

# Computación inteligente y ética de la IA

## Proyecto de la Asignatura

### Chatbot para tienda online

### Equipo 5

Integrantes:

1. Manuel Mateo Delgado-Gambino López
2. Manuel Mora Rivas
3. Bernardo Quindimil Micó

## 1 Participación del equipo

El responsable del equipo, que debe estar en la primera fila de la tabla, es el **garante del cumplimiento de lo referido a las prácticas injustas** en el apartado 4 del contrato de enseñanza-aprendizaje de la asignatura.

La información de la participación se refleja en la tabla modelo según lo siguiente:

En la columna participación el responsable del equipo indicara para cada integrante (incluyéndose) alguna de las siguientes opciones:

Opción **SI** (el integrante participa activamente en la elaboración de la asignación)

Opción **NO** (el integrante no ha participado en la elaboración de la asignación)

Opción **%** (el integrante participa en un porcentaje respecto a la Opción SI.

El % implica que ese integrante obtendrá ese porcentaje de la calificación obtenida por el equipo.

Nombre	Participación
Manuel Mateo Delgado-Gambino López	Si
Manuel Mora Rivas	Si
Bernardo Quindimil Micó	Si

## 2 Título: Chatbot de tienda OnLine

### 3 Descripción

Un Chatbot es un programa informático que utiliza inteligencia artificial (IA) y procesamiento de lenguaje natural (NLP) para comprender las preguntas de los clientes y automatizar las respuestas a dichas preguntas, simulando la conversación humana.

Antes los Chatbots estaban basados en texto y respondían a un conjunto limitado de consultas simples. Ahora, han evolucionado y utilizan comprensión del lenguaje natural (NLU) para contestar al usuario, pueden reconocer el texto y aprender a medida que interactúan con el humano; estos chatbots son capaces de comprender mejor al humano y responder con mayor precisión a lo largo del tiempo.

En este proyecto, nosotros creamos un chatbot de una tienda de ropa on line, capaz de resolver dudas al usuario que interactúe con él.

Este está hecho en Python usando las siguientes librerías: nltk, tensorflow, numpy y tflearn. Hemos resuelto también problemas que nos han ido surgiendo como la eliminación del video de referencia, líneas de código obsoletas etc.

## 4 Objetivos

Nuestro objetivo fue realizar un chatbot enfocado en una tienda online y que pudiese responder preguntas como “Que tallas tienes?”, “Puedo devolver un artículo?” El chatbot debe también ofrecer información acerca de una prenda, sobre la política de la empresa... También debe ser capaz de comprender preguntas formuladas de manera natural, reconociendo la consulta del cliente, siempre y cuando esté relacionada con la tienda de ropa y debe proporcionar una respuesta relevante. Debe dar feedback y recopilar comentarios de los usuarios y utilizarlos para mejorar el chatbot con el tiempo.

## 5 Metodología

Hemos realizado el chatbot con la ayuda de un vídeo en youtube y documentándonos en la página oficial de librerías como tensorflow, nltk, numpy y keras, que es un submódulo de tensorflow y es el que más utilizamos. En ese vídeo había muchos comandos de algunas librerías que estaban obsoletos por lo que tuvimos que recurrir a informarnos en la página oficial de Keras para completar el código. También hemos estado buscando información acerca de la importancia de la implementación de los chatbot en las empresas.

## 6 Desarrollo y avances del proyecto

TensorFlow es una biblioteca de código abierto para aprendizaje automático a través de un rango de tareas, y desarrollado por Google para satisfacer sus necesidades de sistemas capaces de construir y entrenar redes neuronales para detectar y descifrar patrones y correlaciones, análogos al aprendizaje y razonamiento usados por los humanos. Actualmente es utilizado tanto en la investigación como en los productos de Google frecuentemente reemplazando el rol de su predecesor de código cerrado, DistBelief. Tensorflow la hemos utilizado como plataforma para poder ejecutar Keras.

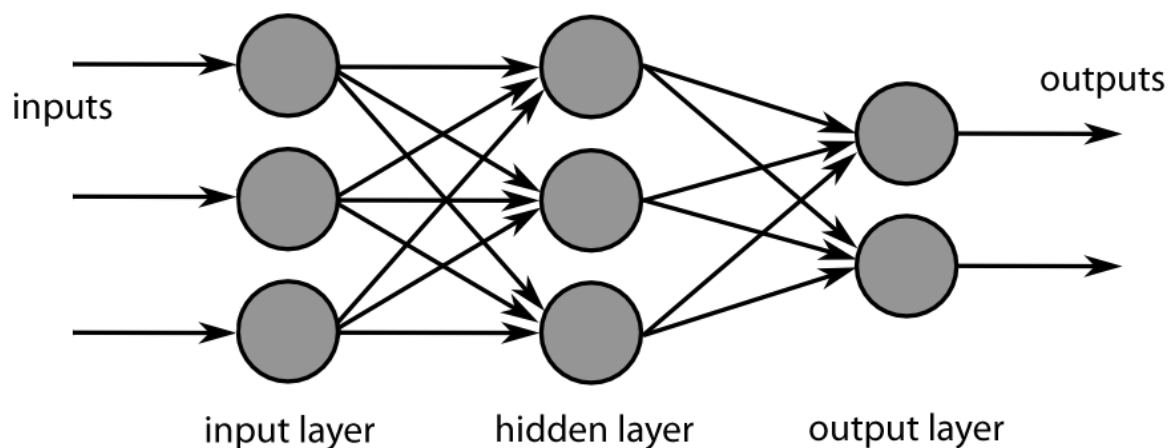
NLTK o kit de herramientas de lenguaje natural es un conjunto de bibliotecas y programas para el procesamiento de lenguaje natural (PLN) simbólico y estadísticos para Python. Nosotros la utilizamos para tokenizar las oraciones, es decir, tenemos una oración “hola, ¿que tal?” y te la divide en tokens “hola”, “que”, “,” “que”, “tal” y “?”.

Keras es una librería de redes neuronales de código abierto escrita en python. Está especialmente diseñada para posibilitar la experimentación en más o menos poco tiempo con redes de Aprendizaje profundo. Es la que más hemos utilizado.

NumPy es una biblioteca de computación numérica en Python que proporciona soporte para matrices y arreglos multidimensionales grandes, junto con una colección de funciones matemáticas para operar en estos arreglos de manera eficiente.

Primero le damos al algoritmo un conjunto de entrenamiento que son preguntas y frases que puede formular un posible cliente como "¿hola, que tal estas?", esta oración se va a stemmizar (convertir las palabras derivadas en palabras raíz por ejemplo estuvimos estamos estaré se transforman en estar) para simplificar el problema, tras esto se van a tokenizar las oraciones como explicamos previamente con NLTK.

Con la ayuda de estas librerías hemos realizado una red neuronal con una capa de entrada, una oculta y una de salida. La cantidad de neuronas de la capa de entrada viene determinada por la longitud de los tokens de entrada del conjunto de entrenamiento (que son las palabras y símbolos de las frases de un posible cliente), en la capa oculta hay 10 neuronas y en la capa de salida viene determinada por la longitud de los tokens de la salida (las palabras y símbolos stemmizados del chatbot). Cada neurona de una capa está enlazada con todas las neuronas de la capa siguiente y cada enlace tiene un peso atribuido. La salida de cada neurona será el resultado de realizar una función de activación a la suma ponderada de las entradas multiplicadas por los pesos. Nosotros hemos utilizado dos funciones de activación: ReLU, que devuelve la entrada de la neurona si es positiva, de lo contrario, devuelve un 0; y soft max, que devuelve la salida de la capa con probabilidades que suman 1. De las neuronas de la capa de entrada a las neuronas de la capa oculta se aplica a los inputs la función de activación ReLU. De las neuronas de la capa oculta a las capas de salida se propaga con la función soft max. Nuestra red neuronal es feed forward que quiere decir que no tiene lazos de retroalimentación y las conexiones van en una única dirección desde las entradas a las salidas.



Supongamos que el usuario ingresa la frase "¿Hola, como estas?". Aquí está cómo se procesaría esa frase a través de la red neuronal:

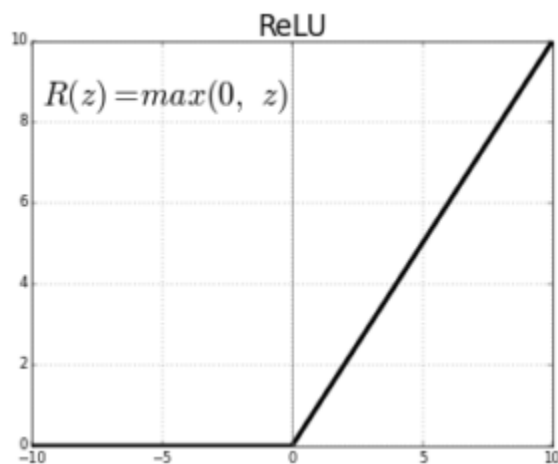
Tokenización y Stemming: La frase se divide en palabras individuales y se aplica stemming a cada palabra. Supongamos que nuestro stemmer no cambia ninguna de las palabras en este caso. Entonces, la frase tokenizada y con stemming sería ["hola", ",", "¿", "como", "estar", "?"].

Creación del vector de características: Para cada palabra en el vocabulario del bot, verificamos si la palabra está en la frase tokenizada. Si es así, agregamos un 1 al vector de características, de lo contrario agregamos un 0. Supongamos que nuestro vocabulario es: ["hola", "como", "estas", "adios", "hasta", "luego", "quiero", "comprar", "ropa"]. Entonces, el vector de características sería [1, 1, 1, 0, 0, 0, 0, 0, 0].

Pasar el vector de características a través de la red: Este vector se pasa a la capa oculta de la red, que realiza una suma ponderada de los elementos del vector y luego aplica la función de activación 'ReLU' el cual devuelve 0 si el valor es menor que 0 o devuelve el valor de x si x es mayor que cero y como tenemos un array de 0 y 1 solo puede multiplicar cada elemento del array por 0 o por 1.

Por ejemplo, si tenemos una neurona con pesos [0.2, 0.3, 0.1, 0.4, 0.5] y el array de entrada es [0,1,0,1,1], la suma ponderada sería:

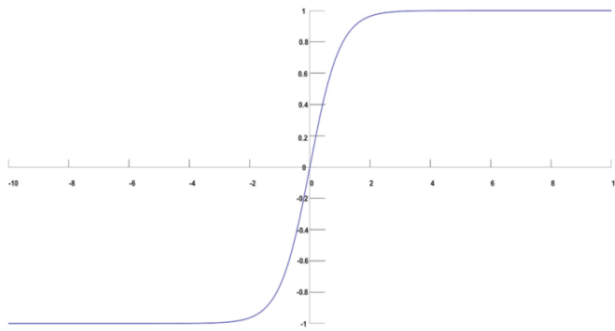
$$0.2*0 + 0.3*1 + 0.1*0 + 0.4*1 + 0.5*1 = 0 + 0.3 + 0 + 0.4 + 0.5 = 1.2$$



$$f(x) = \max(0, x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

Pasar el nuevo vector de características a la capa de salida: Este vector se pasa a la capa de salida, que realiza una suma ponderada de los elementos del vector y luego aplica la función de activación 'softmax'. Esto produce un vector de probabilidades.

Supongamos que este vector es [0.7, 0.1, 0.2, 0, 0, 0, 0, 0, 0].



$$f(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

Selección del tag: La red selecciona el tag con la mayor probabilidad como la predicción de la red para la frase. En este caso, el primer tag ("saludo") tiene la mayor probabilidad (0.7), por lo que ese sería el tag seleccionado.

Generación de la respuesta: La red selecciona una respuesta al azar de las respuestas asociadas con el tag seleccionado.

Supongamos que selecciona "¡Hola!".

Por lo tanto, la respuesta de la red a la frase "¿Hola, como estas?" sería "Hola!".

## 7 Problemas y soluciones

Al inicio nos hemos guiado de un video, el problema es que, al ponernos a trabajar, ese video no estaba disponible, así que hemos tenido que buscar otro como referencia, que incluía una manera diferente de hacerlo con respecto al anterior video, librerías, organización del algoritmo distinta...

En el vídeo había una gran cantidad de líneas de código que estaban obsoletas y tuvimos que arreglárnoslas buscando en la página oficial de algunas librerías como tensorflow.

## Conclusión:

Hemos aprendido cómo funciona un chatbot, así como su uso en las empresas y en el mundo de las tiendas on-line. Hemos indagado en su estructura interna para comprender como funciona.

En las empresas, mejoran considerablemente la experiencia del cliente, con la mejora de experiencia, el aumento de las ventas.

Al ser un bot, necesita un poco de mantenimiento periódico, pero sale rentable a la larga por lo que resulta en una reducción de costes para la empresa importante.

## Bibliografía

Cursos y asesorías de programación. (2019, 18 septiembre). Creando ChatBot con IA en Python #01 [Vídeo]. YouTube. <https://www.youtube.com/watch?v=4bCgUs1fsXc>

¿Qué es un chatbot? | IBM. (s. f.). <https://www.ibm.com/es-es/topics/chatbots>

colaboradores de Wikipedia. (2024, 23 abril). Keras. Wikipedia, la Enciclopedia Libre. <https://es.wikipedia.org/wiki/Keras>

IA coach. (2020, 14 abril). Redes neuronales feedforward [Vídeo]. YouTube. <https://www.youtube.com/watch?v=GLmldPT49E>