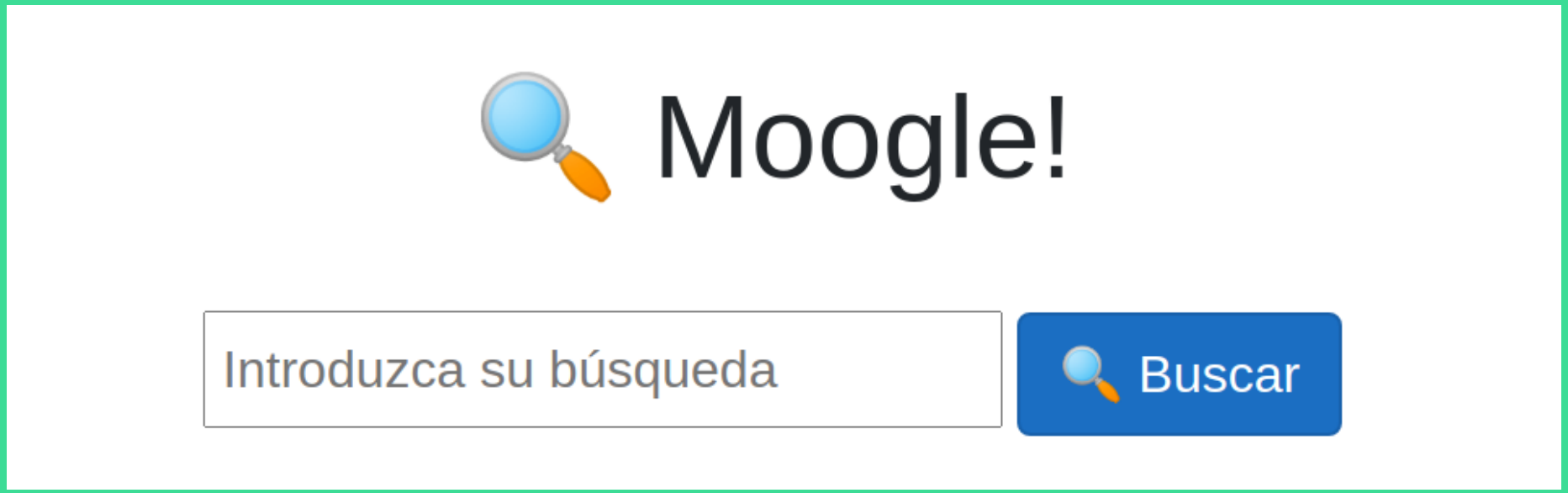
Presentación Moogle!

David Michel García Batista

Julio de 2023

Moogle! ¿En qué consiste esta herramienta?

La aplicación web Moogle! es un modelo de búsqueda que nos devolverá los documentos más relevantes y similares a la información que ingrese su usuario en lo que denominaremos **Query**.



El nombre de este programa es netamente original. Cualquier parecido con otro programa o aplicación de la vida real es pura coincidencia

¿Cómo funciona?

Nuestra problemática es simple, solucionarla no tanto

Tenemos una serie de documentos .txt y en dependencia de una Query ingresada por el usuario debemos devolver los más relevantes

¿Qué hacer entonces?

¡Pues claro!

Cómo buenos matemáticos que somos, calcularemos la **relevancia** de cada documento con respecto a la Query para compararlos y así determinar aquellos documentos más importantes. Él metodo será utilizando el TF-IDF (Ver fig 1 Anexos) y la similitud de cosenos

De qué va la cosa:

Nuestro programa funcionará con varias clases encaminadas a cada función que se necesitará para resolver nuestra problemática:

1. Recibir el contenido de los documentos y procesarlos para poder trabajar
2. Calcular la importancia de estos documentos y de la Query ingresada
3. Crear una base de datos que almacene toda esta información y permita compararla y maniobrarla

Y pues como tenemos 3 funciones, se designará una clase a cada una de ellas.

Clase Document

Será quien realizará el trabajo inicial, se llenará las manos de información y procesará cada texto

- Buscar (Directory.GetCurrentDirectory)
- Recopilar información (Directory.GetFiles)
- Fragmentar textos (Método Split Text)
- Extraer Títulos (Método Get Title)

A eso se dedica, de eso vive, ese es su propósito...
Un poco triste no creen?

Clase TF-IDF

Será quien haga el trabajo sucio del proyecto. Números van y números vienen
Calcular el TF-IDF (Term Frequency-Inverse Document Frequency) de cada palabra. Una vez obtenidos se puede hallar la relevancia con la Query mediante la similitud de cosenos
(Ver fig 2 y 3 Anexos)

Clase Database

Debido a que la originalidad es nuestra principal característica, esta clase como su nombre indica será quien creará una base de datos para organizar la información
Se utilizarán dos componentes:

1. Diccionarios
2. Listas

Con ellas se agruparán todos los documentos con cada palabra y su TF-IDF asociado así como la Query para determinar las relevancias y comparando para finalmente devolver los resultados.

Anexos

$$TF = \frac{Number\ of\ times\ a\ word\ "X"\ appears\ in\ a\ Document}{Number\ of\ words\ present\ in\ a\ Document}$$

$$IDF = log\left(\frac{Number\ of\ Documents\ present\ in\ a\ Corpus}{Number\ of\ Documents\ where\ word\ "X"\ has\ appeared}\right)$$

$$TF\ IDF\ = TF * IDF$$

Fig.1: Fórmulas para calcular TF-IDF

Anexos

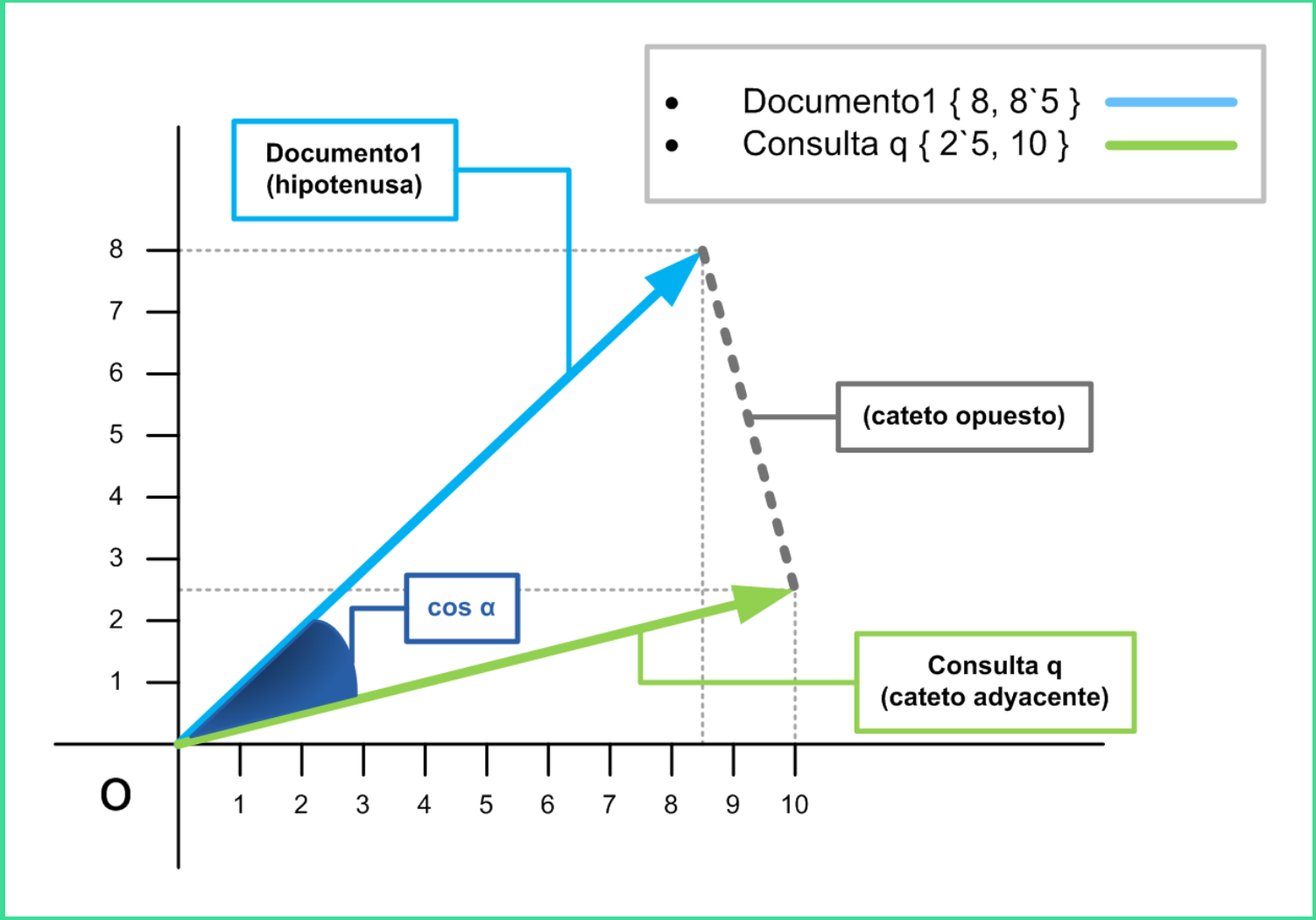


Fig.2: Representación geométrica de la similitud de cosenos

Anexos

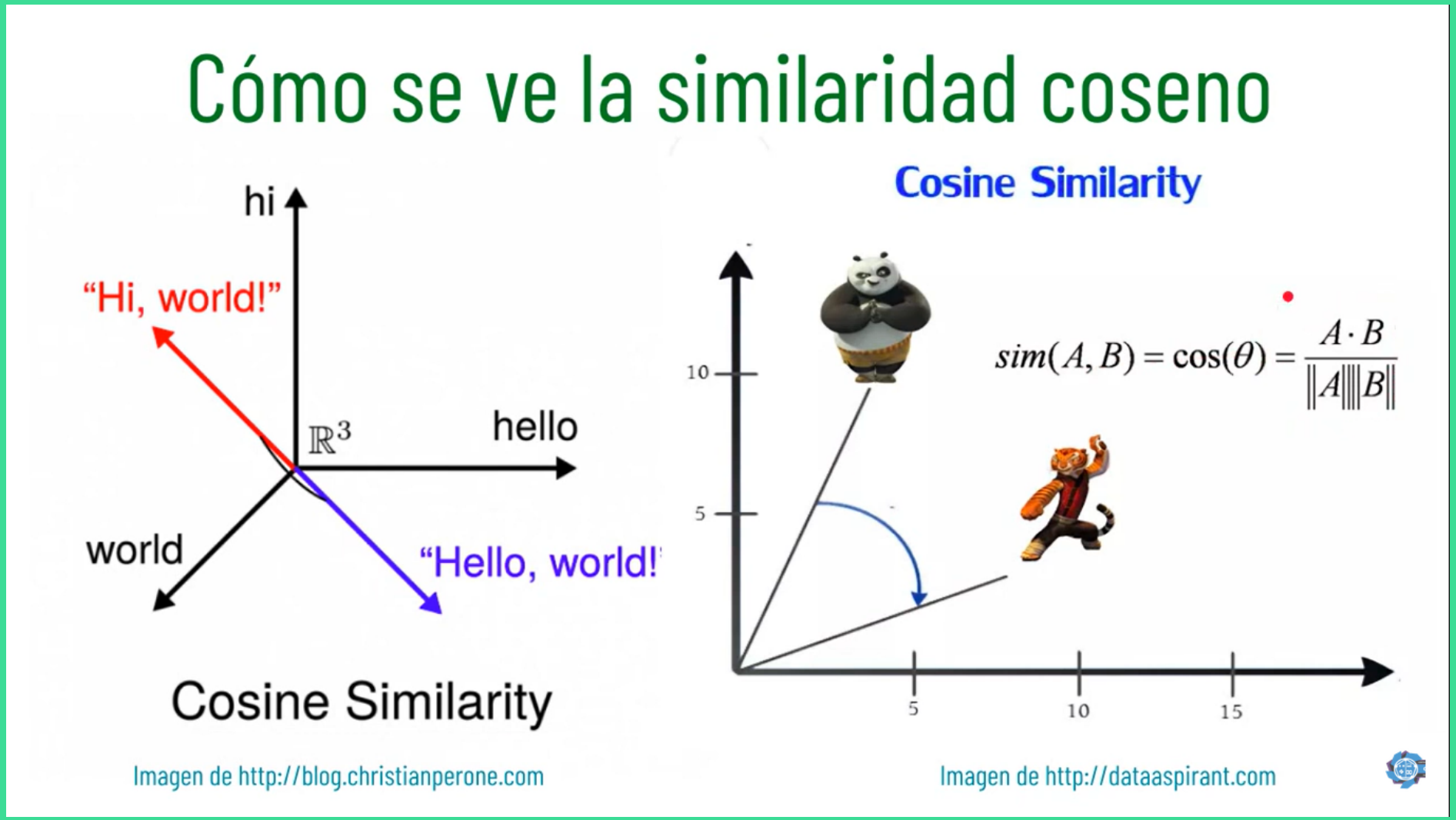


Fig.3: Fórmula Similitud de cosenos