



Sociedad de Inteligencia Artificial de la
Facultad de Ingeniería
Actividad del curso: Redes Neuronales

Téllez González Jorge Luis

25 de marzo, 2022

1. ¿Cuál es la principal diferencia entre las redes neuronales recurrentes y las LSTM?

- El principio fundamental de las RNN's radica en que, al basarse en el uso de bucles dentro del flujo de información que reciben, estas aprenden de la información obtenida en un paso anterior; permitiendo su reuso para mejorar el funcionamiento del modelo.

Emplea una *memoria* en forma de un estado oculto a_{t-1} y un estado oculto actualizado a_t de forma que en cada iteración a_t ocupará el lugar de a_{t-1} de forma sucesiva.

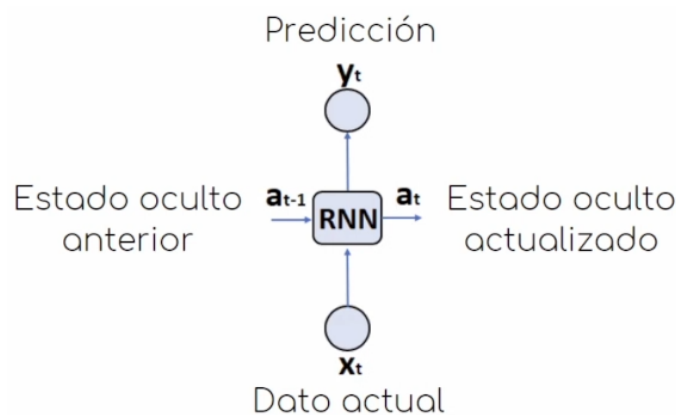


Figura 1: *Estructura básica de una RNN.*

Conforme la red recibe más información, requiere procesarla para predecir una salida en un evento conocido como **dependencia a largo plazo**. Esto tiene como consecuencia que, a medida que se realizan más iteraciones, la información de entrada se va perdiendo con el tiempo, por lo que se habla de una memoria a corto plazo en la red.

- Las redes LSTM, de forma semejante a las recurrentes, emplean bucles de flujos de información. Sin embargo, a diferencia de las primeras, incorpora puertas para el control de la memorización de datos anteriores. Esta característica la hace útil para la manipulación de **series temporales** donde se requiere tener en cuenta los valores de una secuencia sin perder información de forma gradual.

2. ¿En qué tipos de trabajos es más útil el uso de redes neuronales?

En el caso de las RNN, su aplicación básica se encuentra en el procesamiento del lenguaje natural, donde la red debe familiarizarse, por ejemplo, con lo que había antes de una palabra a predecir; como puede ser el caso del autocompletado de Gboard o las sugerencias de correctores de texto como Grammarly.

3. Investiga a qué se refieren los pesos del modelo en la parte del entrenamiento.

Los pesos se refieren a un parámetro dentro de una red neuronal que transforma datos de entrada dentro de las capas ocultas de la red. Una red neuronal se puede visualizar como una serie de nodos o *neuronas*, donde cada nodo es un conjunto de entradas, pesos y un sesgo. Cuando una entrada pasa a un nodo, se multiplica por un valor de peso y la salida obtenida puede pasar a la siguiente capa de la red. La técnica de pesos se emplea para optimizar el funcionamiento de una red neuronal de forma que las predicciones sean precisas; especialmente cuando se tiene demasiada información de un mismo tipo.

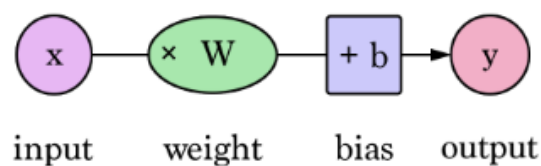


Figura 2: Diagrama de una entrada en un nodo.

4. Escribe todas y cada una de las funciones que intervienen en el diagrama de una red neuronal LSTM.

- **Forget gate:** permite descartar información no útil para el modelo.
- **Update gate:** identifica información útil para añadir a la memoria del modelo.
- **Output gate:** se encarga de crear el nuevo estado oculto.
- **Celda de estado:** permite manipular selectivamente información que se desea mantener o desechar de la memoria.

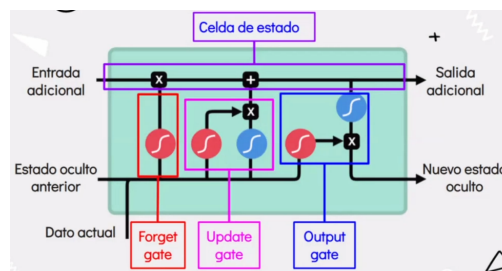


Figura 3: *Diagrama de una red neuronal LSTM.*

5. Investiga la importancia que tienen TensorFlow y Keras en las redes neuronales, además menciona que otro tipo de librerías se llegan a utilizar y en qué ayudan a la red neuronal.

Tensorflow es una plataforma de código abierto enfocada al aprendizaje automático en el entorno de Python. Al proveer de múltiples herramientas, facilita la creación de modelos de forma rápida y con una alta portabilidad local, en navegador o la nube. Nacido como una creación de investigadores e ingenieros del *Google Brain Team*, actualmente es una de las bibliotecas más esenciales en los



entornos del *Machine Learning*.

Por otra parte, *Keras* es un framework de alto nivel que puede trabajar sobre *Tensorflow* y es empleado para facilitar el desarrollo de redes neuronales; como pueden ser aquellas de reconocimiento de imágenes. Tiene como objetivos ser lo más simple posible para lograr que el desarrollador se centre en las partes más importantes del problema a resolver, el ser flexible para que los flujos de trabajo sean lo más rápidos y sencillos y, finalmente, ser una plataforma de alto rendimiento y escalabilidad para múltiples cargas de trabajo.

Existen otras bibliotecas útiles para unir el machine learning con otros campos estrechamente relacionados como la ciencia de datos. *Scikit-Learn* es una de estas bibliotecas, la cual brinda un gran conjunto de herramientas matemáticas, enfocadas principalmente a la estadística, para emplearse en la resolución de problemas de ciencia de datos y potencialmente en conjunto con otras bibliotecas. Su uso puede resultar extremadamente útil al realizar análisis estadísticos de diversos volúmenes de datos.

6. ¿Cuál es el principal problema que tienen las redes neuronales recurrentes y cómo podría solucionarse?, además describe un ejemplo cotidiano de este tipo de problema.

El modelo básico de las redes neuronales recurrentes presenta el problema de que, al tener una memoria de corto plazo, la información de entrada eventualmente se va perdiendo con el tiempo. Por ejemplo, si se tuviese un sistema de predicción de la siguiente palabra que escribirá el usuario, mientras más se intente predecir, se corre el riesgo de que la información de las palabras anteriores a la predicha vaya siendo *olvidada* por el modelo, provocando que las predicciones nuevas eventualmente pierdan sentido o precisión.



7. Del primer ejemplo práctico ¿Con qué comandos se establecen tanto entradas como salidas y el número de iteraciones de aprendizaje?

Para establecer tanto entradas como salidas así como el número de iteraciones (o épocas) se usa:

```
model.fit(entrada, salida, epochs=X)
```

8. ¿Cuál es el efecto que tiene el aumento en la complejidad de la red neuronal en el segundo ejemplo práctico?

Conforme se añaden aún más capas en la red neuronal el gráfico a la salida comienza a mostrar de forma más precisa la ubicación de los 3 grupos de datos de forma más precisa; diluyendo entre cada aumento de complejidad la mezcla entre los distintos tipos de datos (marcados en amarillo, azul y rojo). La mejor distribución y selección de los datos en los 3 grupos se obtiene con un total de 50 capas ocultas.

Bibliografía

DeepAI. (s.f.). *Weight (Artificial Neural Network)*. Consultado el 13 de marzo de 2022, desde <https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network>

Keras. (s.f.). Consultado el 13 de marzo de 2022, desde <https://github.com/keras-team/keras>

Keras. (s.f.). Consultado el 13 de marzo de 2022, desde <https://github.com/keras-team/keras>

Molloy, D. (2006). *STL Introduction*. Consultado el 13 de marzo de 2022, desde <http://www.eeng.dcu.ie/~ee553/ee402notes/html/ch06s03.html>

Scikit-Learn, Herramienta básica para el data science en Python. (s.f.). Consultado el 13 de marzo de 2022, desde <https://www.master-data-scientist.com/scikit-learn-data-science/>



Tensorflow. (s.f.). Consultado el 13 de marzo de 2022, desde <https://github.com/tensorflow/tensorflow>

Los créditos de las fotografías pertenecen a sus respectivos autores. ©