# Recuperación de información

## Motores de búsqueda, clasificación de documentos



# Índice

---

- Motivación
- Objetivo
- Introducción
- Representación de un documento
- TF-IDF
- Más cosas...

# Motivación

Y esto, ¿para qué sirve?
¿Me interesa?

Búsquedas en internet

Clasificación de docs.

¿Cómo lo hacen?

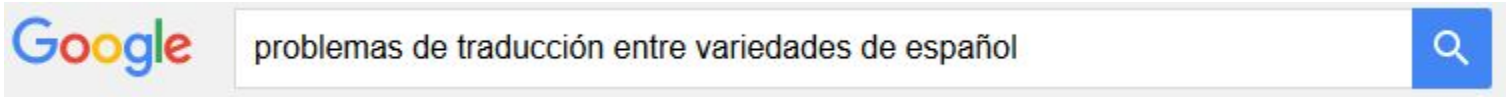
---

# Las búsquedas en internet

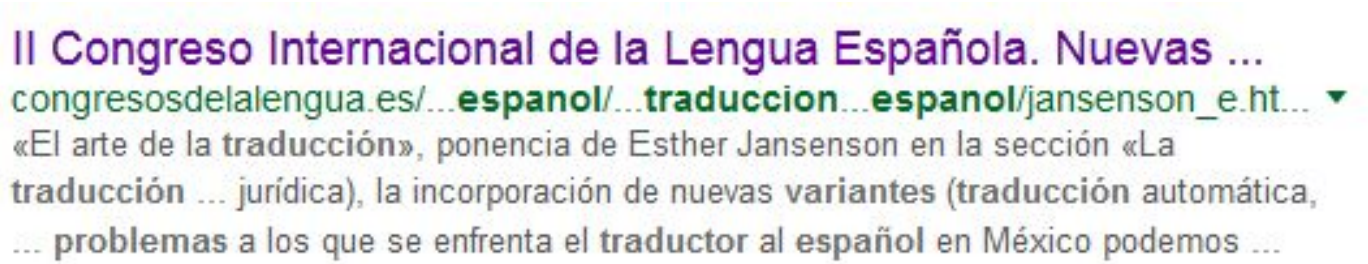
# Las búsquedas en internet

---

- Búsqueda basada en **palabras clave** (keywords)




- ¡No sólo busca en los títulos! ¿Se lo lee todo!

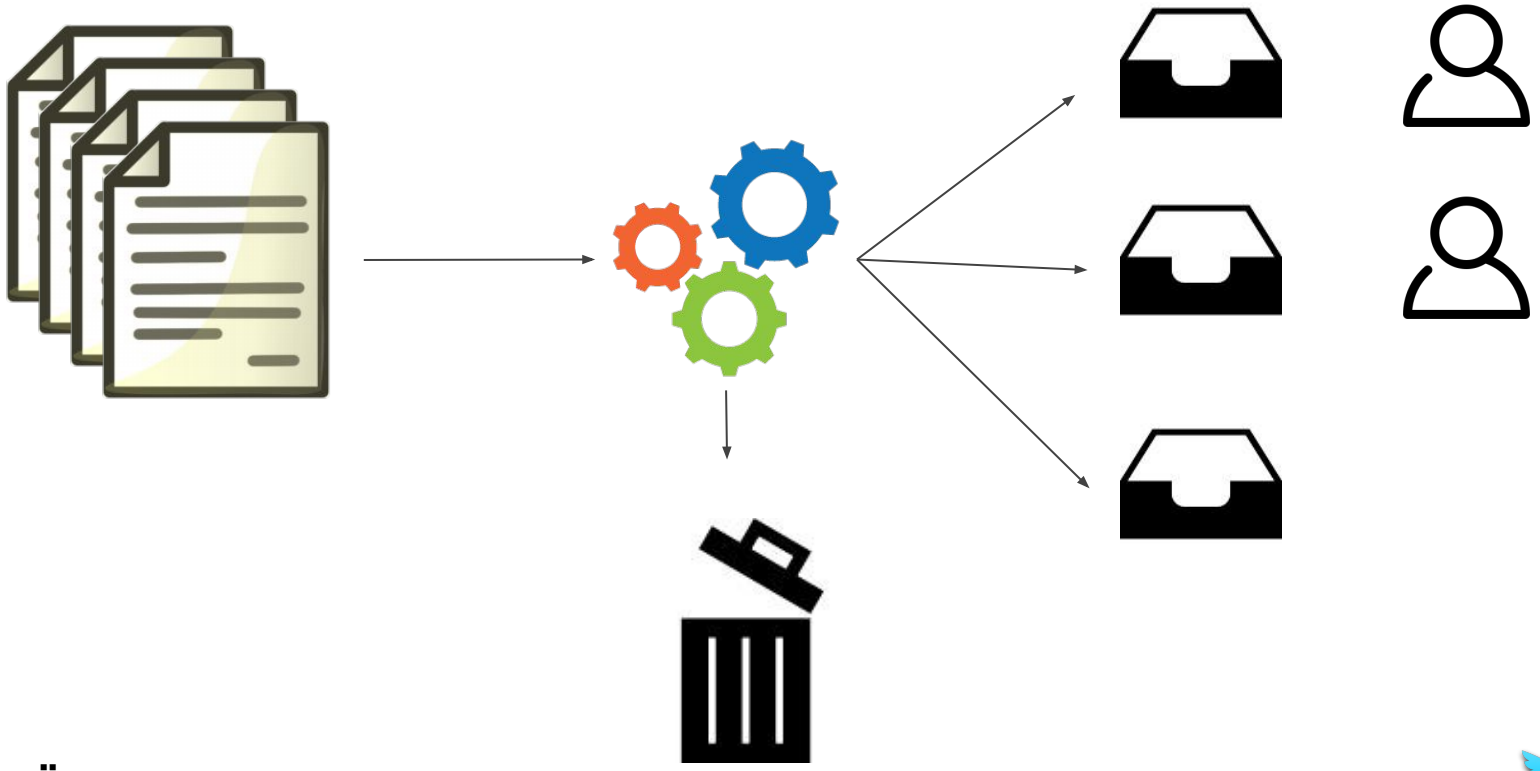



- Encuentra documentos con **palabras parecidas/relacionadas**

# Clasificación de documentos

---

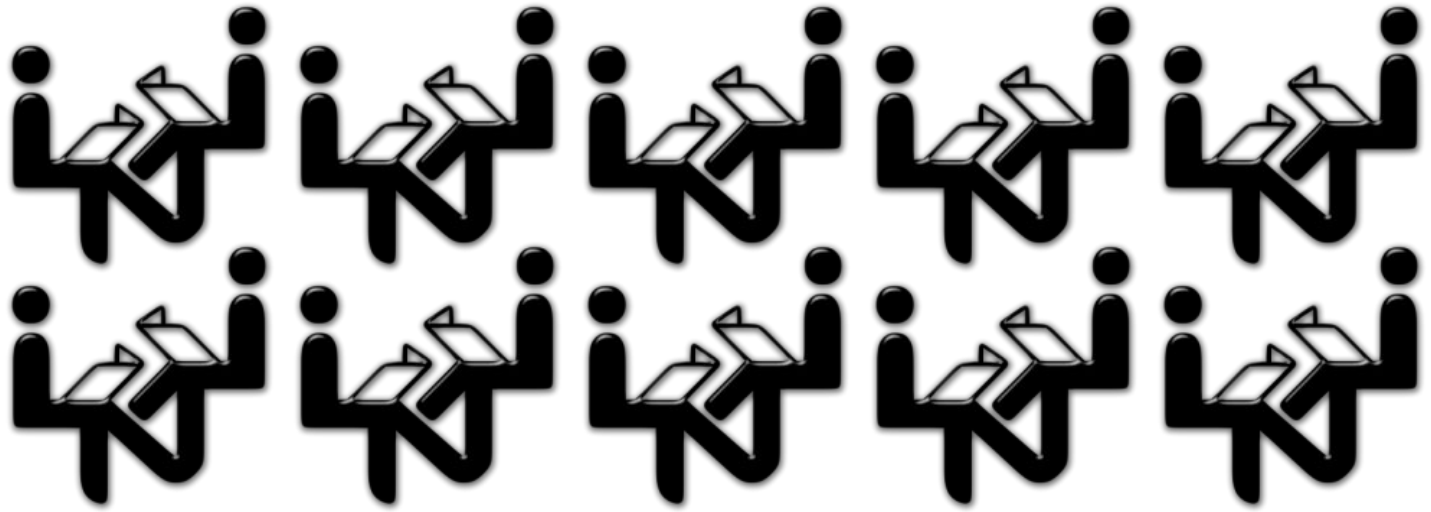
- Cada uno con sus similares: motores de recomendación, clasificación automática en tiempo real,...



# ¿Cómo lo hacen?

---

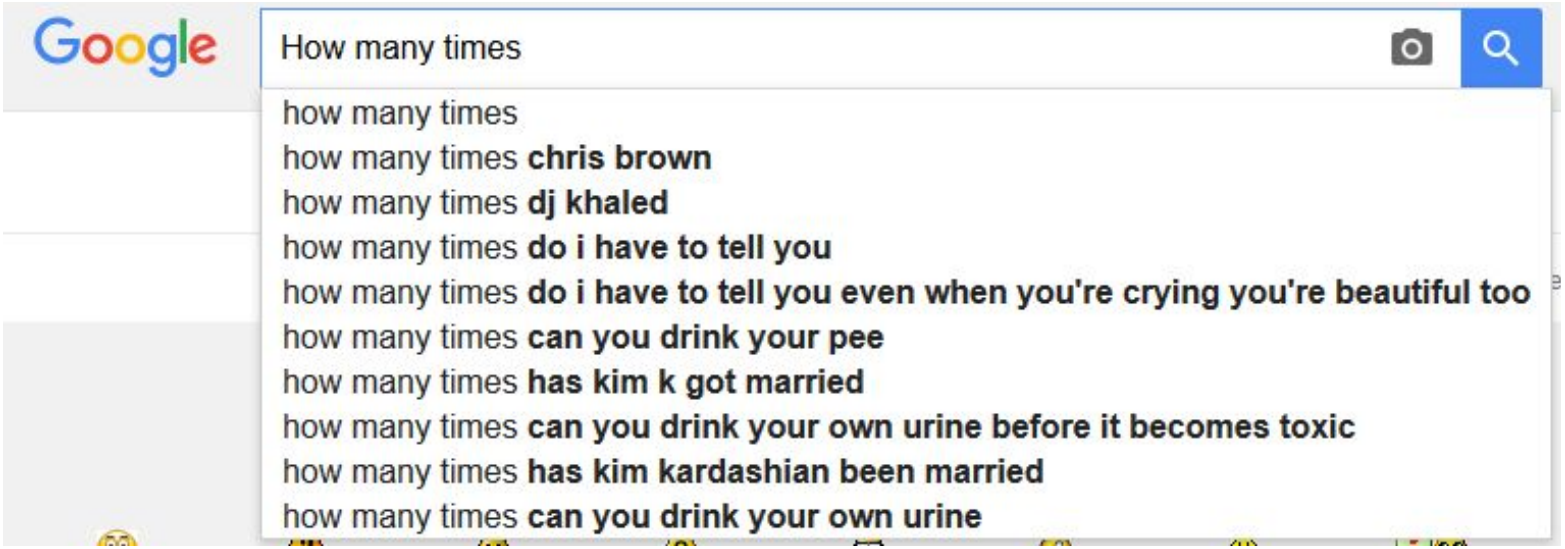
¿Tiene Google un ejército de voraces lectores que responden en tiempo real a las preguntas que hacemos en el navegador?



# ¿Cómo lo hacen?

---

¿Hay (muchísimas) preguntas preparadas?



# Objetivo

Si ya estás motivado, ahora voy a plantearte un reto, ¿te atreves?

# Objetivo

---

Hoy queremos lograr dos cosas:

- Entender la problemática común a los sistemas de recuperación de información

- Enseñar a construir un motor de búsqueda ¡¡!!

# Introducción

Antes de entrar en harina, vamos a repasar algunos conceptos.

Recuperación de info.

Métricas

¿Cuál es el mojo?

---

# Recuperación de información

---

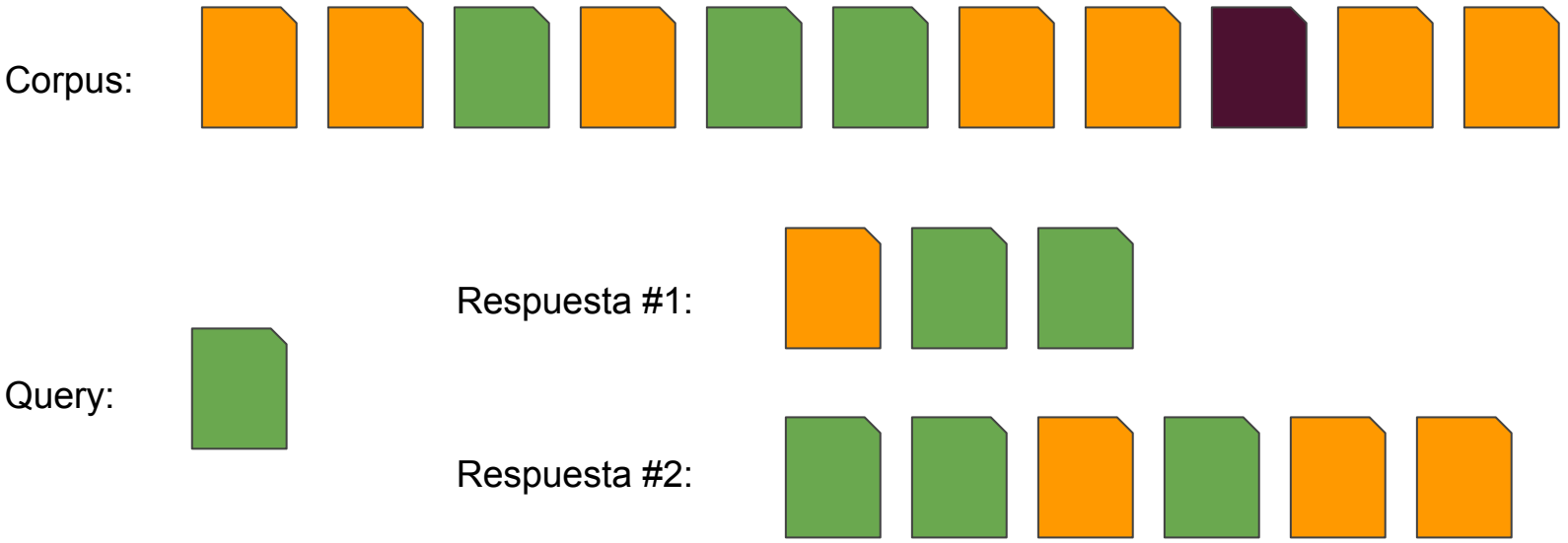
- La **recuperación de información** (information retrieval) trata de buscar documentos dentro de una colección o corpus.

- Su objetivo no es entender el documento ni extraer información del mismo. A eso se le llama **extracción de información** (information extraction), que está muy cerca del data mining.

# Recuperación de información – Métricas

- Precisión

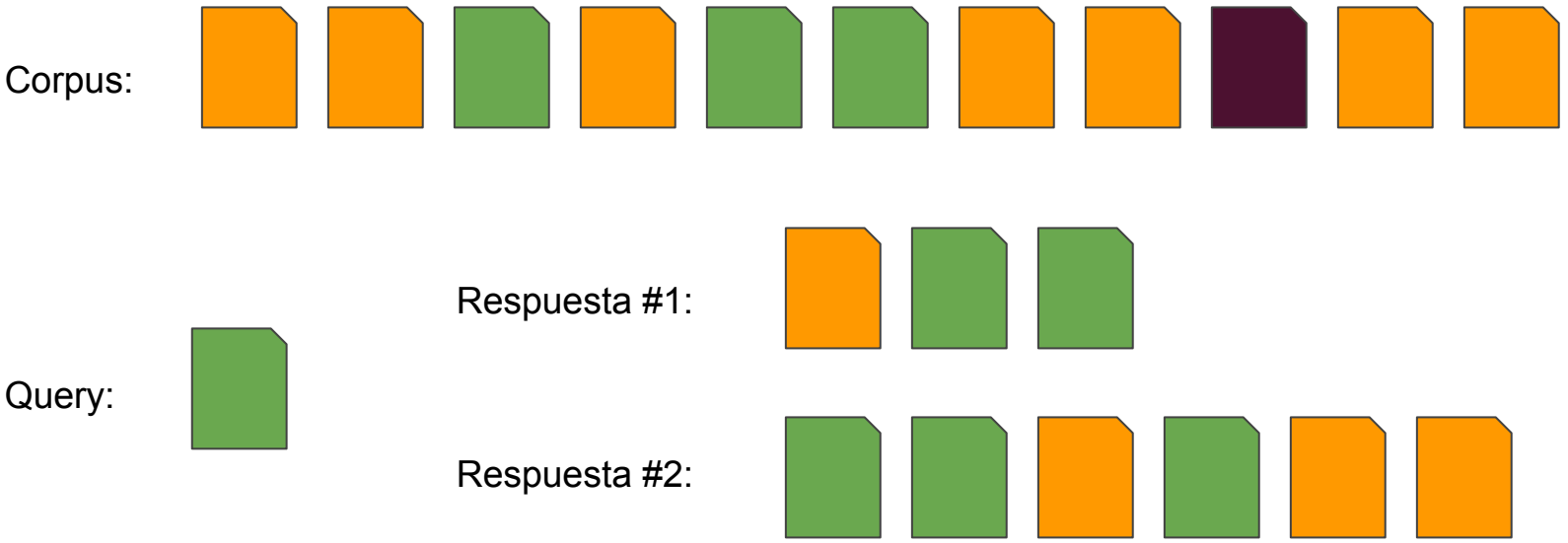
$$\text{Precisión} = \frac{|\{\text{documentos relevantes}\} \cap \{\text{documentos recuperados}\}|}{|\{\text{documentos recuperados}\}|}$$



# Recuperación de información – Métricas

- Exhaustividad/Recall

$$\text{Exhaustividad} = \frac{|\{\text{documentos relevantes}\} \cap \{\text{documentos recuperados}\}|}{|\{\text{documentos relevantes}\}|}$$

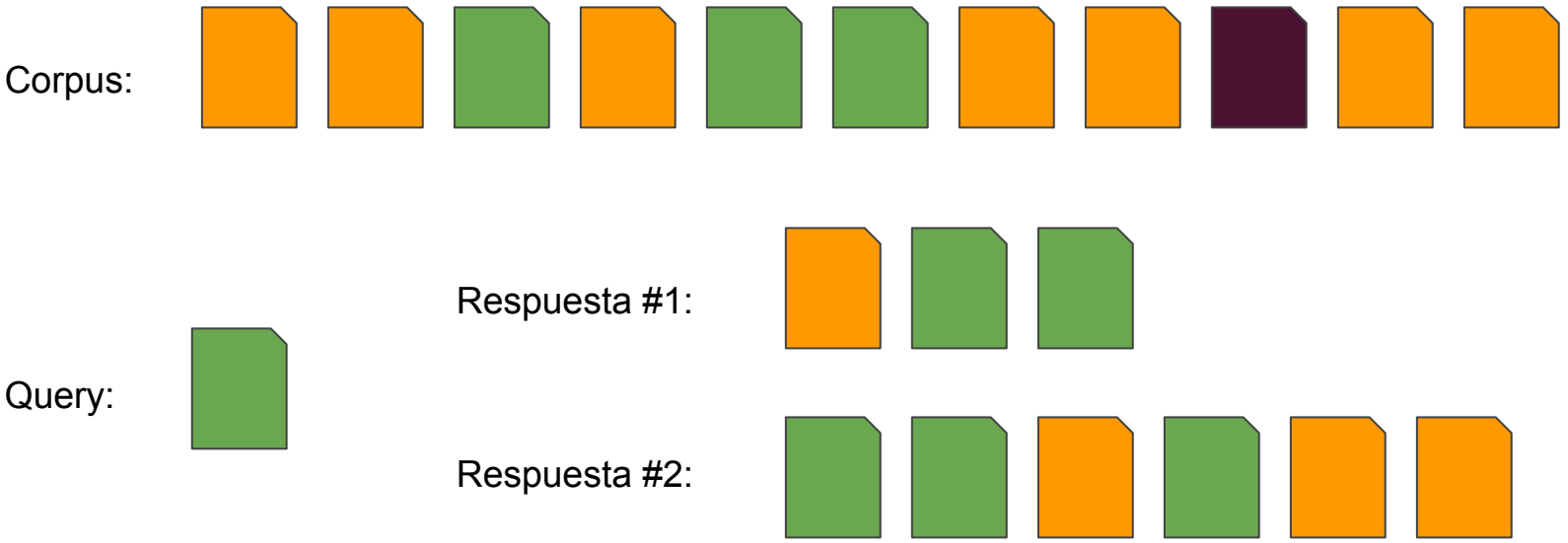


# Recuperación de información – Métricas

---

- F-measure

$$F = \frac{2 \cdot \text{precision} \cdot \text{recall}}{(\text{precision} + \text{recall})}$$

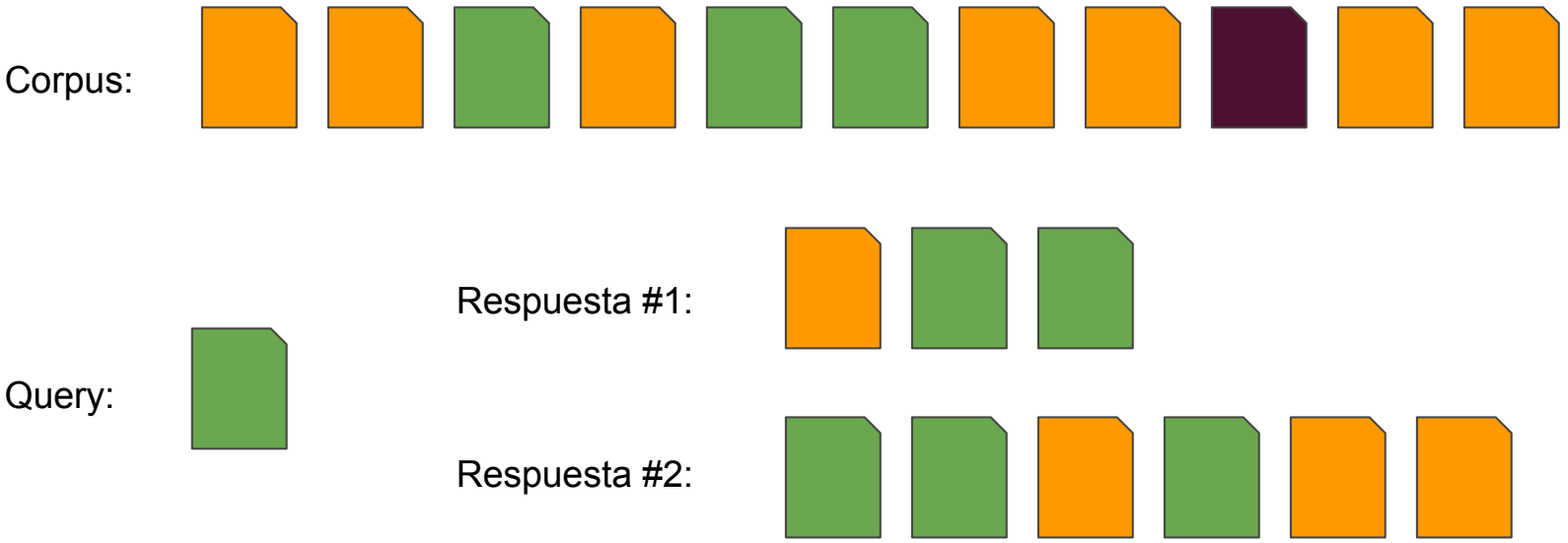


# Recuperación de información – Métricas

---

Otras medidas que también me interesan:

- **El orden**. Quiero los documentos más relevantes primero.

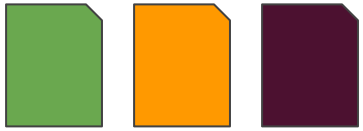


# ¿Cuál es el mojo?

---

- Definir las **características del documento** que son importantes para nosotros (las que van a utilizar los usuarios para buscar).

Color:



Palabras clave:

# ¿Cuál es el mojo?

---

Estas características deben ser:

- baratas de almacenar

- rápidas de comparar

- suficientes/adecuadas

# Representación de un documento

La clave en cualquier sistema de recuperación de información es cómo se representan los documentos.

Documento

Palabras

Base de datos

Vectores

Distancia coseno

Análisis

---

# Caracterizando un documento

---

- Un documento es un conjunto de palabras.

- Mis usuarios van a utilizar palabras para buscar.

- Es fácil leer las palabras de un documento

¡Tiene buena pinta!

# Documento & Query → Palabras

---

- Puedo representar un documento (y la query del usuario) por las palabras que aparecen en él.

Isabel I de Castilla fue reina de Castilla desde 1474 hasta 1504, reina consorte de Sicilia desde 1469 y de Aragón desde 1479, por su matrimonio con Fernando de Aragón

| Forma | Apariciones |
|---|---|
| Isabel | 1 |
| I | 1 |
| de | 4 |
| Castilla | 2 |
| fue | 1 |
| reina | 2 |
| desde | 1 |
| ... | |

# Documento & Query → Palabras → Tabla

---

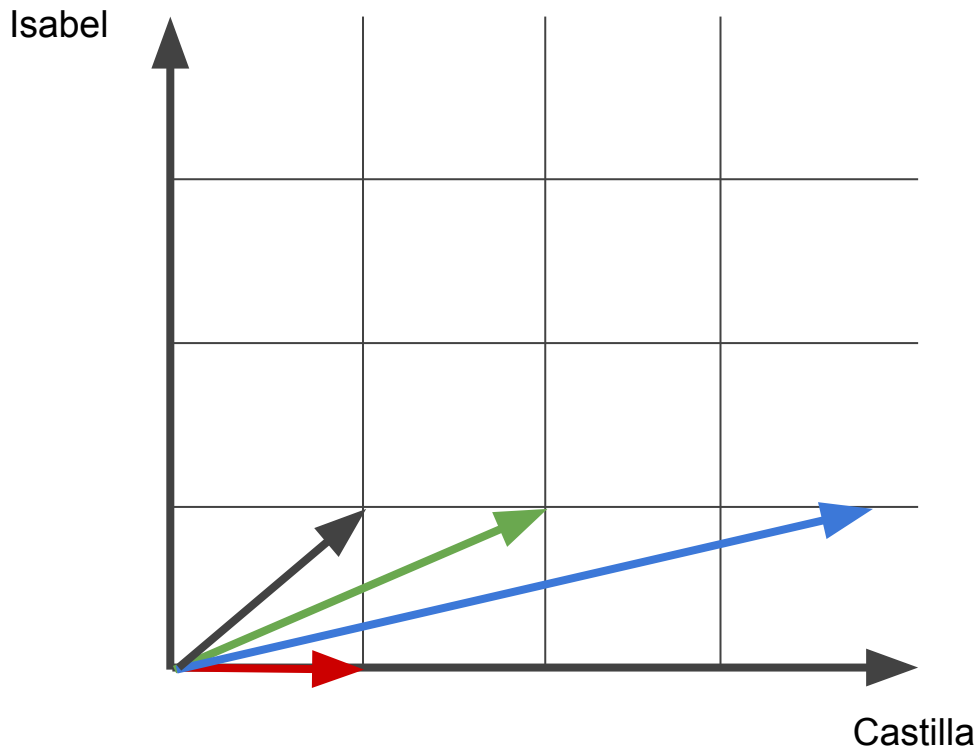
- … y almacenarlo en una tabla en mi base de datos

| | Isabel | de | Castilla | reina | …. |
|---|---|---|---|---|---|
| Doc #1 | 1 | 4 | 2 | 2 | |
| Doc #2 | 0 | 10 | 1 | 4 | |
| ... | | | | | |
| Doc #3 | 1 | 5 | 4 | 0 | |
| ... | | | | | |
| Query | 1 | 0 | 1 | 0 | |

# Búsqueda – documentos como vectores

---

- Ya tengo números, a los ordenadores les gusta trabajar con ellos, ¡algo podré hacer!

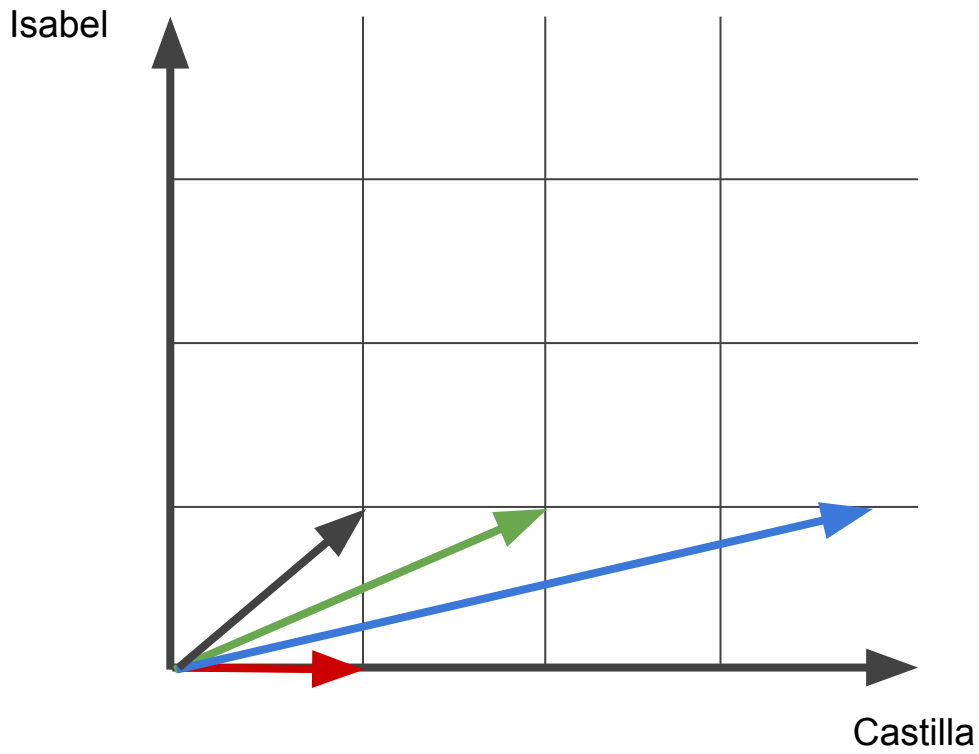



| | Isabel | Castilla |
|---|---|---|
| Doc #1 | 1 | 2 |
| Doc #2 | 0 | 1 |
| ... | | |
| Doc #3 | 1 | 4 |
| ... | | |
| Query | 1 | 1 |

# Búsqueda – distancia/similaridad coseno

---

- Cuanto menor sea el **ángulo** de los vectores, más similares son los documentos




| | Isabel | Castilla |
|---|---|---|
| Doc #1 | 1 | 2 |
| Doc #2 | 0 | 1 |
| ... | | |
| Doc #3 | 1 | 4 |
| ... | | |
| Query | 1 | 1 |

# Análisis – Representación vectorial

---

- ¿Baratas? El conjunto de palabras es finito, pero aún así excesivamente grande y mucho más cuando tenemos un idioma desinencial.

- Objetivo: reducir el conjunto

- Ideas
  - Lematizar: fue, es ⇒ ser
  - Eliminar palabras vacías: de, a, el,...
  - Almacenar sólo sustantivos
  - ¿...?

## ¿Cuál es el mojo?

---

Estas características deben ser:

- baratas de almacenar
- rápidas de comparar
- suficientes/adecuadas

# Análisis – Representación vectorial

---

- ¿Rápidas? Desde luego mucho más que leer todos los documentos cada vez que quiero buscar algo.

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

- Objetivo: simplificar la fórmula

- Ideas:
  - Normalizar vectores para eliminar denom.
  - Trabajar con frecuencias de palabras

## ¿Cuál es el mojo?

---

Estas características deben ser:

- baratas de almacenar
- rápidas de comparar
- suficientes/adecuadas

# Análisis – Representación vectorial

---

- ¿Adecuadas? Sí, pero mejorables.
- Objetivo: dar más importancia

- Ideas:
  - Identificar términos compuestos: mucho mejor “Isabel de Castilla” o “Real Madrid” que sus palabras por separado.
  - Darle más valor a las palabras que aparecen en el título, en negrita,...
  - Valorar más las palabras raras que definen muy específicamente un documento.

## ¿Cuál es el mojo?

---

Estas características deben ser:

- baratas de almacenar
- rápidas de comparar
- suficientes/adecuadas

# TF-IDF

Una representación buena y realmente sencilla: Term frequency - Inverse document frequency

Term frequency

Inverse document frequency

Cálculo

Análisis

---

@jgsogo

# TF - Term frequency

---

- La frecuencia de una forma es el número de veces que aparece en el texto.

Isabel I de Castilla fue reina de Castilla desde 1474 hasta 1504, reina consorte de Sicilia desde 1469 y de Aragón desde 1479, por su matrimonio con Fernando de Aragón

| Term | Count |
|---|---|
| Isabel | 1 |
| I | 1 |
| de | 4 |
| Castilla | 2 |
| fue | 1 |
| reina | 2 |

# IDF - Inverse document frequency

---

- Nos interesa reducir el peso de las palabras que son muy comunes en el conjunto de los documentos frente a las palabras específicas que caracterizan muy bien cada documento.

- Palabras que “elimina”:
  - palabras vacías: preposiciones, artículos,... esta métrica reduce su peso sin necesidad de hacernos una lista.
  - palabras comunes: en un corpus sobre reyes, las palabras rey/reina no serán significativas.

# TF-IDF – ¿Cuál es la fórmula?

---

- TF-IDF: se cálcula como el producto

$$\mathrm{tfidf}(t, d, D) = \mathrm{tf}(t, d) \times \mathrm{idf}(t, D)$$

- tf(t,d): frecuencia del término “t” en el documento “d”.
- idf(t,D): inverse document frequency del término “t” en el corpus de documentos “D” que estamos analizando.

# TF-IDF – Cálculo de tf(t,d)

---

- Para calcular la frecuencia se pueden utilizar algunas variantes:

| **Term** | **$f_{t,d}$** | **Binaria** | **log(1+$f_{t,d}$)** | $0.5 + \frac{0.5 \times f_{t,d}}{\max\{f_{t,d} : t \in d\}}$ |
|---|---|---|---|---|
| Isabel | 1 | 1 | 0,301 | |
| I | 1 | 1 | 0,301 | |
| de | 4 | 1 | 0,699 | |
| Castilla | 2 | 1 | 0,477 | |
| fue | 1 | 1 | 0,301 | |
| reina | 2 | 1 | 0,477 | |
| Real Madrid | 0 | 0 | -- | |

# TF-IDF – Cálculo de idf(t,D)

---

- Para calcular la frecuencia se pueden utilizar algunas variantes:

$$\mathrm{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

- numerador: número de documentos que forman el corpus
- denominador: número de documentos en los que aparece el término “t”.

# Análisis

---

A favor:

- Cálculo sencillo
- Elimina palabras comunes sin necesidad de listas
- Ofrece un valor de similaridad continuo en [0, 1]

En contra:

- El factor idf(t,D) es dependiente del corpus, por lo que cada vez que se añade o elimina (o modifica) un documento hay que recalcular TODOS los números de la tabla.
- No tiene en cuenta la semántica, palabras sinónimas son consideradas distintas
- Los documentos muy largos quedan infrarrepresentados.
- No se presta ninguna atención a concordancia o colocación entre palabras

# Y ahora...

¿Serías capaz de programar un motor de búsqueda?

# En el tintero

Algunos temas relacionados que pueden resultar muy interesantes

# Cosas para hablar

---

Expansión de queries: idealmente un tesauro.

SIRI cambiará la forma en que buscamos → ahora los buscadores empiezan a responder preguntas.