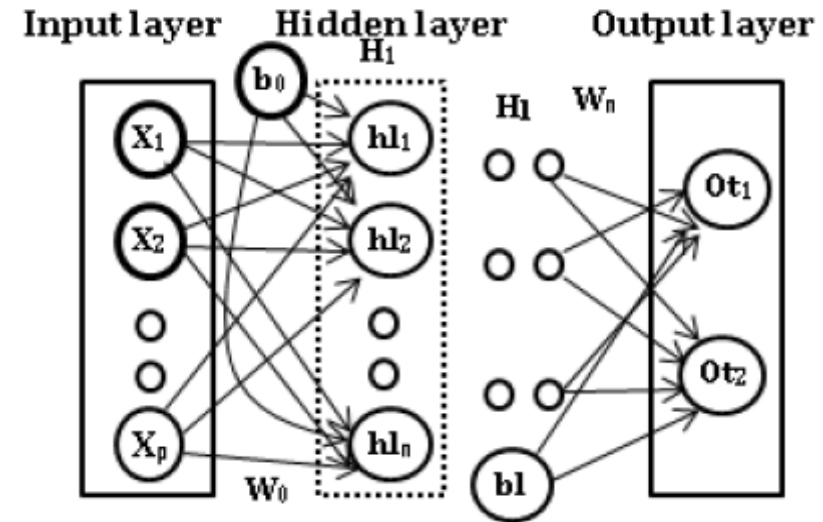


# Implementación. Redes implementadas (y IV)

Modelo LSTM



Modelo Perceptrón multicapa



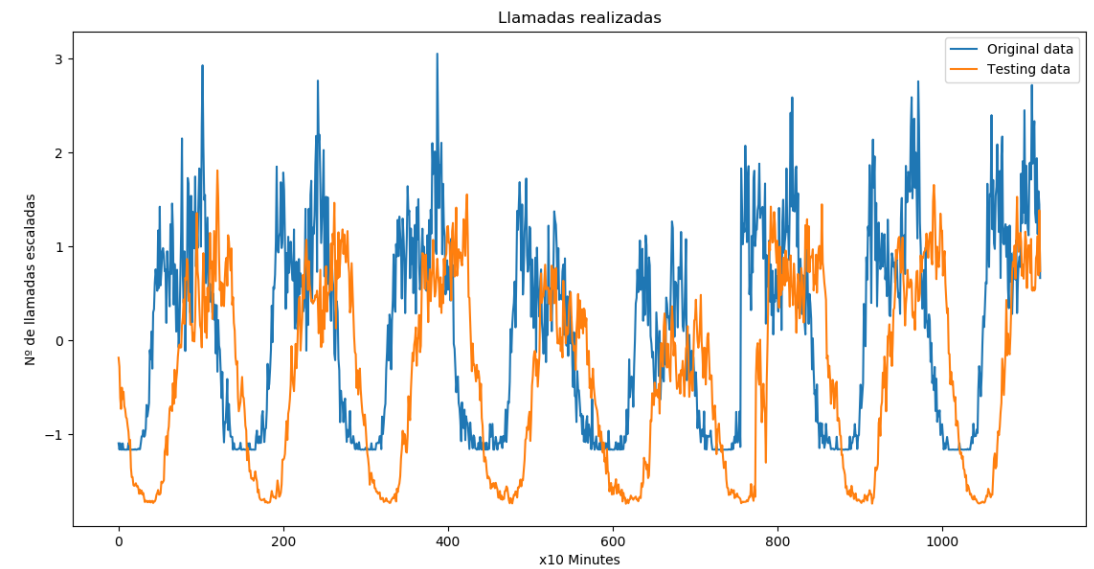
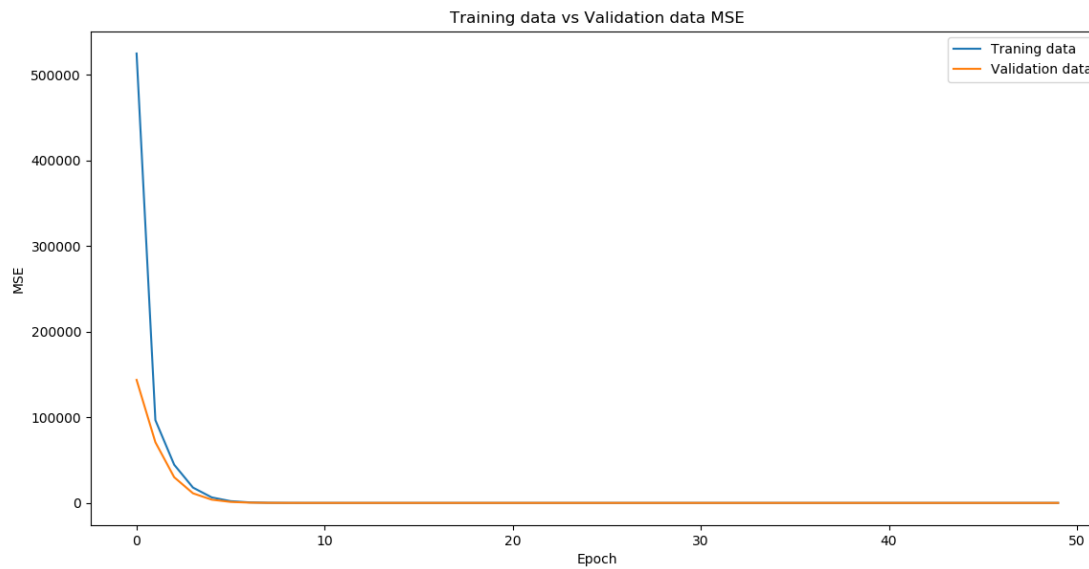
El Perceptrón multicapa ha sido implementado para poder comparar el rendimiento del modelo LSTM



# Resultados. Perceptrón multicapa

Red	Muestras vista	Capas	Nodos	Ventana	Batch	Epochs	L Rate	MSE	MSE Relativo (%)
Perceptrón	18	2	300	20	40	50	0,0005	1,17	87,9

Esta red tiene un error muy elevado para ser utilizada en la gestión de bandas de frecuencias



# Resultados. Red LSTM (y II)

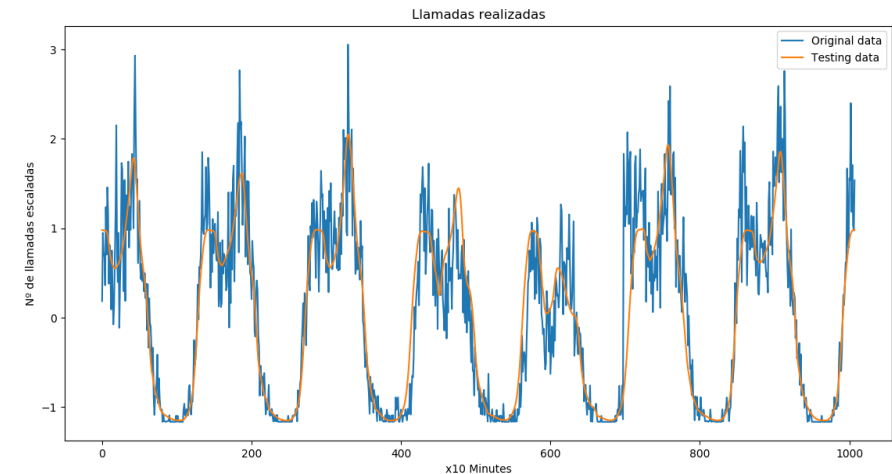
Red	Muestras vista	Capas	Nodos	Ventana	Batch	Epochs	L Rate	MSE	MSE Relativo (%)
LSTM	6	1	32	84	96	70	0,001	0,14	5,9
LSTM	12	2	64	84	168	70	0,0009	0,16	6,94
LSTM	18	3	64	84	168	70	0,0005	0,17	7,12

La red LSTM de 3 capas con un horizonte de predicción de 18 muestras vista (3h.) es una opción viable

Buena evolución de aprendizaje. No hay síntomas de overfitting

Ejemplo para máx. 1000 llamadas cursadas habituales

Llamadas  $\geq 850/1,0712$   
Llamadas error=850-794



Acción	Máx. llamadas	Umbral llamadas (%)	Predicción	$MSE_r$ (%)	Llamadas error	Llamadas reales
Act. Ch. 3	1000	$\geq 85$	3 h.	7,12	56	794
Des. Ch. 3	1000	$< 70$	3 h.	7,12	47	653

$Llamadas reales * 1.0712 \geq \%Máx. llamadas habitual$

$Llamadas reales * 1.0712 < \%Máx. llamadas habitual$

# Conclusiones

---

- Todos los objetivos propuestos se han cumplido.
- Aportación relevante: Predicción de 3 horas.
- Es conveniente formarse sobre una tarea en lugar de investigarla mientras se ejecuta.
- La cantidad de datos para entrenar la red es importante, pero lo es más la calidad de los mismos.

FIN

¡Gracias por su atención!