

## **RNN Y LSTM**

**FREDDY ALEJANDRO FLOREZ BOHORQUEZ**

**ORLANDO CORREA MARTINEZ**

**LAURA LOPEZ ARBELAEZ**

**SEMESTRE 7**

**PROFESOR**

**INGENIERO ARIEL RUIZ**

**MATERIA**

**INTELIGENCIA ARTIFICIAL**

**UNIVERSIDAD SANTIAGO DE CALI**

**CALI-VALLE**

**2022**

**TALLER RNN Y LSTM**

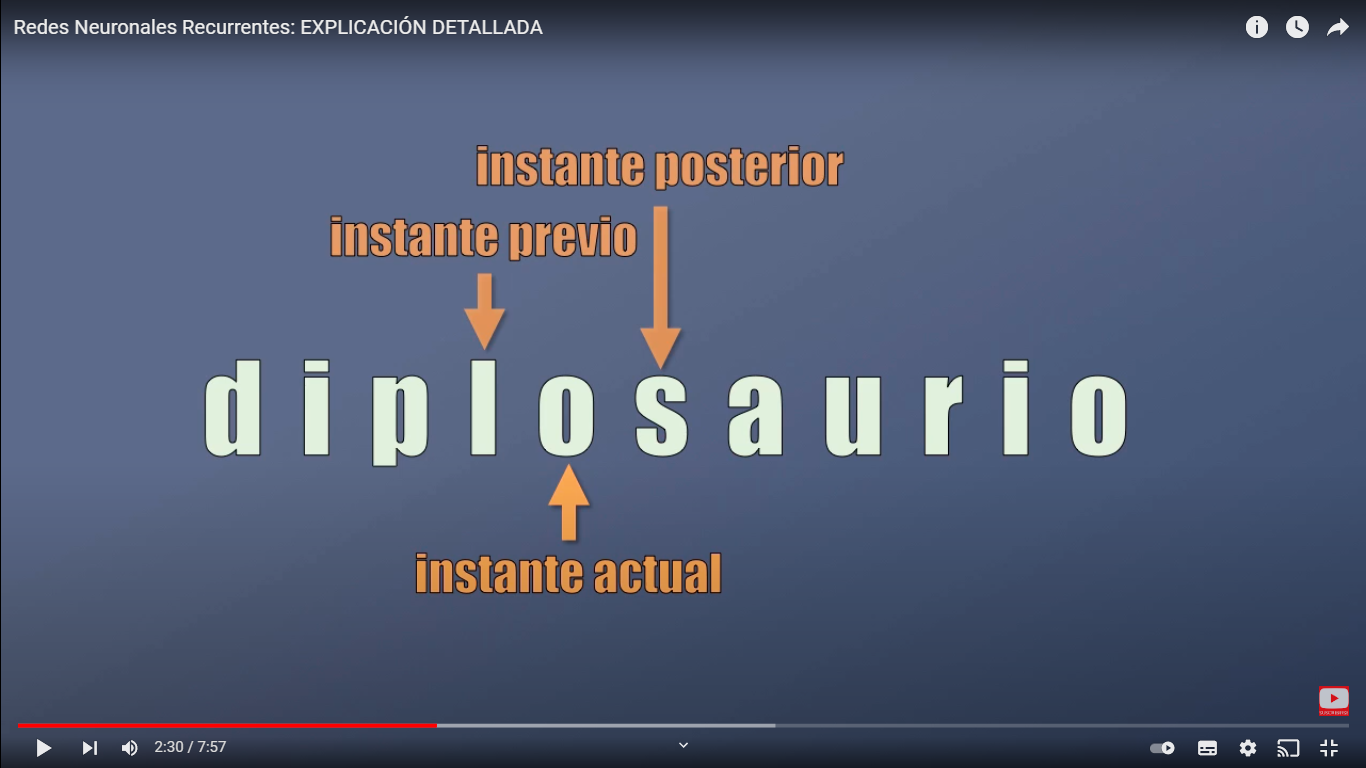
1. **Qué es una RNN, cómo funciona y sus campos de aplicación.**

**R/ Que es una RNN:**

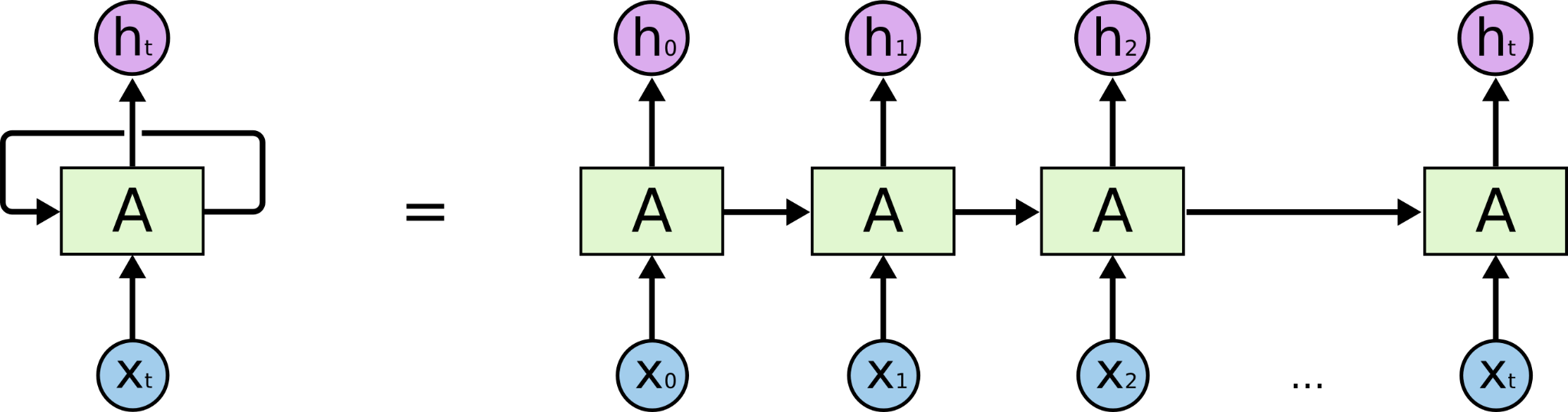
Una red neuronal recurrente es una arquitectura que permite analizar secuencias como conversaciones, textos o videos para ello utiliza el concepto de recurrencia, además de generar un dato de salida, genera el estado oculto que es allí donde esta guarda la memoria de los estados anteriores o las funciones de activación **Hidden State** anteriores o futuros para mejorar el entrenamiento de la red que estos tiempos son conocidos como datos ordinales(previo-pasado, actual-presente, posterior-futuro).

**Cómo funciona:**

En las redes neuronales recurrentes estas se componen de capas de entrada, capa oculta y de salida, Estas redes tienen la capacidad de predecir el evento siguiente gracias a lo que poseen datos recurrentes o funciones de activación anteriores ver figura 1 además de ello se componen de un modelo de entrenamiento como el de las redes neuronales tradicionales de eso tienen que tener en cuenta que la red comparte el mismo peso, sabiendo esto realizaron el proceso de retropropagación y descenso de gradiente esto para el aprendizaje por refuerzo.

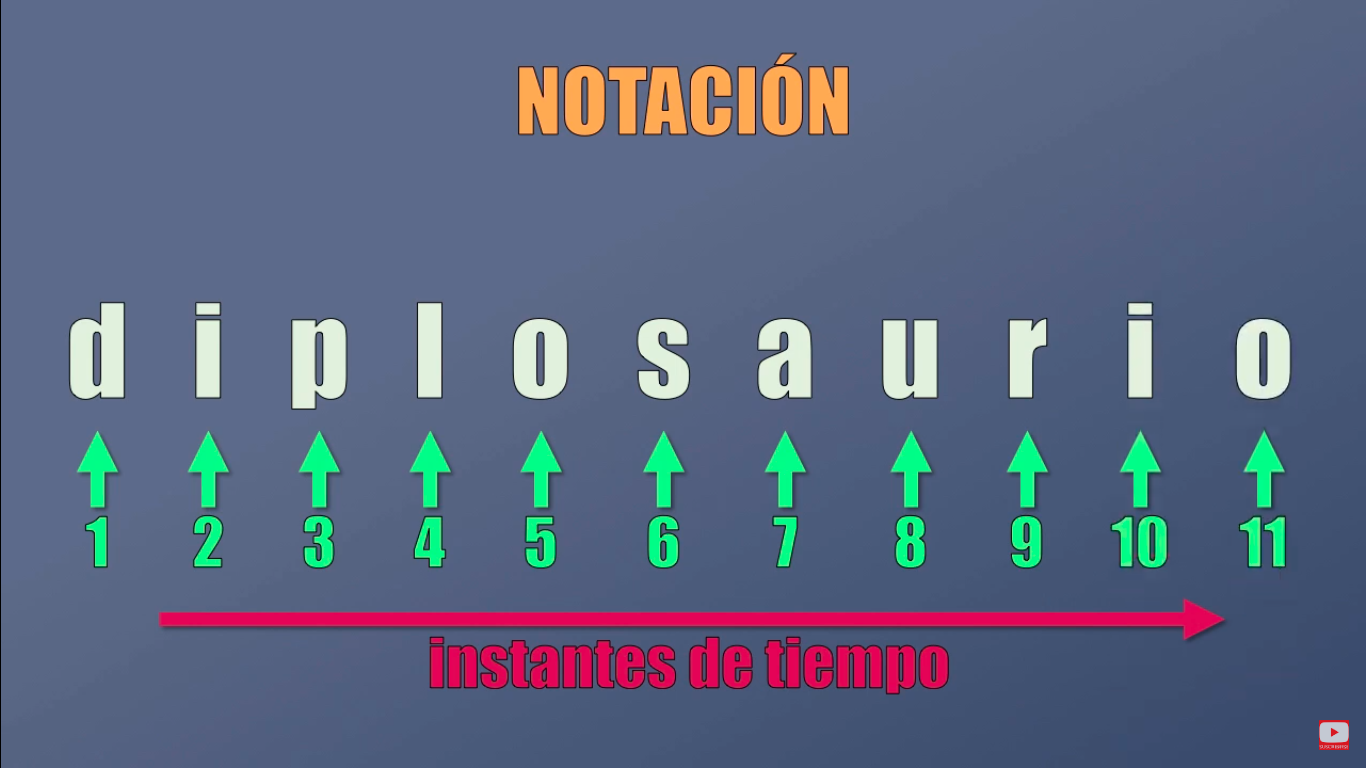


**figura 1: En la imagen se observa los instantes de tiempo en una RNN y se ve como los instante de tiempo antecesor predecesor afectan la generación de palabras (MEMORIA).**



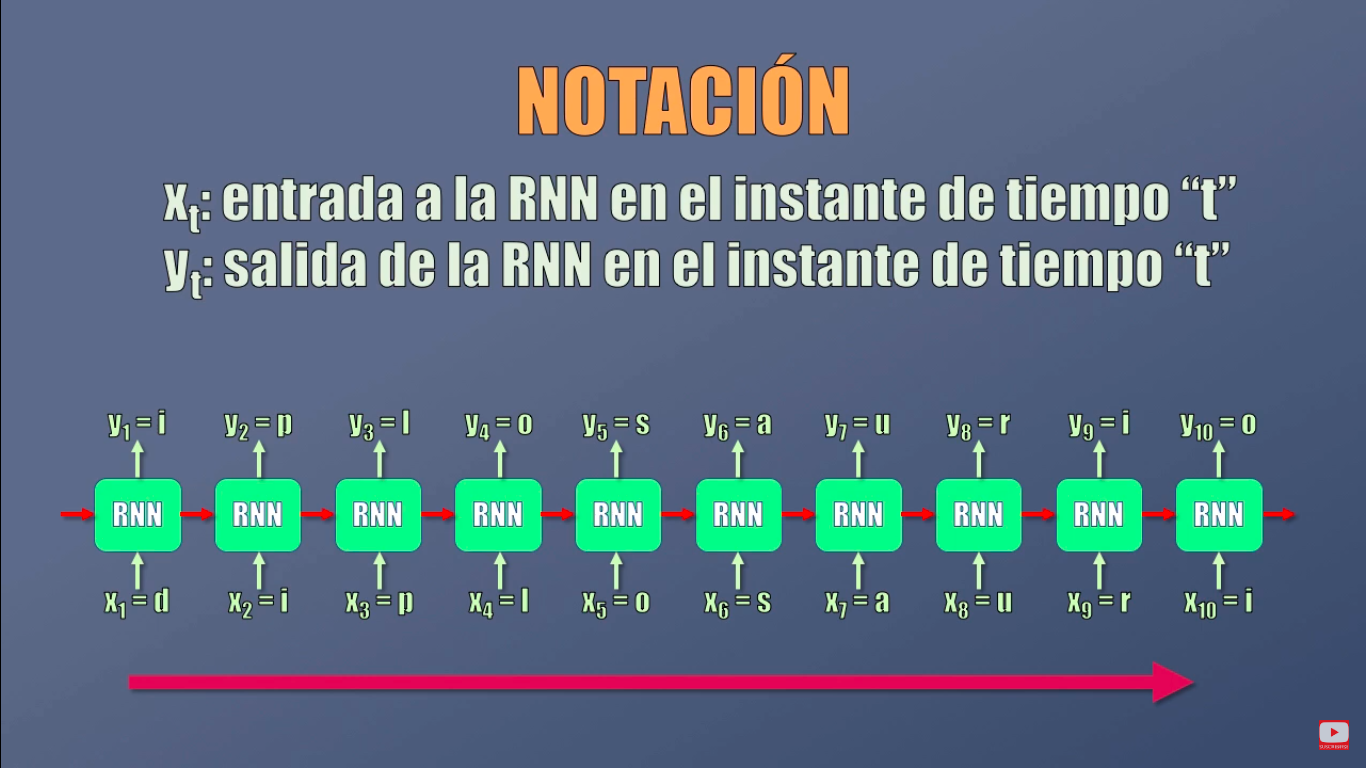
**Figura 2:Estructura de una RNN**

En la **figura 2** se ve como se como se maneja un ejemplo de una frase en la parte de la izquierda sería la parte comprimida de la frase y la de la derecha sería la parte de cómo se representa la parte descomprimida de la frase palabra por palabra además se puede observar las capas que se hablan y en especial la capa oculta que es donde se ve la memoria de la red con las activaciones de la palabra en el instante o estado anterior de la red.



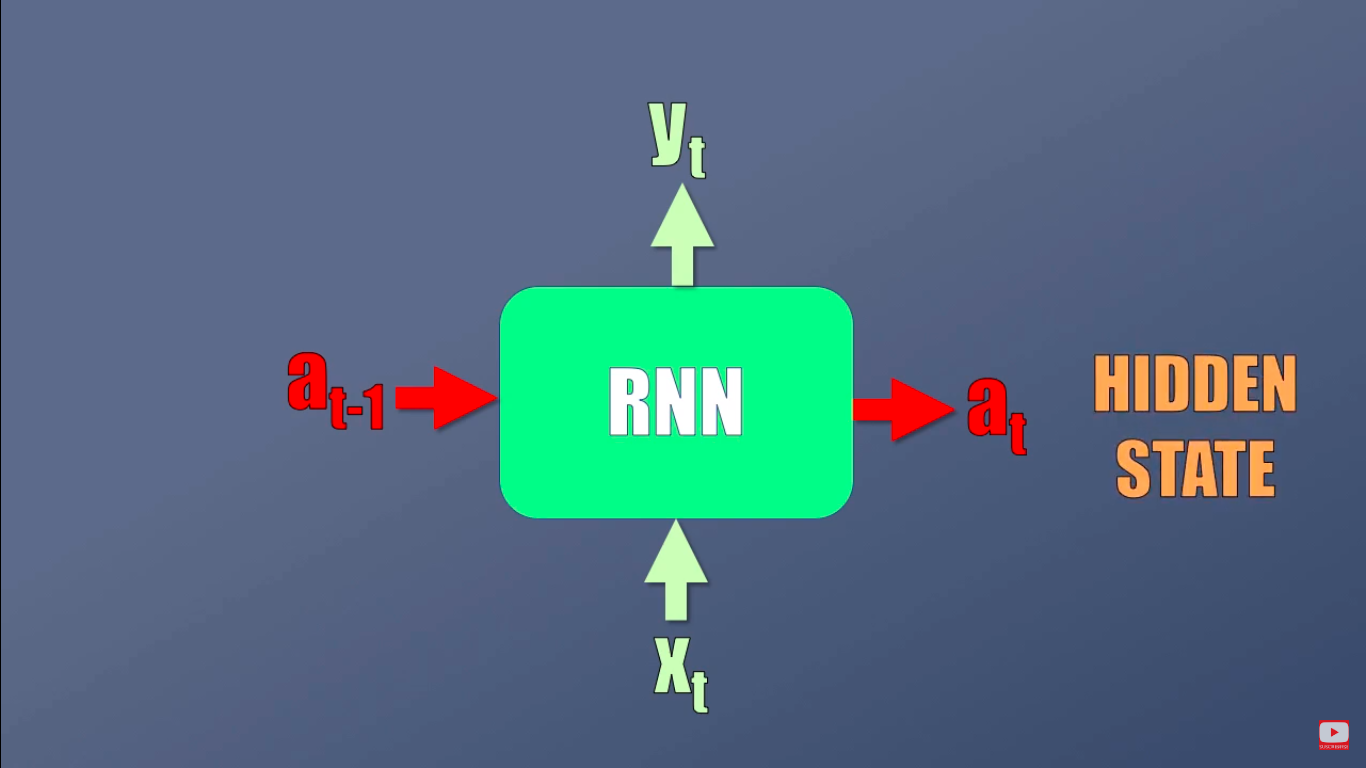
**Figura 3 de los instantes de tiempo que va a tener en cuenta la red Xt**

En la siguiente imagen se observa de manera desglosada lo que ocurre en cad instante de tiempo de Xt y Yt además de ello se puede observar la memoria de la red por medio de las flechas que están en color rojo que él lo que me representan como el estado oculto o la memoria

****

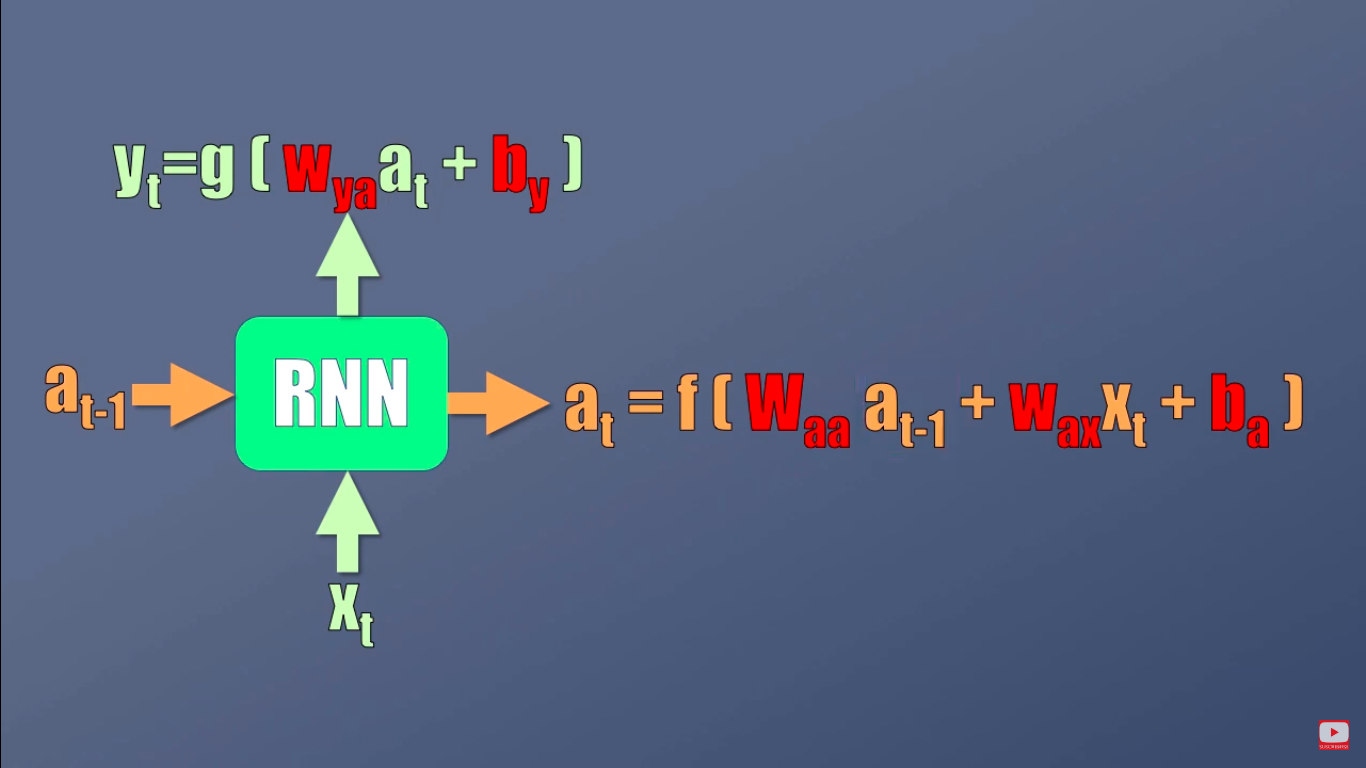
**Figura 3 https://www.youtube.com/watch?v=hB4XYst\_t-I**

Continuando anteriormente con el desglose ahora podemos ver cómo en cada instante de tiempo la red tiene 2 entradas y 2 salidas partiendo con las entradas que una es la del instante de tiempo que es Xt y la otra es la activación del estado anterior que es at-1, mientras que en las salidas es la predicción actual que es Yt y la activación actual que es at que esta activación acoge el nombre **Hidden State** o estado oculto además cabe aclarar que son estas activaciones que corresponden a la memoria de la red puesto que permiten preservar y compartir la información entre un instante de tiempo y otro

****

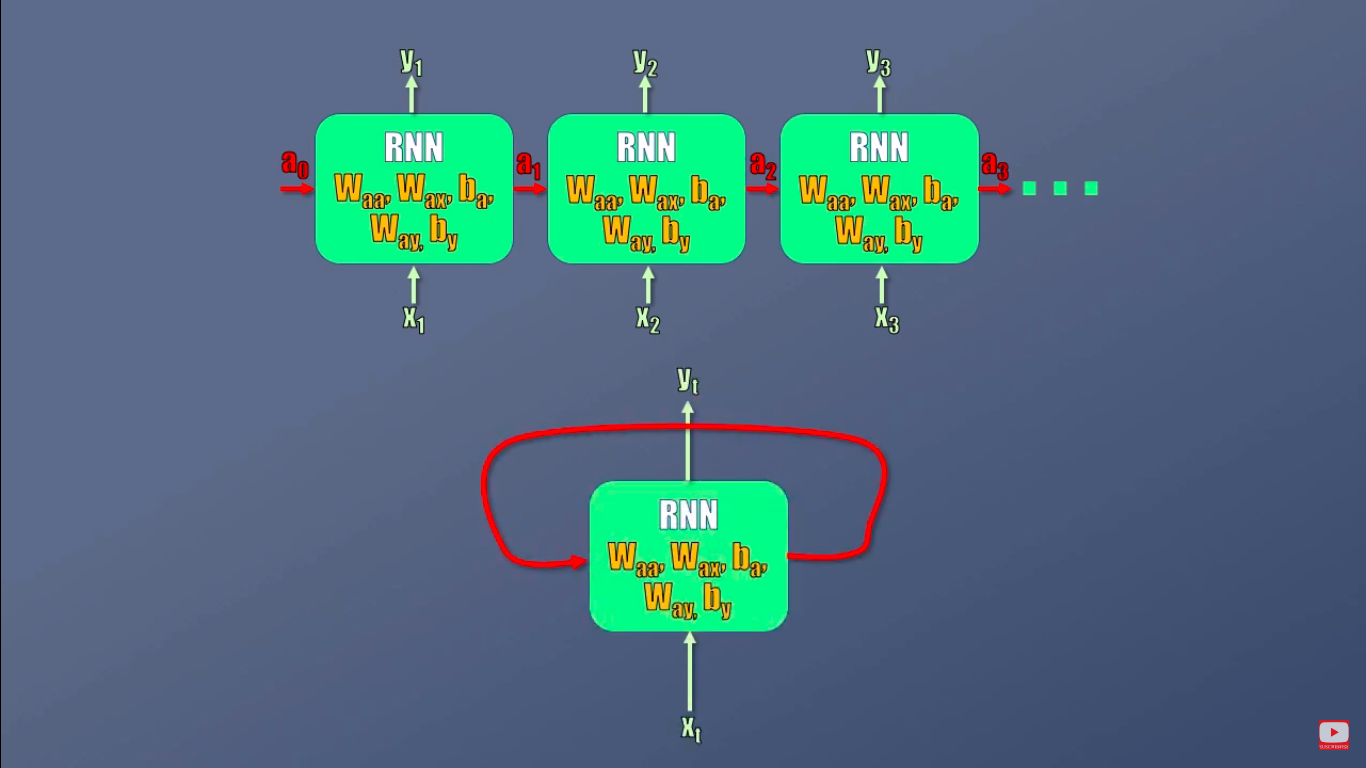
**Figura 4 https://www.youtube.com/watch?v=hB4XYst\_t-I**

En la siguiente imagen se mostrará cómo se calcula la activación en la red neuronal recurrente que primero es la transformación de los datos entrada anterior es decir la activación del estado anterior y la entrada actual y luego de esto llevarlos a una función de activación no lineal además de eso cómo se mejora los valores de los coeficientes de Waa, Wax y ba. Esto se mejora usando el proceso que se utiliza en las redes neuronales que es el proceso de entrenamiento de igual forma se usa la activación del instante previo y se realiza el mismo proceso de transformaciones y luego el proceso de fusión de activación al igual que el proceso anterior se mejoran los valores de los coeficientes por medio del entrenamiento y si observamos bien podemos ver la recurrencia de la red evaluando at en Yt que en esta se encuentran datos del estado anterior como lo es la activación anterior de Waa\*at-1 en este teniendo esto en cuenta se puede ver como la red preserva y comparte la información entre uno y otro instante de tiempo o conocido como **memory cell**

****

**Figura 5 https://www.youtube.com/watch?v=hB4XYst\_t-I**

En la siguiente imagen se da a aclarar que en la red neuronal recurrente no cambian los valores de los coeficientes para la activación ni para la salida sino que estos se conservan a lo largo de la red sino lo que cambia es los valores asociados a ellos por medio de lo que realizan todas las redes neuronales que es el entrenamiento para el ajuste.

****

**Figura 6 https://www.youtube.com/watch?v=hB4XYst\_t-I**

además se debe de tener en cuenta que el modelo de entrenamiento cambia ligeramente debido a que este suma los errores de los anteriores estados ocultos esto con el fin de determinar los gradientes, ya que es propio de los datos de secuencia pero en si es la misma base de un modelo de entrenamiento de una red neuronal conocido como **Backpropagation Through Time (BPTT)**.

También las **RNN** tienen el problema del gradiente de explosion y el desvanecimiento, estos problemas se definen por el tamaño del gradiente, que es la pendiente a través de la función loss o de pérdida de la **RNN**. Cuando el tamaño del gradiente es demasiado pequeño continúa haciéndose más pequeño esto actualizando los parámetros de peso hasta que se vuelve insignificante cuando esto ocurre el algoritmo no está aprendiendo en el caso del gradiente explosivo se tiene cuando el tamaño del gradiente es muy grande por ende el modelo se vuelve inestable por los pesos crecerán mucho, unas de las soluciones puede hacerlo reduciendo capas ocultas.

Además hay diferentes tipos de redes neuronales recurrentes como lo es la de uno a muchos, muchos a uno y muchos a muchos, que cada uno tiene una aplicación en particular para cada tipo por ejemplo de **uno a muchos**, es un único dato de entrada y para la salida es una secuencia de caracteres que describe una imagen, para de **muchos a uno** es texto que a la salida me permite decir una categoría del servicio si es buena o mala. En **muchos a muchos** podemos encontrar de audio a texto

Las redes neuronales recurrentes cuentan con arquitecturas variantes que estas son redes neuronales recurrentes bidireccionales (**BRNN**), otra es memoria corto plazo (**LSTM**) y por último unidades recurrentes cerradas (**GRU**). De estas variantes de **RNN** podemos ver aplicaciones como reconocimiento de escritura que utilizan pantallas táctil y tienen como salida eso que está escribiendo de manera como dibujo en pantalla este ejemplo es reconocido por ser una variante de **LSTM** de **RNN**. Además de esto las redes neuronales recurrentes tienen aplicación en el a campo de internet de las cosas (**IoT**) y en automatización, otros campos de en la que las **RNN** aplican es en el campo multimedia y en las áreas de medicina y biología, en todos estos campos se a llevado múltiples avances con estas redes neuronales recurrentes que han llevado mejora.

**Campos de aplicación:**

**Automatización y internet de las cosas (IoT)**

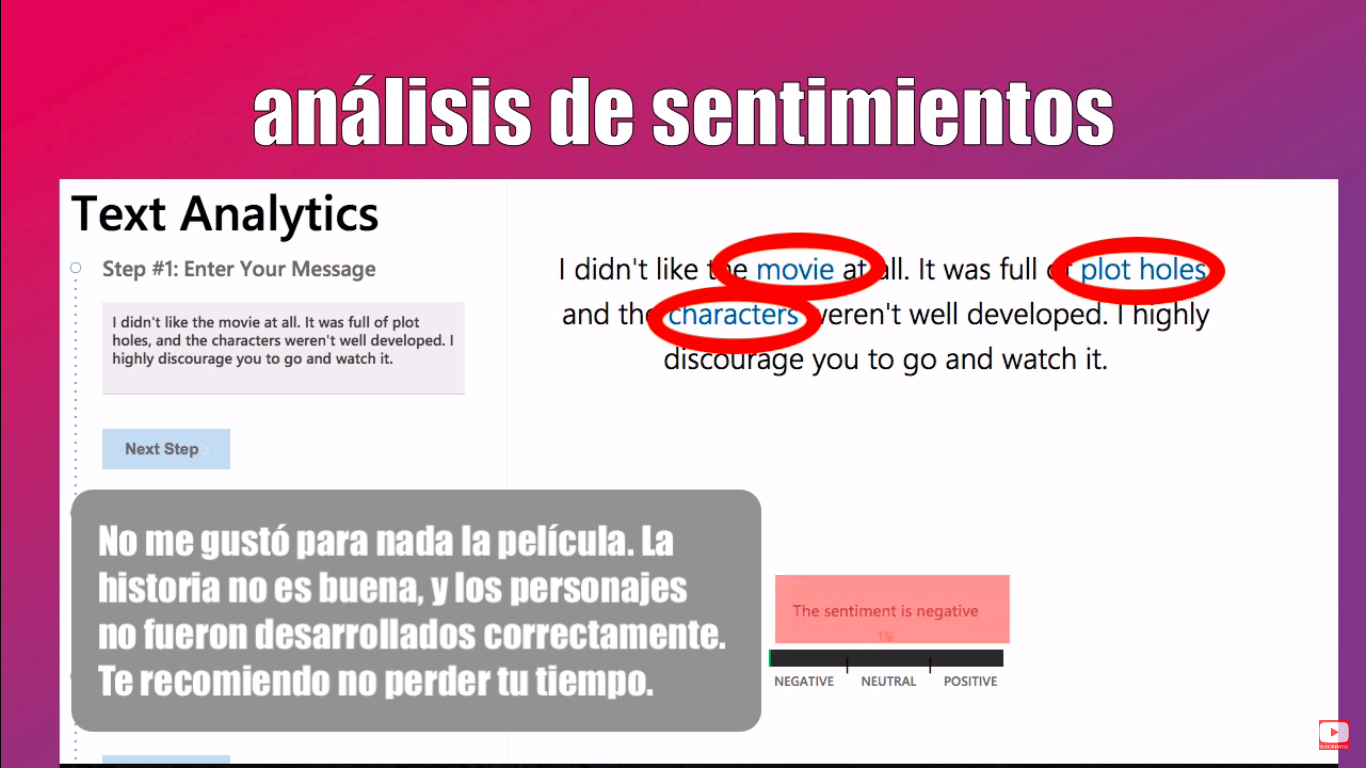
En la siguiente figura 2 se evidencia cómo a partir del procesamiento del lenguaje humano o el **NLP** se puede tomar la decisiones en IoT como la de apagar la luz de una casa entre muchas más.

****

**figura 7 :aplicación de redes neuronales en el campo de IoT**

**Psicología y el Cine**

En la siguiente figura 3 podemos evidenciar cómo aporta una RNN al saber lo que un público del cine que sentimientos siente frente a una película o otros contenidos y así saber qué impacto tiene dicho contenido para las personas que lo consumen.

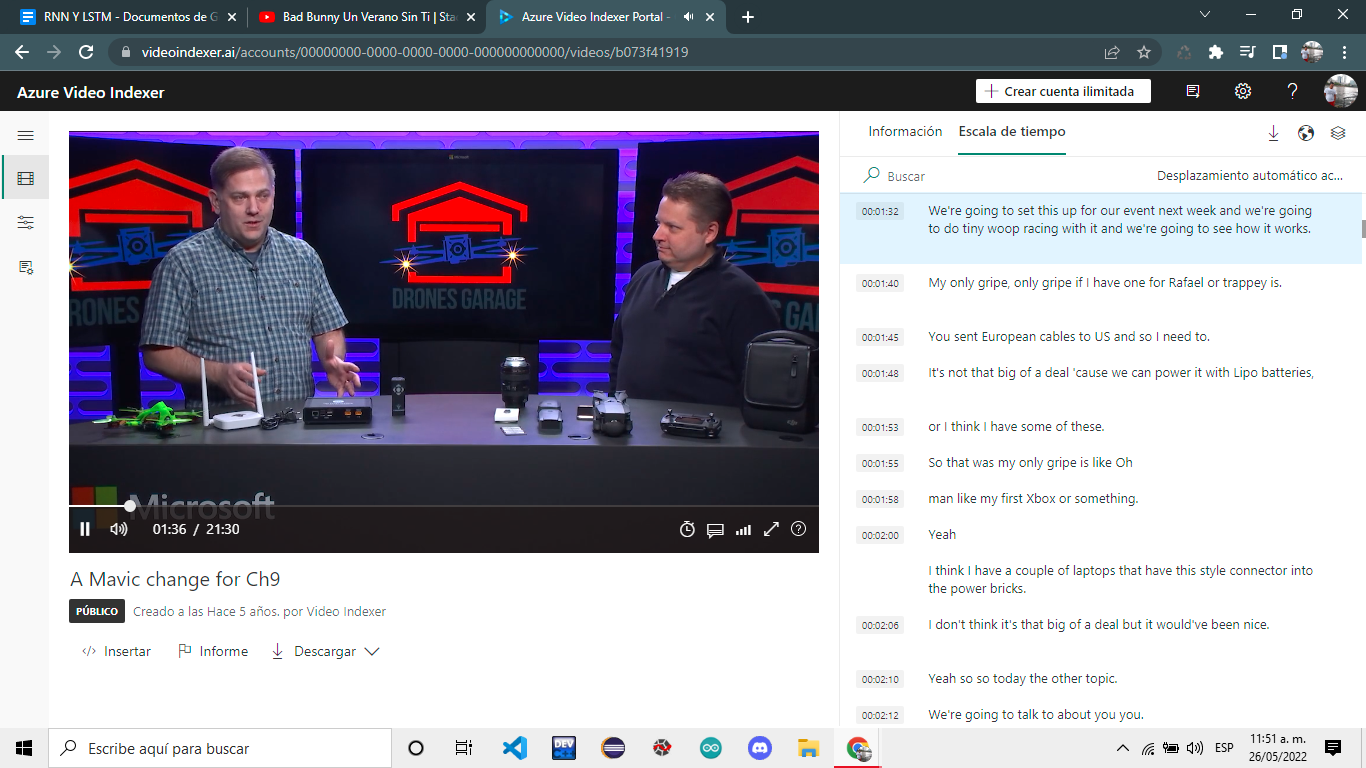


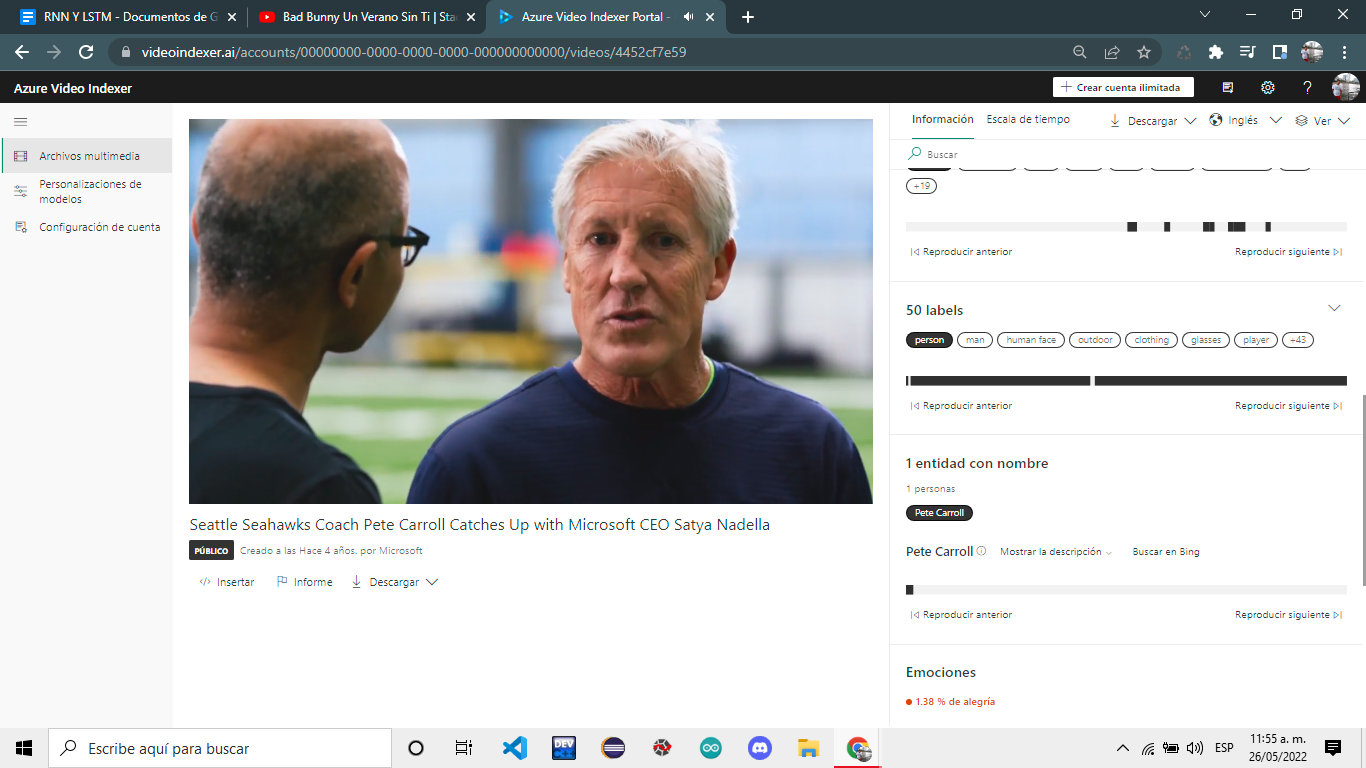
**figura 8 aplicación de redes neuronales en el campo de Psicología y el Cine**

**Indexación de videos**

En esta aplicación es posible analizar secuencias de video ver figura 4 para por ejemplo identificar quien aparece en el video en qué instantes de tiempo los hace, además también permite hacer una traducción de lo que la persona está diciendo para comprender el tema que esté tratando también identifica el cambio del tema y la intencionalidad además permite la cantidad de palabras y dice el porcentaje de sentimiento más usado y la cantidad de temas tratados a lo largo del video.

[**https://www.videoindexer.ai/**](https://www.videoindexer.ai/)



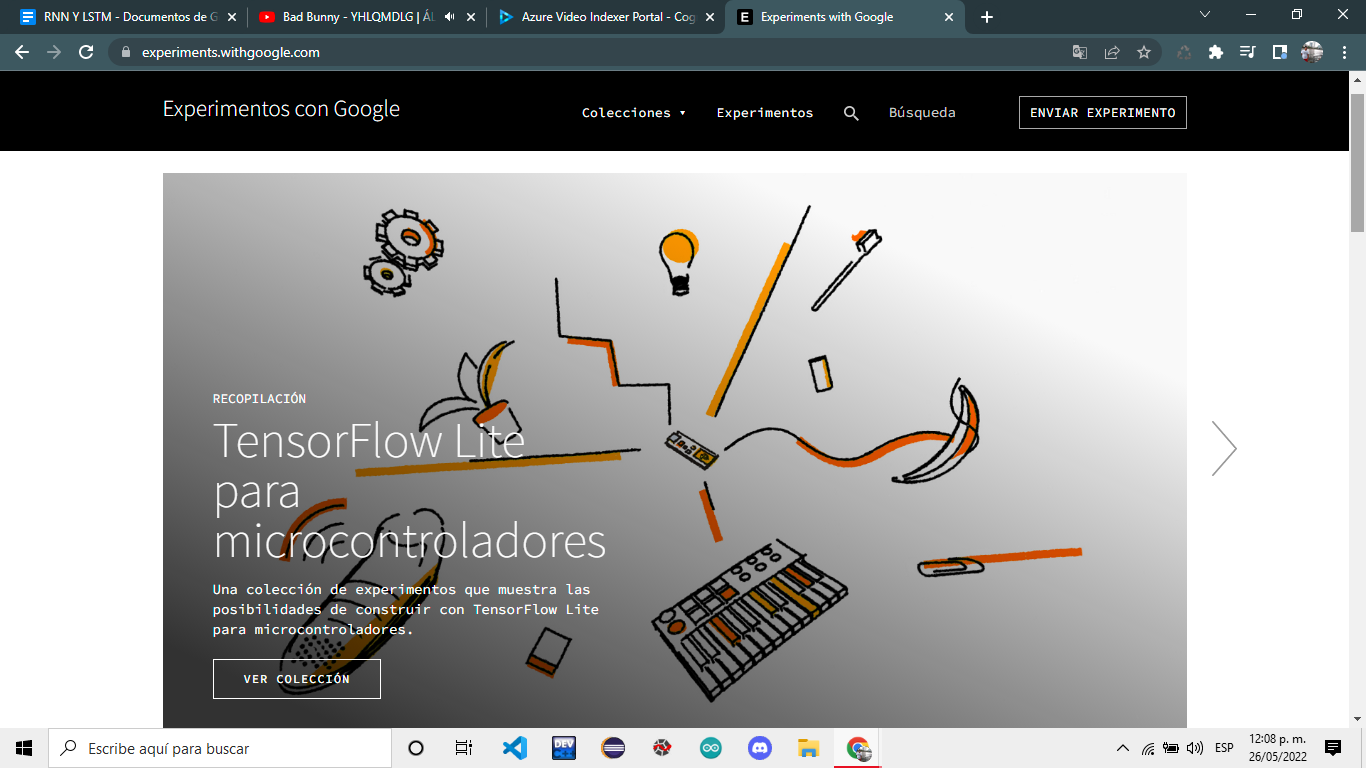


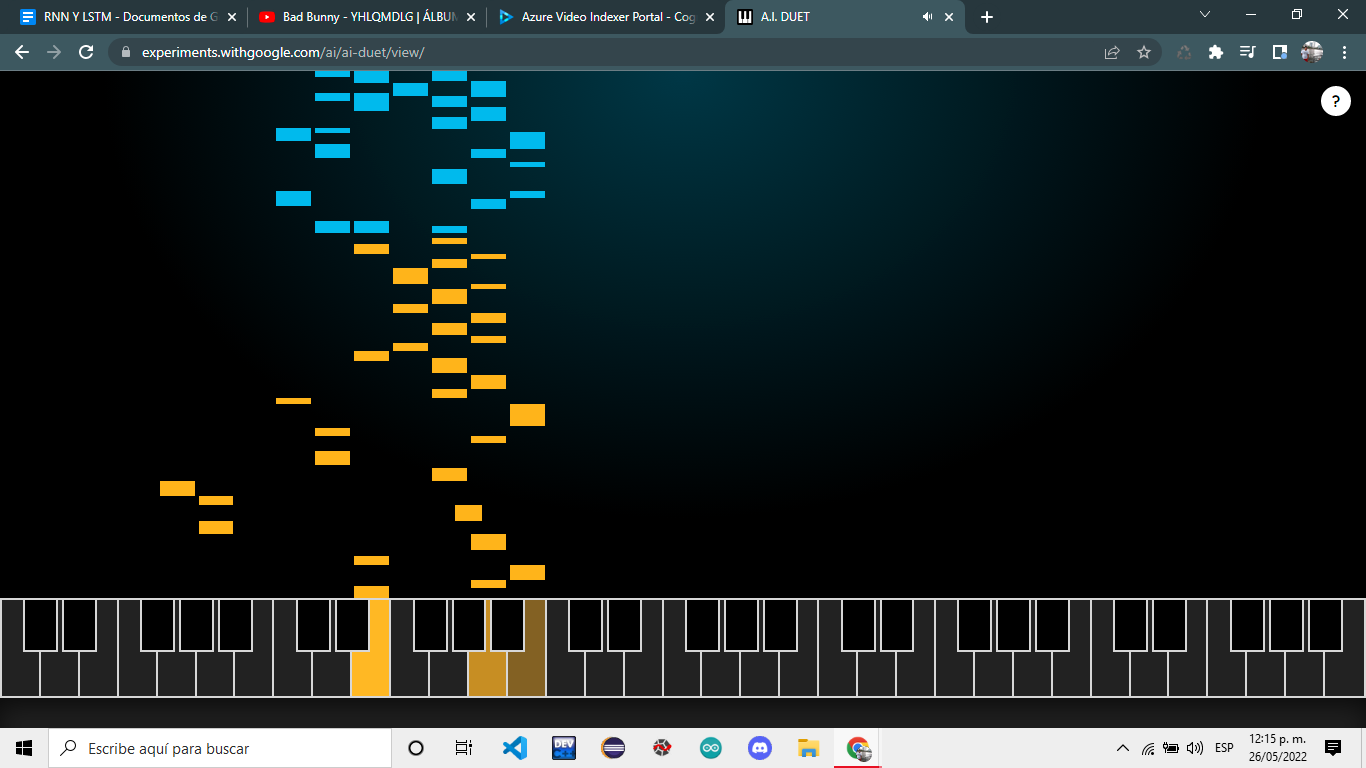
**figura 9: Ejemplo de una RNN de indexación de video**

**Aplicación de una RNN en la música**

En la siguiente figura 5 se puede observar cómo ingresando unas notas la aplicación genera melodías musicales correspondientes de acuerdo a esas notas.

[**https://experiments.withgoogle.com/ai/ai-duet/view/**](https://experiments.withgoogle.com/ai/ai-duet/view/)





**figura 10: Ejemplo de una RNN de melodías por medio de algunas notas**

1. **Describa la arquitectura de una LSTM y la secuencia de operación en la que una LSTM resuelve una oración escrita con sentido para el lenguaje humano.**

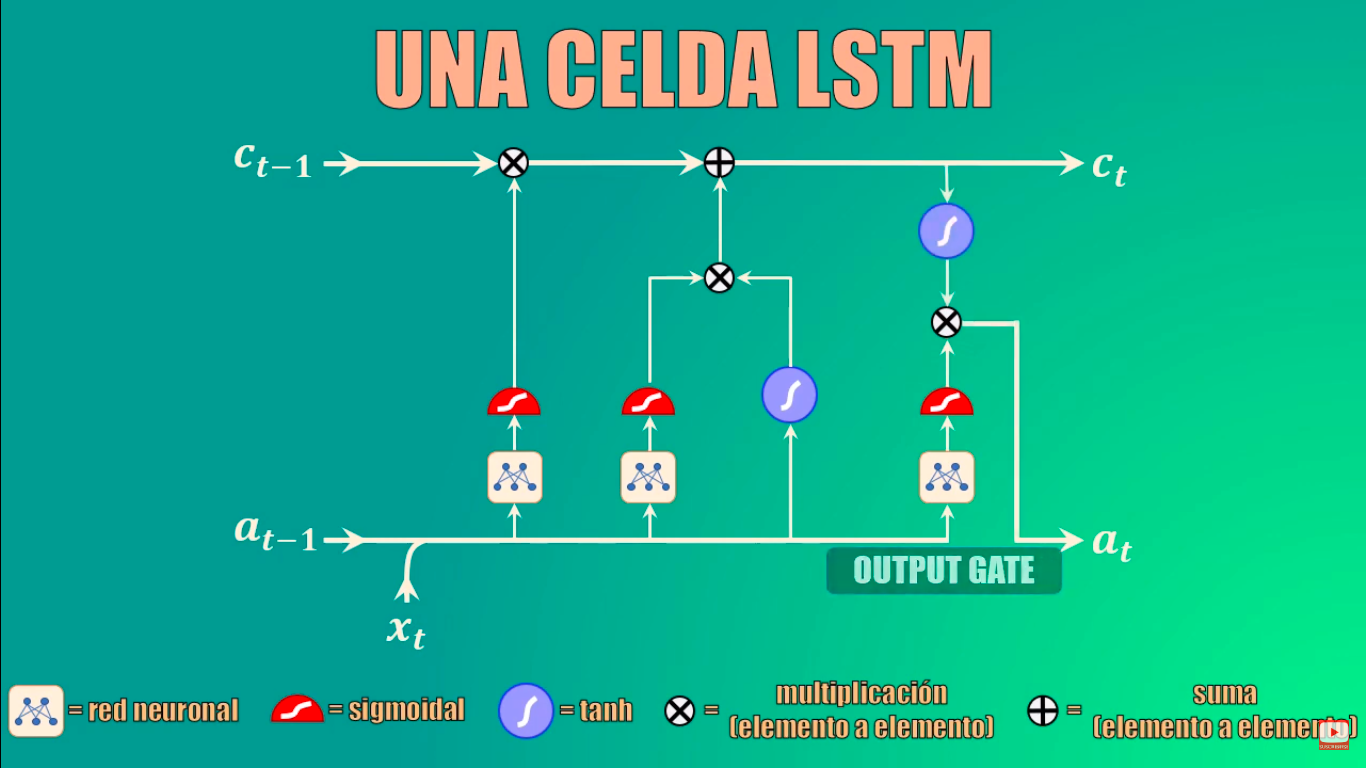
**R/**

**Arquitectura de una LSTM**

Una red LSTM es capaz de recordar un dato relevante en la secuencia y de preservarlo por varios instantes de tiempo por lo tanto puede ser de corto plazo como la de las redes neuronales recurrentes como las de largo plazo es decir que este tipo de red analiza como nosotros los seres humanos analizamos las secuencias como por ejemplo teniendo en cuenta eventos especiales en cambio en las redes sería las palabras más relevantes que a pasado en la memoria de largo plazo y estas deciden si borrar esa información que no tiene ese peso y esto lo puede hacer debido a que este tipo de red neuronal recurrente extendida cuenta con puertas que permiten hacer esto como lo son puerta de entrada de lectura y actualizar o borrar esto lo puede ir haciendo siempre y cuando aporte a la secuencia adicional a esto podemos ver que en las LSTM tiene una celda LSTM que esta lo que aporta en una entrada y una salida adicional al modelo de RNN tradicional ver la figura 11 en la imagen se puede ver que tanto la entrada como la salida se denomina **celda de estado (cell state)**  que esta celda lo que su funcion es añadir o remover datos a la red para que no quede informacion en la red se necesita de varias compuertas como lo son forget gate, que esta permite eliminar elemento de la memoria y el otro que permite adicionar es update gate, que esta permite adicionar nuevos elementos a la memoria y por ultimo la output gate, que lo que permite crear el estado actualizado estas son como redes neuronales que funcionan como si abrieramos y cerraramos lo que permite es fluir informacion o no esto lo podemos ver en la figura 12 de como seria la representacion de lo que se a dicho hasta el momento.



**Figura 11 Donde podemos ver elemento adicional a la RNN tradicional**

****

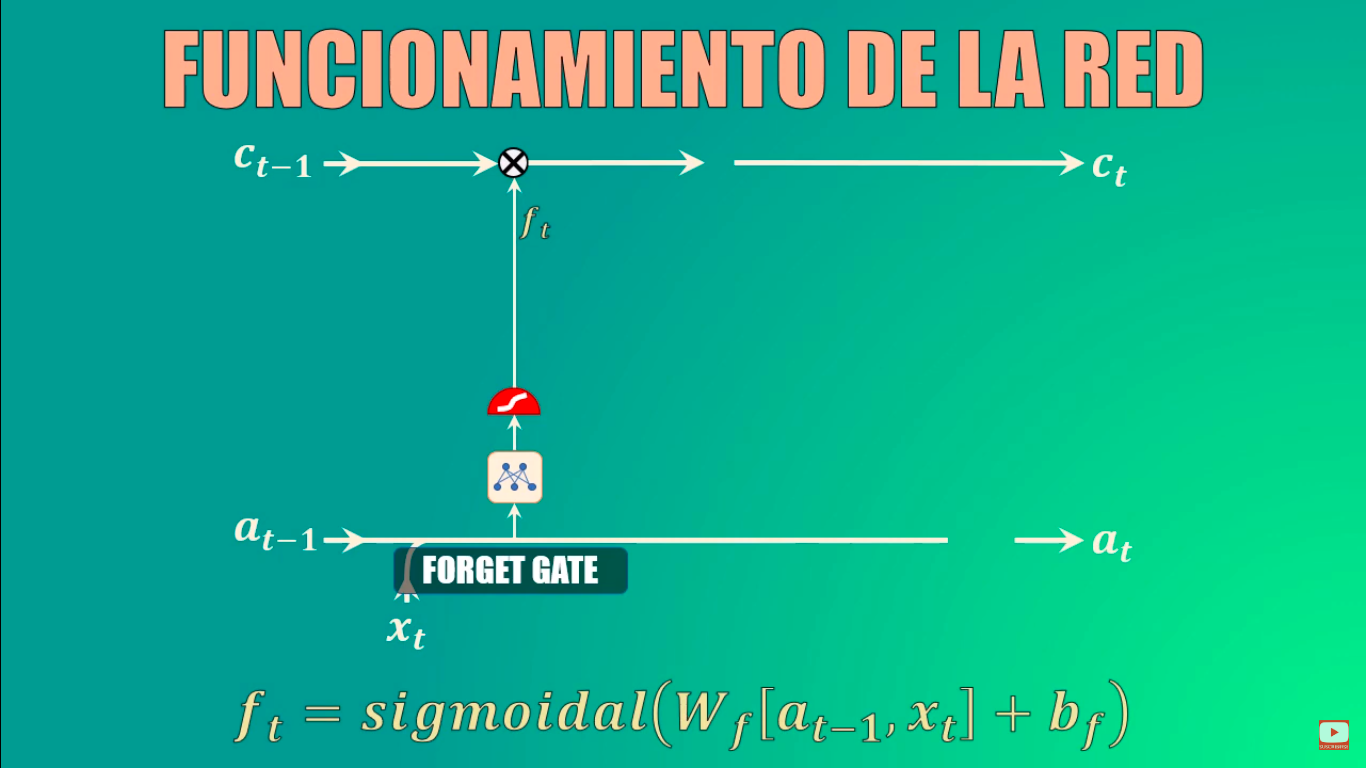
**Figura 12 Estructura de lo que se puede hacer en una LSTM**

A continuación entraremos más a detalle cómo funciona cada una estas compuertas que tiene la celda **LSTM** ver figura 13 cada una tiene tres elementos los cuales son una red neuronal, una función sigmoidal y un elemento multiplicador la cual la función sigmoidal le da el flujo si es cerrado o abierto pues al alcanzar valores entre cero y uno permite cancelar por completo valores de entrada. Estas compuertas permiten manipular la celda de memoria.

****

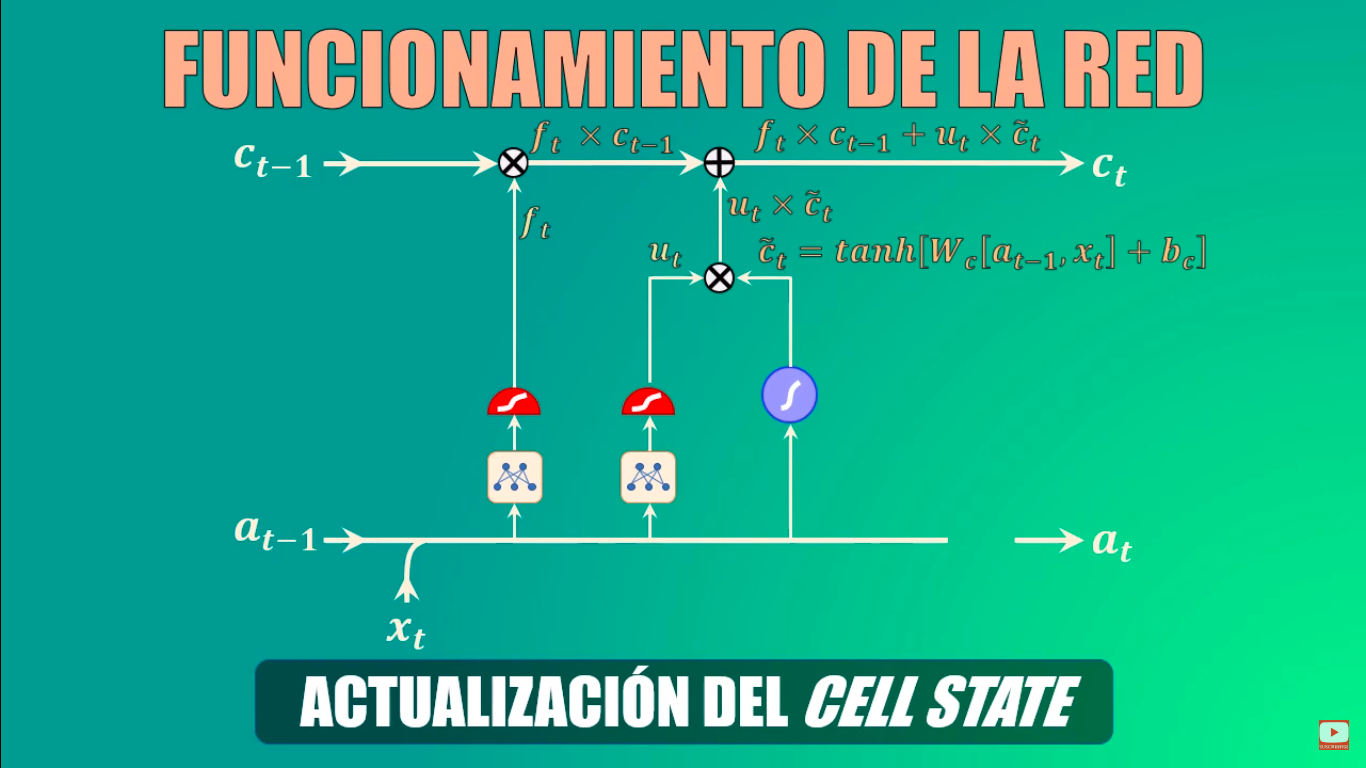
**Figura 13 Elementos de una compuerta además cerrada o abierta para el flujo**

Continuando con la arquitectura la compuerta forget gate ver figura 14 esta permite descartar que información no va entrar y por ende no va a entrar a la celda de estado para ello toma el estado oculto anterior y la entrada actual los transforma y los lleva a la función de activación sigmoidal y los coeficientes de Wf y Bf se aprenden por el entrenamiento y como salida genera el vector Ft si algunos de los datos es cero o cercano a cero la LSTM eliminara esa porción de información mientras que si los valores son uno o cercanos la información se mantendrá la información en la celda de memoria además podemos observar la destincion de las palabras si estan en plural o singular y mirar si corresponde al sentido de la oracion si es asi continuamos con la compuerta uodate gate para indicar si el sujeto es singular en el caso que ese sea correcto para realizar dicha operacion utilizamos l estado oculto anterior y la entrada actual se transforma se lleva de nuevo a una funcion de activacion sigmoidal luego los mismos coeficientes con el entreno que biene aprendiendo y como salida genera el vector de salida Ut en este caso los valores a preservar son los datos a preservar seran los cercanos a uno teniendo ya generado los datos forget gate y update gate podemos actualizar la celda de estado o la memoria de la red LSTM, en primer lugar eliminamos los datos irrelevantes del vector de la celda de estado multiplicando el valor anterior de esta celda o por el vector generado por el forget gate luego un vector con los candidatos manegando la estructura de la figura 15 luego filtramos los valores de eso realizamos un producto punto entre Ut x Ct y los resultados los sumamos en la celda de estado generando asi la memoria actualizada de la cell state.



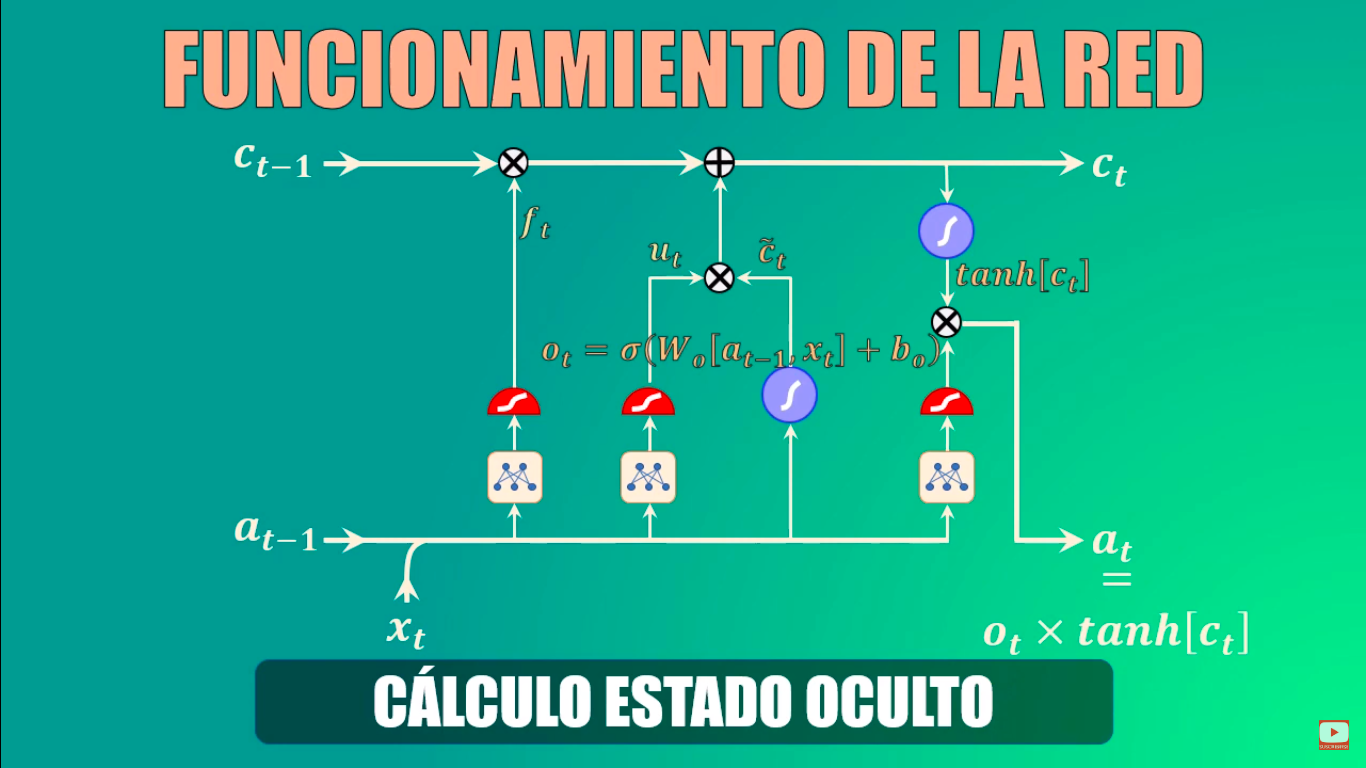
**Figura 14 la compuerta Forget gate**

A continuación la compuerta de update gate de la LSTM y con el proceso que fue explicado anteriormente.



**Figura 15 compuerta Update gate y como se realizó el proceso de actualización de la celda de estado**

luego debemos calcular el nuevo estado oculto para lo cual continuamos con la última compuerta que es la output gate o compuerta de salida para ello se usa la funcion de activacion de tangente hiperbólica y luego la compuerta de salida para saber qué porciones del cell state entrarán a formar parte del estado oculto al igual que en compuertas anteriores continuamos con el mismo proceso y las constantes aprendidas durante el entrenamiento y por último se filtran los valores con la compuerta de salida con todo esto ya se tendieron en cuenta todos los elementos de la variante de RNN la LSTM

****

**Figura 17 cómo se calcula el estado oculto de acuerdo a los elementos de la LSTM y las operaciones que realiza**

continuando con esto adicional a una RNN tradicional esta cuenta con una banda transportadora que cuenta con elementos que me permiten modificar borrar y la de salida esto con el entrenamiento que realiza cada compuerta que va haciendo en el cell state ver figura 18 además se puede con ir entrenando las compuertas forget gate y update gate es fácil que el estado de la banda C0 se propague hasta la C5 o hasta más estados posteriores.

Gracias a la cell state se puede garantizar una memoria de largo estado a diferencia de la red neuronal recurrente normal y esta banda se puede manipular fácilmente por medio de las compuertas con el fin de obtener información relevante lo que hace de esta variante de la RNN mucho más robusta porque tiene tanto como la de corto y largo plazo en el ámbito más común estas son las más usadas en este campo por lo que son más completas.



**Figura 18 se ve la banda donde entran y intervienen las compuertas para la celda de estado de la LSTM**

**Referencias:**

[**https://torres.ai/redes-neuronales-recurrentes/**](https://torres.ai/redes-neuronales-recurrentes/)

[**https://medium.com/analytics-vidhya/natural-language-processing-from-basics-to-using-rnn-and-lstm-ef6779e4ae66**](https://medium.com/analytics-vidhya/natural-language-processing-from-basics-to-using-rnn-and-lstm-ef6779e4ae66)

[**http://colah.github.io/posts/2015-08-Understanding-LSTMs/**](http://colah.github.io/posts/2015-08-Understanding-LSTMs/)