



# DECSAI

Departamento de Ciencias de la Computación e I.A.

Universidad de Granada



## Text Mining

© Fernando Berzal, [berzal@acm.org](mailto:berzal@acm.org)

## Text Mining



- Minería de textos
- Preprocesamiento en minería de textos
  - Preprocesamiento sintáctico
  - Preprocesamiento semántico
  - Reducción de términos
- Asociación en minería de textos
  - Extracción de información
- Agrupamiento en minería de textos
  - Modelado de temas
- Clasificación en minería de textos
  - Análisis de sentimientos
- Esquemas de Winograd



# Minería de textos



Minería de datos sobre bases de datos no estructuradas.

p.ej. bases de datos de documentos, libros, artículos, informes de organizaciones públicas, páginas web, comentarios en redes sociales...

## vs. Recuperación de información

- En los sistemas de recuperación de información, se buscan documentos de acuerdo a unos requisitos.
- En minería de textos, se extrae conocimiento desconocido, no trivial y potencialmente interesante (de forma comprensible para el usuario final).



# Minería de textos



## Beneficios potenciales

- Eficiencia a la hora de generar nuevo conocimiento y encontrar soluciones potencialmente innovadoras.
- Reducción del tiempo necesario de lectura de textos cuando se buscan recursos clave (p.ej. Bioinformática, Medicina, Derecho, investigación científica...).





## Aplicaciones

- Aplicaciones comerciales: gestión de recursos humanos, CRM [customer relationship management], gestión de riesgos, análisis de sentimientos, campañas de publicidad/marketing...
- Aplicaciones en seguridad: lucha contra el fraude, prevención de delitos, lucha antiterrorista...
- Mecanismos de comunicación, desde el filtrado de spam o el uso de asistentes virtuales para gestionar el calendario hasta la creación de chat bots.



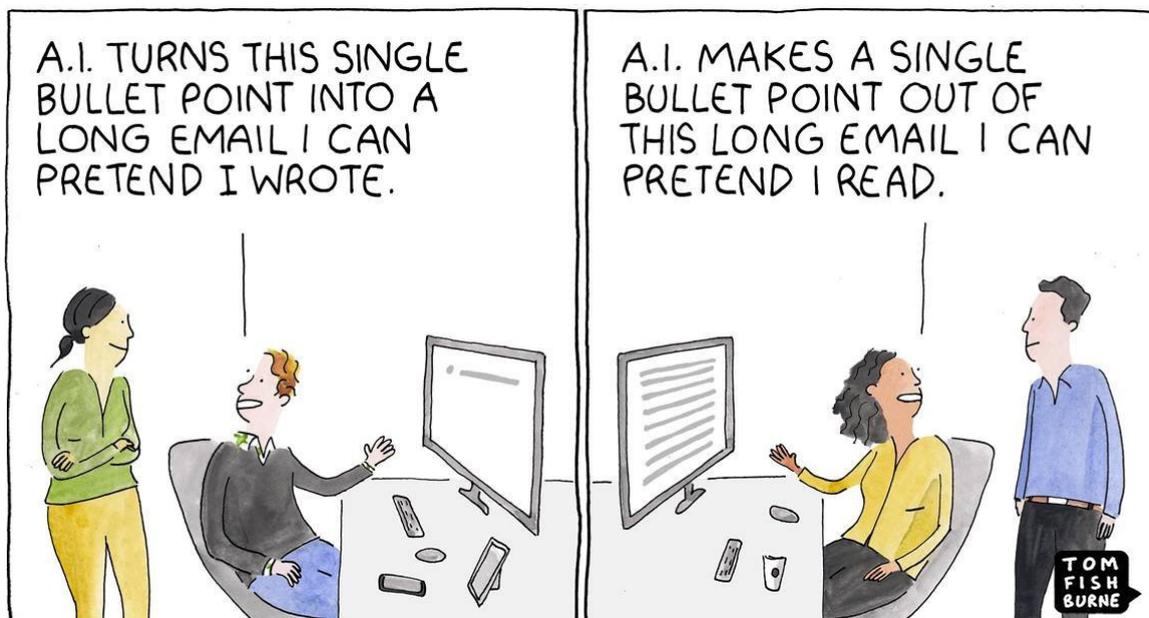
La minería de textos como [pen-]última etapa del procesamiento de textos:

- Preprocesamiento
- Búsqueda de información
- Extracción de información
- **Minería de textos**
- Visualización de resultados



## Técnicas de minería de textos

- Categorización de textos (agrupamiento).
- Clasificación de textos (p.ej. análisis de sentimientos).
- Extracción de información (lingüística computacional).
- Análisis de tendencias y detección de desviaciones.
- Construcción de resúmenes.





## [off-topic] IA generativa

Model: GPT-4

**BS** Act as: Editor  
Degree of revision: Substantial Revision  
Type of edit: Enhance clarity and consistency  
Change style to: Academic, PhD Work  
Change tone to: Analytical  
Change reader comprehension level to: advanced, assume extensive prior knowledge  
Change length to: 1000 Words

My Text:

Brav.. hav u chaked the gpt-4 thing?? U can writ any branles drunken driveli in it and it changes it to any level even PhD hyper brain style.. trulli insane.. you, got to check it bro! Its a tru gift for all of us lliterate assholes around the the global flat plane! Now we can have universit jobs.. hhah crazi stuff...

 Title: A Comprehensive Analysis of the GPT-4 Language Model and its Potential Impact on Scholarly Communication

Abstract: This article explores the functionality and capabilities of OpenAI's GPT-4, a sophisticated language model capable of generating coherent and semantically rich text. We examine the model's potential to transform written communication, particularly for those with limited literacy skills, by analyzing the model's capacity for tailoring text to various academic and linguistic requirements. Furthermore, we discuss the implications of the model's wide-ranging applications, including its potential to democratize access to higher education and professional opportunities.

Introduction



# Preprocesamiento de textos



A diferencia de las técnicas clásicas de minería de datos, que trabajan con datos perfectamente estructurados, en minería de textos:

- Cada documento de texto es una colección ordenada de palabras (términos) y signos de puntuación con significado asociado cuya posición en el texto proviene de restricciones sintácticas y semánticas.
- Aunque puedan tener cierta estructura implícita, de mucha mayor riqueza que los datos estructurados, los textos son inherentemente desestructurados.
- Los textos incluyen ambigüedades propias del lenguaje natural y pueden estar escritos en diferentes idiomas y dialectos..



# Preprocesamiento de textos



La dificultad de procesar textos directamente nos obliga en la práctica a utilizar representaciones intermedias:

- Bolsas de términos [bags of words].
- Estructuras matriciales (p.ej. embeddings).
- Grafos conceptuales, redes semánticas y ontologías.

El preprocesamiento de textos se encarga de transformar datos no estructurados en datos estructurados (transformar los textos de forma que se puedan tratar de forma automatizada).



# Preprocesamiento sintáctico



Para obtener una bolsa de términos [bag of words], i.e. el conjunto de términos que aparecen en cada documento, tal vez anotados (p.ej. con su frecuencia):

- Tokenización
- Lematización [stemming]
- Eliminación de signos de puntuación
- Eliminación de palabras "vacías" [stop words]
- Reconocimiento de n-gramas
- Reconocimiento de tipos sintácticos [POS]





## Tokenización

Como en el análisis léxico de un compilador, se divide el texto en tokens, cadenas de caracteres que representen palabras.

- Extracción del texto de un documento (HTML, PDF...).
- Segmentación del texto extraído

## Problemas

- Varios tokens pueden representar el mismo término.
- Palabras “compuestas” pueden requerir un preprocesamiento específico (p.ej. nombres propios, nombres de compuestos químicos...)



## Lematización

Las diferentes formas ortográficas de un mismo término (plurales, formas verbales, derivaciones) pueden resultar problemáticas para el análisis automatizado de textos.

p.ej. aprender, aprendizaje, aprende... → [aprend](#)

En inglés existen algoritmos (p.ej. Porter [1980]) que obtienen buenos resultados. En español, no tanto :-)

## Problemas

La lematización no siempre es útil ni recomendable.





## Signos de puntuación

Hay que diferenciar cuándo un signo ortográfico es realmente de puntuación (un token independiente) y se usa como delimitador, frente a situaciones en las que forma parte de un término.

p.ej. - ' , . ; pueden ser delimitadores o formar parte de términos (fechas, números, nombres compuestos...).

- En cada idioma, el uso de los signos de puntuación puede variar.
- Lo más habitual en muchas aplicaciones es eliminarlos y no considerarlos en el análisis de los textos.



## Palabras “vacías” [stop words]

Palabras que no proporcionan demasiada información desde un punto de vista no lingüístico.

- Desempeñan un papel meramente funcional (p.ej. artículos y proposiciones).
- A menudo, se utiliza una lista de “stop words” que directamente se eliminan de los textos.
- Obviamente, dependen del idioma.





## **N-gramas**

En un texto, existen grupos de palabras que tienen significado por sí mismos y deberían tratarse como un único término.

p.ej. base de datos, sistema operativo...

Existen algoritmos de detección automática basados en la idea de que el n-grama es mucho más frecuente de lo esperado (como secuencia de palabras independientes).

**Idea:** Algoritmo Apriori



## **Reconocimiento de tipos sintácticos**

### **POS [Part-of-Speech] Tagging**

En ciertas aplicaciones, es necesario identificar las funciones sintácticas de los términos (nombre, adjetivo, determinante, adverbio, verbo...) y etiquetar cada término con su función sintáctica.

### **Algoritmos**

- Basados en reglas lingüísticas (antiguos y limitados)
- Basados en técnicas de aprendizaje automático (NLP)



# Preprocesamiento semántico



Una vez limpios y anotados los términos desde el punto de vista sintáctico, se pueden utilizar relaciones de tipo semántico:

- **Sinonimia:** Distinta forma, mismo significado.  
p.ej. clase, lección
- **Homonimia:** Misma forma, distinto significado.  
p.ej. banco (finanzas, geografía fluvial, para sentarse)
- **Polisemia:** Misma forma, distintos significados relacionados, p.ej. banco & banco de sangre.
- **Hiponimia:** Una palabra es una subclase de otra.  
p.ej. perro, mascota, animal, ser vivo, entidad.



# Preprocesamiento semántico



## Desambiguación

### [WSD: Word Sense Disambiguation]

Proceso mediante el cual se asigna a cada término el sentido concreto con el que se utiliza en el texto.

p.ej. <https://wordnet.princeton.edu/>

WordNet Search - 3.1  
- [WordNet home page](#) - [Glossary](#) - [Help](#)

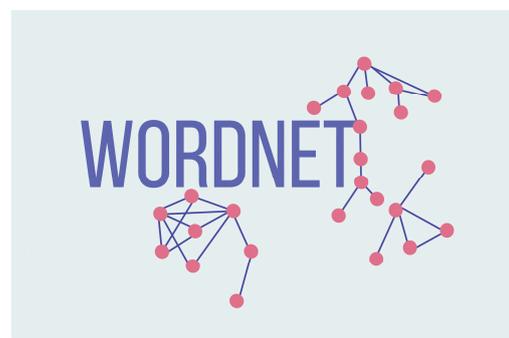
Word to search for:

Display Options:

Key: "S." = Show Synset (semantic) relations, "W." = Show Word (lexical) relations  
Display options for sense: (gloss) "an example sentence"

**Noun**

- [S. \(n\) wordnet](#) (any of the machine-readable lexical databases modeled after the Princeton WordNet)
- [S. \(n\) WordNet, Princeton WordNet](#) (a machine-readable lexical database organized by meanings; developed at Princeton University)



# Reducción de términos



## Diccionario de términos

Conjunto de términos de una colección de documentos.

Cada documento...

- viene caracterizado por los términos que contiene.
- se representa basándonos en el diccionario.

Las técnicas de reducción de términos tienen como objetivo reducir el número de términos del diccionario.

NOTA: Se puede crear un término artificial [OOD/OOV: out of dictionary/vocabulary] para representar términos desconocidos o poco frecuentes que no queremos tratar por separado.



# Reducción de términos



## Algoritmos

- **Técnicas directas de preprocesamiento:**  
lematización, eliminación de stop words, reconocimiento de n-gramas, eliminación de tipos sintácticos no significativos [POS tagging], desambiguación de sentidos...
- **Técnicas basadas en medidas de importancia:**  
Se estima la importancia de los distintos términos en nuestra colección de documentos y nos quedamos sólo con los k más importantes o un porcentaje sobre el total.



# Reducción de términos



## Medidas de importancia

Término  $k$  en el documento  $i$  con  $n_i$  términos (de  $N$ )

- Frecuencia de un término en un documento

$$w_{ik} = f_{ik} w_k$$

- TF\*IDF

[Term Frequency \* Inverse Document Frequency]

$$w_{ik} = tf * idf_k = f_{ik} * \log_2(N/d_k)$$

- Ruido

$$w_{ik} = f_{ik} s_k \quad s_k = \log_2 t_k - n_k \quad n_k = \sum f_{ik}/t_k \log_2 f_{ik}/t_k$$

- Similitud entre documentos

$$w_{ik} = f_{ik} \delta_k \quad \delta_k = \sigma_k - \sigma$$



# Reducción de términos



## Algoritmos

- **Selección por componentes principales:**

PCA sobre los vectores que representan cada documento (pesos de cada término en el documento).

- **Análisis de semántica latente:**

Representación en función de factores, combinaciones de términos relacionados entre sí.

- **Word & text embeddings**

Técnicas basadas en Deep Learning.

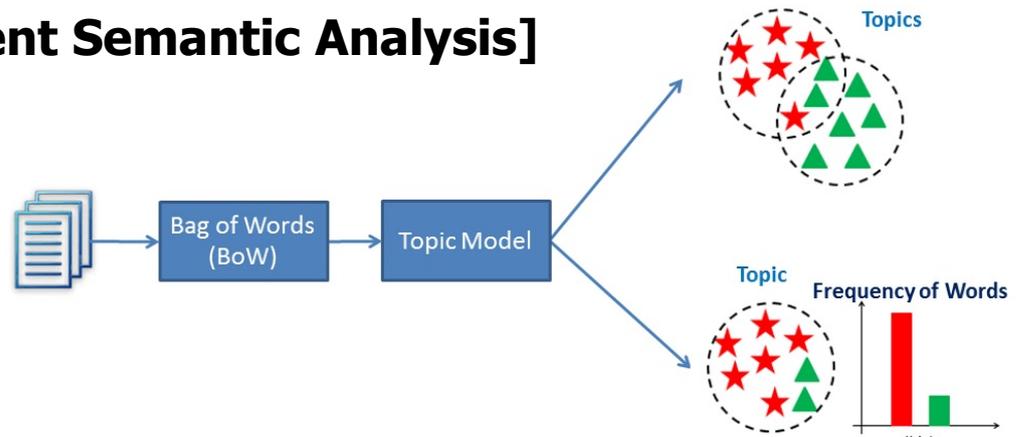


# Reducción de términos



## Análisis de semántica latente

### [LSA: Latent Semantic Analysis]



Term Document Matrix

	Doc-1	Doc-2	Doc-3	Doc-4
Term-1				
Term-2				
Term-3				
Term-4				

Word Assignment to Topics

	Topic-1	Topic-2
Term-1		
Term-2		
Term-3		
Term-4		

Topic Importance

	Topic-1	Topic-2
Topic-1		
Topic-2		

Topic Distribution Across Documents

	Doc-1	Doc-2	Doc-3	Doc-4
Topic-1				
Topic-2				

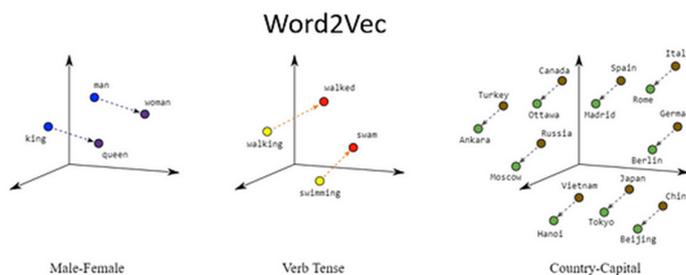


# Reducción de términos

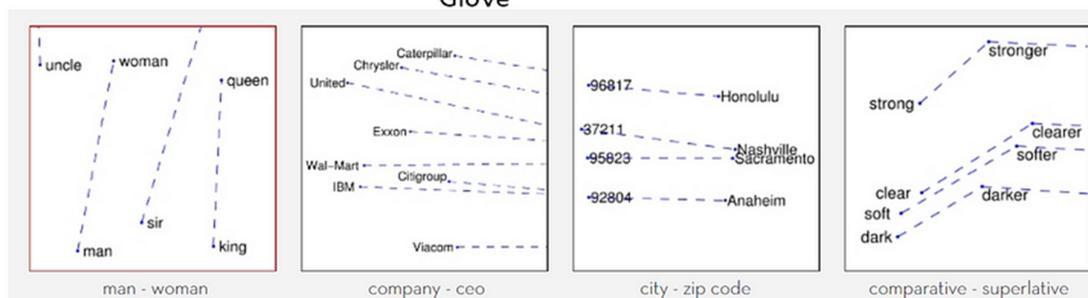


## Word embeddings

Word2Vec (Google, 2013) & GloVe (Stanford, 2014)



GloVe



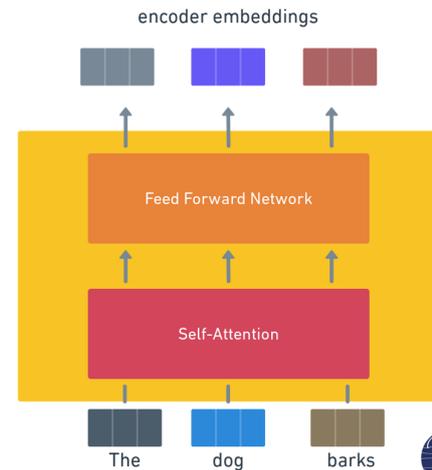
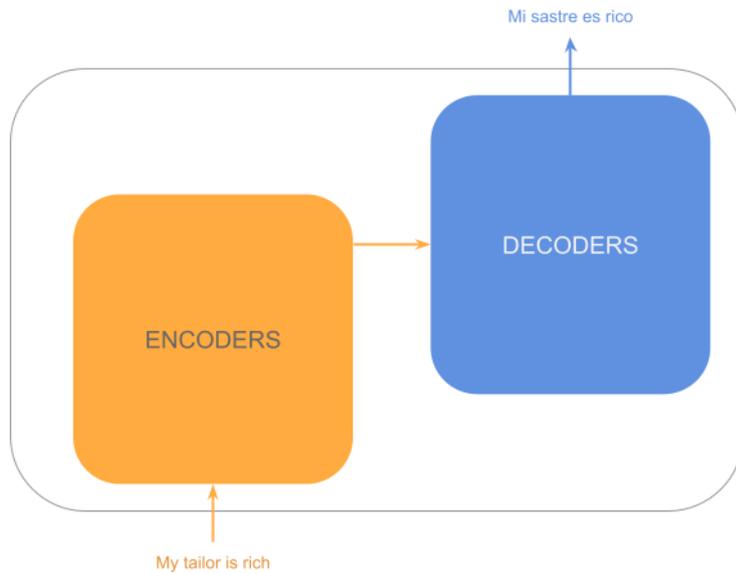
# Reducción de términos



## Text embeddings

### @ LLMs [Large Language Models]

Transformers, p.ej. BERT (Google, 2018)



# Asociación en TM



## Preprocesamiento de textos

- Identificación de n-gramas

## Extracción de información

- Reconocimiento de entidades
- Resolución de referencias
- Extracción de relaciones entre términos





## Identificación de n-gramas [a.k.a. phrase mining] Segmentación mejorada de textos

- No supervisada (p.ej. ToPMine): Identificación de frases frecuentes (secuencias con PrefixSpan).
- Débilmente supervisada (p.ej. SegPhrase o AutoPhrase), con algunos datos previamente etiquetados



FIGURE 5.23

Autophrase: automated phrase mining by distant supervision.



## Identificación de n-gramas [a.k.a. phrase mining] Segmentación mejorada de textos

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
1-grams	problem algorithm optimal solution search solve constraints programming heuristic genetic	word language text speech system recognition character translation sentences grammar	data method algorithm learning clustering classification based features proposed classifier	programming language code type object implementation system compiler java data	data patterns mining rules set event time association stream large
n-grams	genetic algorithm optimization problem solve this problem optimal solution evolutionary algorithm local search search space optimization algorithm search algorithm objective function	natural language speech recognition language model natural language processing machine translation recognition system context free grammars sign language recognition rate character recognition	data sets support vector machine learning algorithm machine learning feature selection paper we propose clustering algorithm decision tree proposed method training data	programming language source code object oriented type system data structure program execution run time code generation object oriented programming java programs	data mining data sets data streams association rules data collection time series data analysis mining algorithms spatio temporal frequent itemsets

FIGURE 5.21

Five topics from a 50-topic run of ToPMine on a full DBLP abstracts data set. Overall we see coherent topics and high-quality topical phrases, which can be interpreted as “search/optimization,” “NLP,” “machine learning,” “programming languages,” and “data mining.”





## IE [Information extraction]

- Reconocimiento de entidades [NER: Named Entity Recognition]: personas, entidades, localizaciones...
- Resolución de correferencias [coreference resolution]: Anáforas y referencias a entidades.
- Extracción de relaciones [RE: Relation Extraction]: Identificación de relaciones entre entidades.

p.ej. Extracción de eventos  
(verbos y objetos asociados)



## Agrupamiento en TM



- Agrupamiento de documentos (p.ej. similitud del coseno)
- Agrupamiento de términos: modelado de temas (p.ej. Brown clustering, word embeddings...)



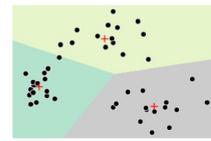
# Agrupamiento de textos



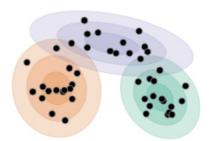
## Text clustering

p.ej. en recuperación de información

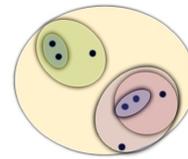
- K-means
- Clustering jerárquico
- DBSCAN
- GMM [Gaussian Mixture Model]
- LDA [Latent Dirichlet Allocation]
- PLSA [Probabilistic Latent Semantic Analysis]
- ...



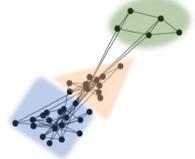
K-means clustering



Mixture model (Gaussian)



Hierarchical clustering



Graph based clustering



# Agrupamiento de textos



## Aplicación: Descubrimiento de sucesos

Identificación de "relatos"  
correspondientes a sucesos nuevos,  
previamente no identificados.



p.ej. Google News (desde 2002)

Providing supplemental information in news search

Abstract

In one aspect, a method includes receiving a query, identifying search results in response to the query, grouping the search results into one or more clusters, identifying similarities across the one or more clusters, determining a related topic associated with the query based on the similarities of the one or more clusters, determining a categories associated with the query, identifying supplemental information based on the related topic and the category, and providing the search results and the supplemental information for display in response to the query.

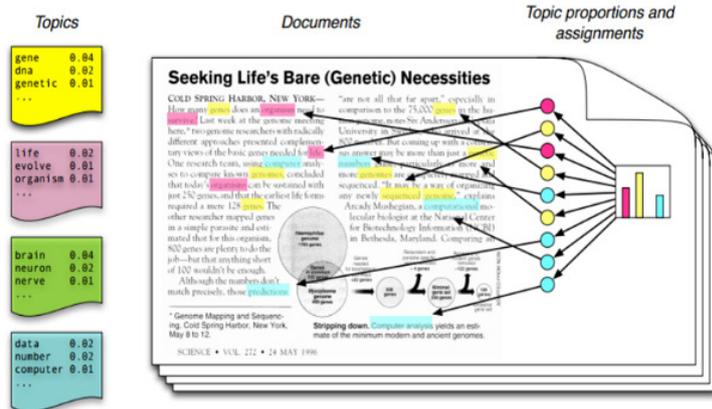


# Modelado de temas



## Topic Modeling

o cómo encontrar estructuras semánticas ocultas en el texto en forma de conjuntos de "temas" (clusters de palabras similares)



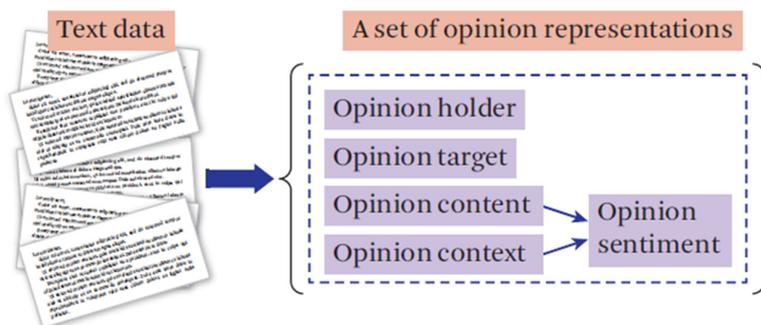
- LDA [Latent Dirichlet Allocation]
- PLSA [Probabilistic Latent Semantic Analysis]



# Clasificación en TM



- Categorización de textos (con categorías predefinidas), p.ej. k-NN, Naive Bayes, regresión logística, clasificadores lineales, SVMs, deep learning...
- Análisis de sentimientos y minería de opiniones (caso particular de clasificación)

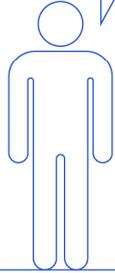


# Análisis de sentimientos



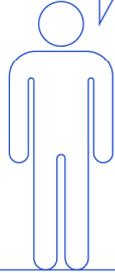
Propositional:

This looks like a perfect plan!



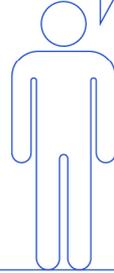
Embedded:

I love being ignored.



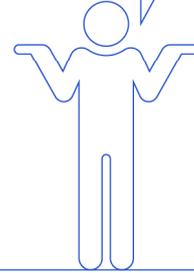
Like-prefixed:

Like those guys believe a word they say.



Illocutionary:

Very helpful indeed!



## Desafíos

- Detección de sarcasmos
- Detección de distintos tipos de negaciones
- Ambigüedad del lenguaje natural
- Multipolaridad



# Análisis de sentimientos



**Roshan Agarwal** @roshansxc  
@IndiGo6E  
Thank you for sending my baggage to Hyd and flying me to Calcutta at the same time. Brilliant service. #DielIndigo  
12:20 PM · 30 Apr 17

**IndiGo** @IndiGo6E · 2h  
Replying to @roshansxc  
Glad to hear that. #KeepFlying6E.

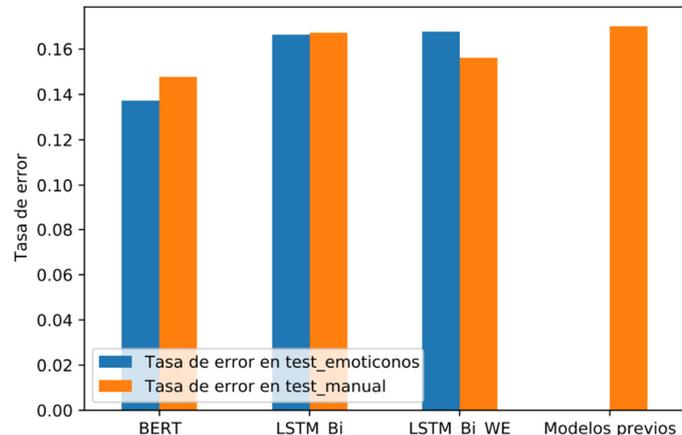
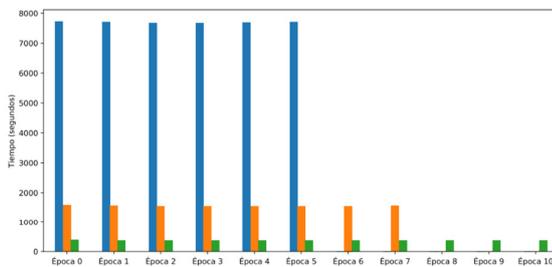
**Roshan Agarwal** @roshansxc · 2h  
are you serious?





## Análisis de sentimientos con transformers / LLMs

Modelo	Nº de parámetros total	Nº de parámetros ajustables	Nº de parámetros "congelados"
Modelo 1: LSTM_Bi	6,242,945	6,242,945	0
Modelo 2: LSTM_Bi_WE	4,884,609	84,609	4,800,000
Modelo 3: BERT	108,311,810	108,311,810	0



TFG Jorge Sánchez González, 2020



## Esquemas de Winograd



Los concejales de la ciudad negaron a los manifestantes la autorización porque [temían/defendían] la violencia.

El trofeo no cabe dentro de la maleta porque es muy [grande/pequeño].

**Problema de NLP:** A primera vista, los esquemas de Winograd parecen requerir, simplemente, la resolución de anáforas (identificar el antecedente de un pronombre ambiguo en una declaración)... pero para resolverlos hace falta emplear conocimiento y sentido común.



# Esquemas de Winograd



## Hey Siri

MIT  
Technology  
Review

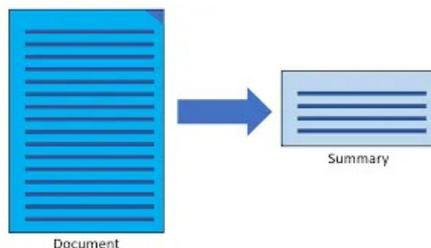
Secciones • Las 10 Tecnologías Emergentes

Inteligencia Artificial

**Un examen más duro  
que el de Turing  
revela que las  
máquinas aún son  
estúpidas**



# Resumen de textos



- **Resumen extractivo** (basado en selección):  
El resumen es una secuencia de frases extraídas del texto original.
- **Resumen abstractivo** (basado en generación):  
El resumen puede contener frases que no estaban en el texto original.

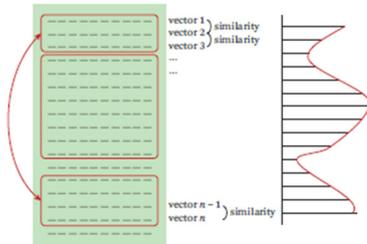


# Resumen de textos

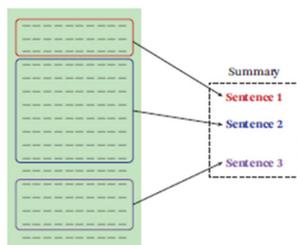


## Resumen extractivo

- Se segmenta el texto en pasajes



- Se elige una frase de cada pasaje:



# Resumen de textos



## Resumen abstractivo

p.ej. LLM [Large Language Models]

### ChatGPT

Examples	Capabilities	Limitations
"Explain quantum computing in simple terms" →	Remembers what user said earlier in the conversation	May occasionally generate incorrect information
"Got any creative ideas for a 10 year old's birthday?" →	Allows user to provide follow-up corrections	May occasionally produce harmful instructions or biased content

Character was based on truth. Even so, lack of evidence has led most researchers to discontinue, putting the claims down to clever marketing. Despite this, many people still believe in its authenticity – author Peter Haining published a book claiming that Todd was a real, revenge-seeking man. However, no evidence supports his claims and the Old Bailey holds no records of any such crime. Whether or not the story does take its inspiration from any real event or rumour is uncertain. Yet, whatever the truth, the tale of Sweeney Todd continues to hold a macabre fascination, nearly two centuries on.

- Summarize in 100 words

ChatGPT Mar 23 Version. Free Research Preview. ChatGPT may produce inaccurate information about people, places, or facts



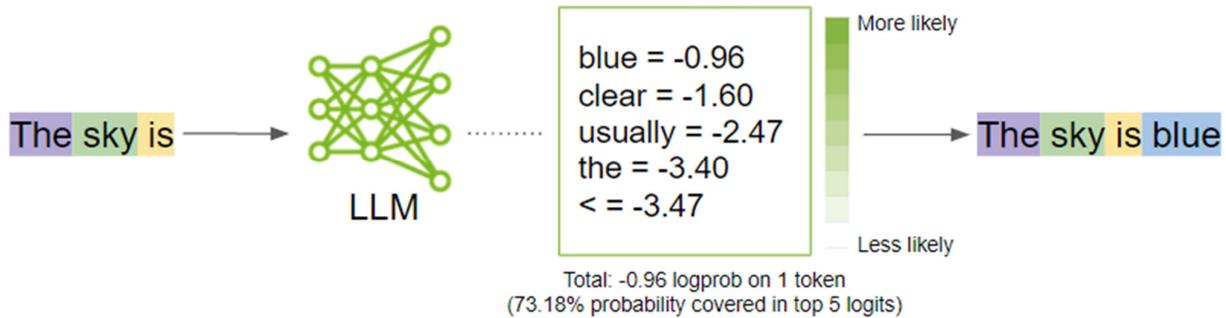
ChatGPT: get instant answers, find creative inspiration, and learn something new. ●



# LLM [Large Language Models]

## Red neuronal

Predicción de la siguiente palabra en una secuencia



## IDEA

Aprender a predecir la siguiente palabra obliga a la red a aprender mucho acerca del mundo...



# LLM [Large Language Models]

## Entrenamiento del modelo base

El LLM "comprime" el contenido de Internet



p.ej. Llama 2 70B (Facebook), <https://ai.meta.com/llama/>

Andrej Karpathy: "Intro to LLMs", November 2023



# LLM [Large Language Models]

## Resultado

Una red que "sueña" documentos (alucina en el argot ;-)

<pre>package lecture06;  import java.util.*;  public class Field {      public static Map&lt;String, Card&gt; cardIndex = new HashMap&lt;String, Card&gt;();      public void addCard(Card card){         this.cardIndex.put(card.getRegistrationNumber(),card);     }      public Card getCard(String registrationNumber) {         return this.cardIndex.get(registrationNumber);     }      public int getCardCount() {         return this.cardIndex.size();     } }</pre>	<p>Hades Heroes</p> <p>Author: Maureen Fergus</p> <p>Product Details:</p> <p>ISBN: 9780545946767</p> <p>Format: Hardback</p> <p>Pages: 176</p> <p>Dims (mm): 140 x 223</p> <p>Pub Date: 01-05-16</p> <p>Pub Country: United States</p> <p>Condition: NEW</p> <p>Description:</p> <p>Work hard. Love. Make your own luck. These are the keys to success in life at school. But when a student named Hercules wanders in and challenges him</p>	<p>Blacknose dace</p> <p>The blacknose dace (<i>Rhinichthys atratulus</i>) is a small freshwater fish found in North America. They are typically found in streams, rivers, and lakes, and are widely distributed throughout the eastern and central United States and Canada.</p> <p>These fish have a slender body with a black stripe running horizontally along their sides, giving them their distinctive "blacknose" appearance. They have a flattened head with a pointed snout and small, round eyes. Their color can vary from dark brown to olive green on their back and sides, with a silvery-white underside.</p> <p>Blacknose dace are omnivorous and feed on a variety of small aquatic insects, crustaceans, and plant matter. They are an important food source for larger predators such as bass,</p>
--	---	--

Java code dream

Amazon product dream (?)

Wikipedia article dream

Andrej Karpathy: "Intro to LLMs", November 2023



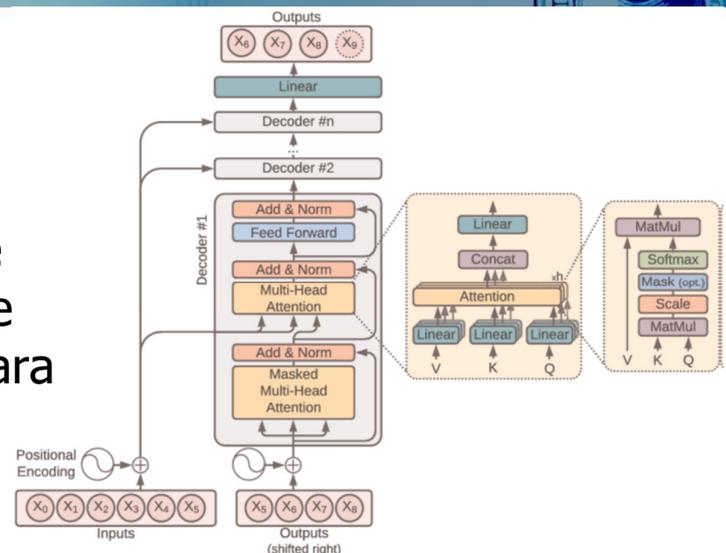
# LLM [Large Language Models]

## Resultado

¿Cómo funciona?

Red con cientos de miles de millones de parámetros, que ajustamos iterativamente para mejorar las predicciones...

... no se sabe realmente cómo "colaboran" todos esos parámetros.



Andrej Karpathy: "Intro to LLMs", November 2023



# LLM [Large Language Models]

## Resultado

En cierto modo, almacena lo que ve, pero de manera imperfecta (compresión con pérdidas).

Recent viral example: "reversal curse"

Q: "Who is Tom Cruise's mother"?

A: Mary Lee Pfeiffer ✓

Q: "Who is Mary Lee Pfeiffer's son?"

A: I don't know ✗



De ahí el "prompt engineering"...



Pedro Domingos ✓  
@pmddomingos

...

Prompt engineering is like trying to shape someone's dreams by telling them things during the day.



Pedro Domingos ✓  
@pmddomingos

...

With enough prompt engineering, you don't even need the LLM.



# LLM [Large Language Models]

## Refinamiento [fine-tuning]: Asistentes digitales

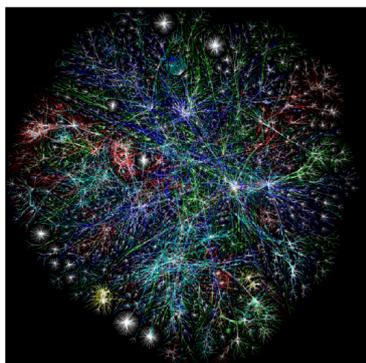
Se reemplaza el conjunto de entrenamiento (i.e. Internet) por uno preparado manualmente...

### Training the Assistant

(Just swap the dataset, then continue training)



written by  
people:



<USER>

Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research.

<ASSISTANT>

"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions...

~100K conversations

Andrej Karpathy: "Intro to LLMs", November 2023



# LLM [Large Language Models]

## Resultado: Asistentes digitales

Ya tenemos un chat bot...

<USER>

Can you help me with this code? It seems like there is a bug.  
print("hello world")

Andrej Karpathy: "Intro to LLMs", November 2023



# LLM [Large Language Models]

## Resumen: Cómo crear tu propio ChatGPT



every  
~year

### Stage 1: Pretraining

1. Download ~10TB of text.
2. Get a cluster of ~6,000 GPUs.
3. Compress the text into a neural network, pay ~\$2M, wait ~12 days.
4. Obtain **base model**.

Knowledge



<USER>  
Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research.

<ASSISTANT>  
"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions...

every  
~week

### Stage 2: Finetuning

1. Write labeling instructions
2. Hire people (or use [scale.ai](#)!), collect 100K high quality ideal Q&A responses, and/or comparisons.
3. Finetune base model on this data, wait ~1 day.
4. Obtain **assistant model**.
5. Run a lot of evaluations.
6. Deploy.
7. Monitor, collect misbehaviors, go to step 1.

Alignment

Andrej Karpathy: "Intro to LLMs", November 2023



# LLM [Large Language Models]

## Refinamiento [fine-tuning]: Aprendizaje por refuerzo

Es mucho más sencillo comparar respuestas que crearlas...

You  
Write a haiku about paperclips

ChatGPT  
Steel curves intertwine,  
Holding life's pages in line,  
Quiet, small design.

You  
Write a haiku about paperclips

ChatGPT  
Silver arcs entwined,  
Holding life's pages, aligned,  
In steel we find, bind.

< 12 / 12 >

You  
Write a haiku about paperclips

ChatGPT  
Silver arcs entwined,  
Humble joiner of pages,  
Quiet strength, aligned.

< 2 / 2 >

e.g. OpenAI RLHF [Reinforcement Learning from Human Feedback]

Andrej Karpathy: "Intro to LLMs", November 2023



# LLM [Large Language Models]

## Ranking ELO

<https://huggingface.co/spaces/lmsys/chatbot-arena-leaderboard>

Model	▲ ★ Arena Elo rating	▲ 🚩 MT-bench (score)	▲ MMLU	▲ License
<a href="#">GPT-4-Turbo</a>	1210	9.32		Proprietary
<a href="#">GPT-4</a>	1159	8.99	86.4	Proprietary
<a href="#">Claude-1</a>	1146	7.9	77	Proprietary
<a href="#">Claude-2</a>	1125	8.06	78.5	Proprietary
<a href="#">Claude-instant-1</a>	1106	7.85	73.4	Proprietary
<a href="#">GPT-3.5-turbo</a>	1103	7.94	70	Proprietary
<a href="#">WizardLM-70b-v1.0</a>	1093	7.71	63.7	Llama 2 Community
<a href="#">Vicuna-33B</a>	1090	7.12	59.2	Non-commercial
<a href="#">OpenChat-3.5</a>	1070	7.81	64.3	Apache-2.0
<a href="#">Llama-2-70b-chat</a>	1065	6.86	63	Llama 2 Community
<a href="#">WizardLM-13b-v1.2</a>	1047	7.2	52.7	Llama 2 Community
<a href="#">zephyr-7b-beta</a>	1042	7.34	61.4	MIT





# LLM [Large Language Models]

## Regulación...

### Grading Foundation Model Providers' Compliance with the Draft EU AI Act

Source: Stanford Research on Foundation Models (CRFM), Institute for Human-Centered Artificial Intelligence (HAI)

	OpenAI	cohere	stability.ai	ANTHROPIC	Google	BigScience	Meta	AI21 labs	ALPHA	ELI	Totals
Draft AI Act Requirements	GPT-4	Cohere Command	Stable Diffusion v2	Claude	PaLM 2	BLOOM	LLaMA	Jurassic-2	Luminous	GPT-NeoX	
Data sources	●○○○	●●●○	●●●●	○○○○	●●○○	●●●●	●●●●	○○○○	○○○○	●●●●	22
Data governance	●●○○	●●●○	●●○○	○○○○	●●○○	●●●●	●●●●	○○○○	○○○○	●●●●	19
Copyrighted data	○○○○	○○○○	○○○○	○○○○	○○○○	●●○○	○○○○	○○○○	○○○○	●●●●	7
Compute	○○○○	○○○○	●●●●	○○○○	○○○○	●●●●	●●●●	○○○○	●○○○	●●●●	17
Energy	○○○○	●○○○	●●●●	○○○○	○○○○	●●●●	●●●●	○○○○	○○○○	●●●●	16
Capabilities & limitations	●●●●	●●●○	●●●●	○○○○	●●●●	●●○○	●●○○	●○○○	●○○○	●●●●	27
Risks & mitigations	●●●○	●●○○	●○○○	●○○○	●●●●	●●○○	●○○○	●○○○	○○○○	●○○○	16
Evaluations	●●●●	●●○○	○○○○	○○○○	●●○○	●●○○	●○○○	○○○○	●○○○	●○○○	15
Testing	●●●○	●○○○	○○○○	○○○○	●●○○	●○○○	○○○○	●○○○	○○○○	○○○○	10
Machine-generated content	●●●○	●●●○	○○○○	●●●○	●●○○	●●○○	○○○○	●●○○	●○○○	●●○○	21
Member states	●○○○	○○○○	○○○○	●○○○	●●●●	○○○○	○○○○	○○○○	●○○○	○○○○	9
Downstream documentation	●●○○	●●○○	●●●●	○○○○	●●○○	●○○○	○○○○	○○○○	○○○○	●●○○	24
Totals	25 / 48	23 / 48	22 / 48	7 / 48	27 / 48	36 / 48	21 / 48	8 / 48	5 / 48	29 / 48	



# LLM [Large Language Models]

## Modelos multimodales...

### DALL·E

<https://openai.com/dall-e-3>

### Midjourney

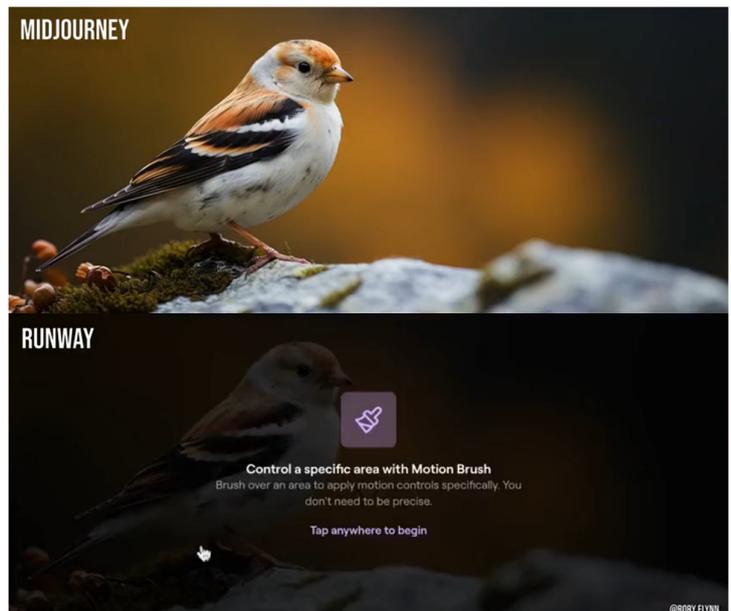
<https://www.midjourney.com/>

### Runway

<https://runwayml.com/>

### Pika

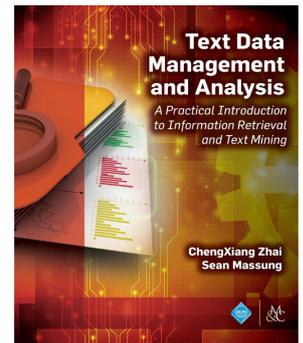
<https://pika.art/>



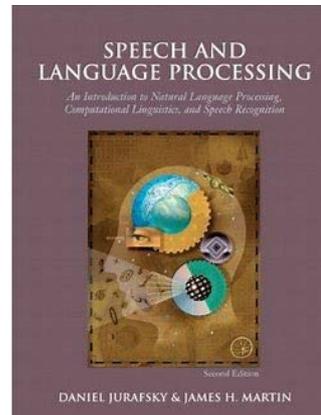
# Bibliografía



- ChengXiang Zhai & Sean Massung:  
**Text Data Management and Analysis:  
A Practical Introduction  
to Information Retrieval  
and Text Mining,**  
ACM Books, 2016  
ISBN 197000116X



- Daniel Jurafsky  
& James H. Martin:  
**Speech and Language Processing: An Introduction  
to Natural Language Processing, Computational  
Linguistics, and Speech Recognition,**  
2<sup>nd</sup> edition, Prentice Hall, 2008.  
ISBN 0131873210



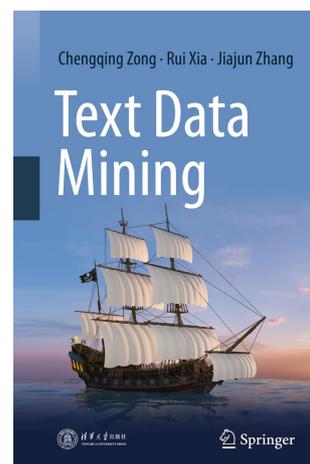
# Bibliografía



- Charu C. Aggarwal &  
ChengXiang Zhai (editors):  
**Mining Text Data,**  
Springer, 2012  
ISBN 1461432227



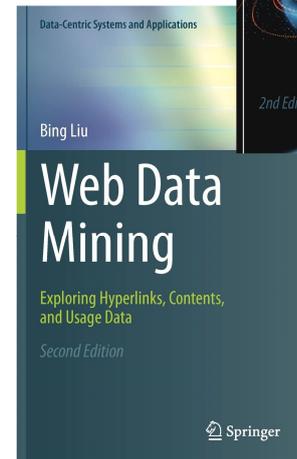
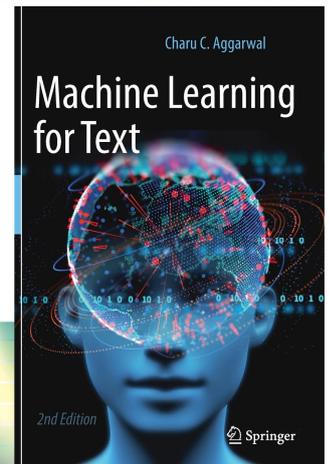
- Chengqing Zong, Rui Xia  
& Jiajun Zhang:  
**Text Data Mining,**  
Springer, 2021.  
ISBN



# Bibliografía



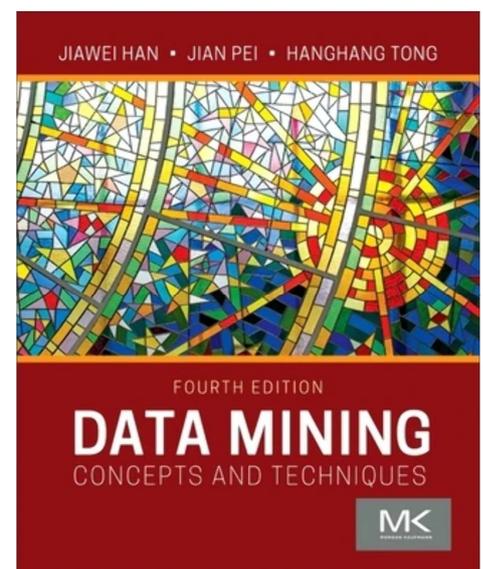
- Charu C. Aggarwal:  
**Machine Learning for Text**  
2<sup>nd</sup> edition, Springer, 2022  
ISBN 3030966224
- Bing Liu:  
**Web Data Mining:  
Exploring Hyperlinks,  
Contents, and Usage Data,**  
2<sup>nd</sup> edition, Springer, 2011  
ISBN 3642194591



# Bibliografía



- Jiawei Han,  
Jian Pei &  
Hanghang Tong:  
**Data Mining:  
Concepts and Techniques,**  
4<sup>th</sup> edition, Morgan Kaufmann, 2022.  
ISBN 0128117605



- 12.1.1 Mining text data
- 12.2.1 Data mining for sentiment and opinion
- 12.2.2 Truth discovery and misinformation identification
- 5.6.1 Phrase mining in massive text data





Pang-Ning Tan,  
Michael Steinbach,  
Vipin Kumar &  
Anuj Karpatne:  
**Introduction to Data Mining,**  
2<sup>nd</sup> edition, Addison Wesley, 2018.  
ISBN 0133128903

