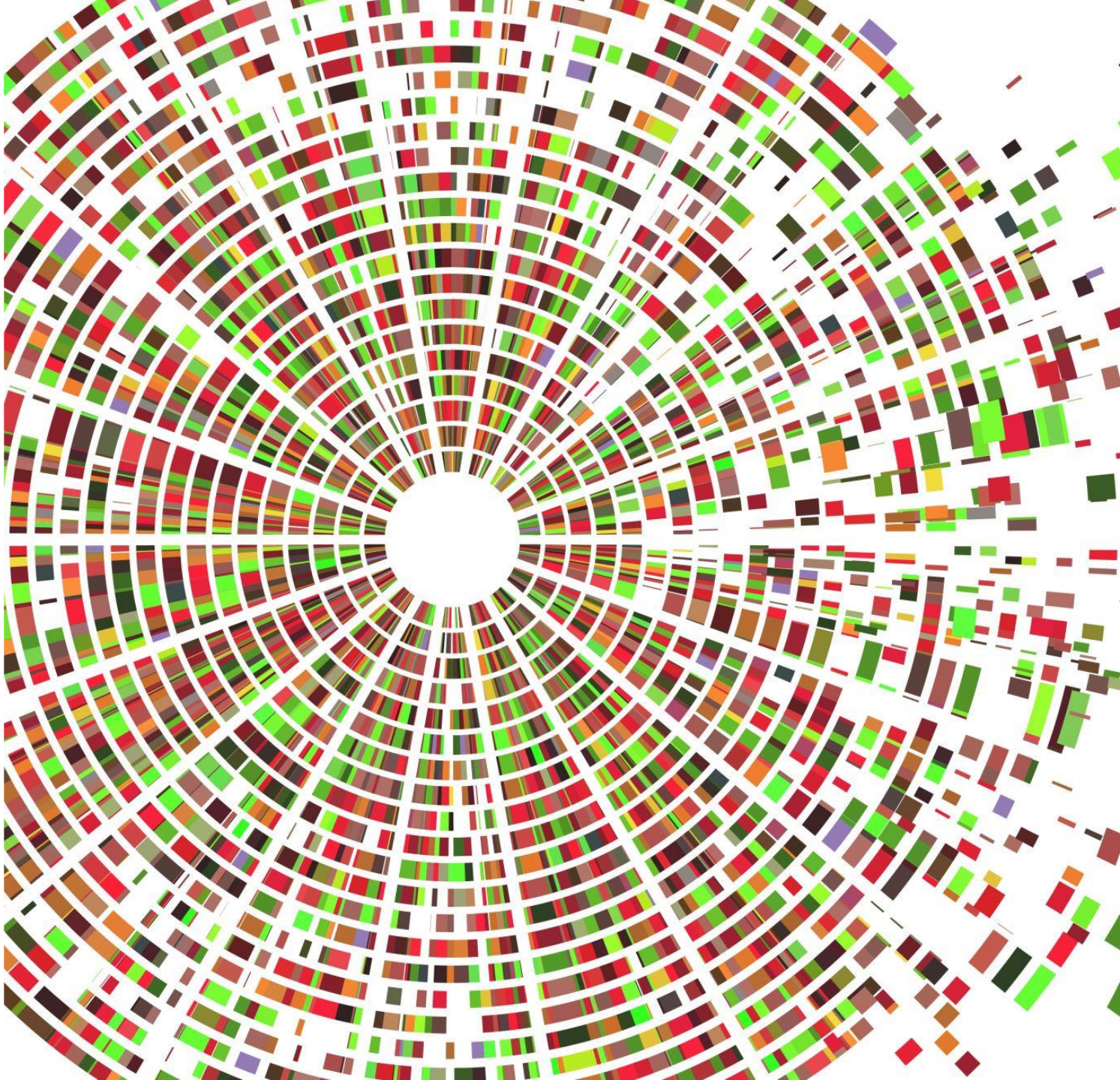
Métodos avanzados de ciencia de datos

Prof. Emily Díaz



Contenido

Análisis de texto: introducción

Casos de uso y diferencias en redes neuronales

Técnicas de análisis de texto, BoW y embeddings

Análisis de texto

¿Qué es el lenguaje?

- Sistema de comunicación que utiliza símbolos, sonidos o signos para expresar ideas, emociones, deseos o pensamientos
- Conjunto de palabras y reglas gramaticales que las personas emplean para comunicarse entre sí

Familias de lenguajes

Algunos datos interesantes sobre las familias

El lenguaje más hablado en el mundo es..

El lenguaje más viejo (conocido) es…

La familia más grande de lenguajes es…

El lenguaje con mayor número de palabras es…

Algunos datos interesantes sobre las familias

El lenguaje más hablado en el mundo es..
Mandarín (1.1B de hablantes nativos)

El lenguaje más viejo (conocido) es... **sumerio, hablando en la antigua Mesopotamia (textos de aproximadamente 3500 AC)**

La familia más grande de lenguajes es...**Niger-Congo con 1500 lenguajes. Incluye Swahili, Yoruba,etc.**

El lenguaje con mayor número de palabras es...
Inglés, principalmente por adoptar palabras de otros idiomas a lo largo de la historia.

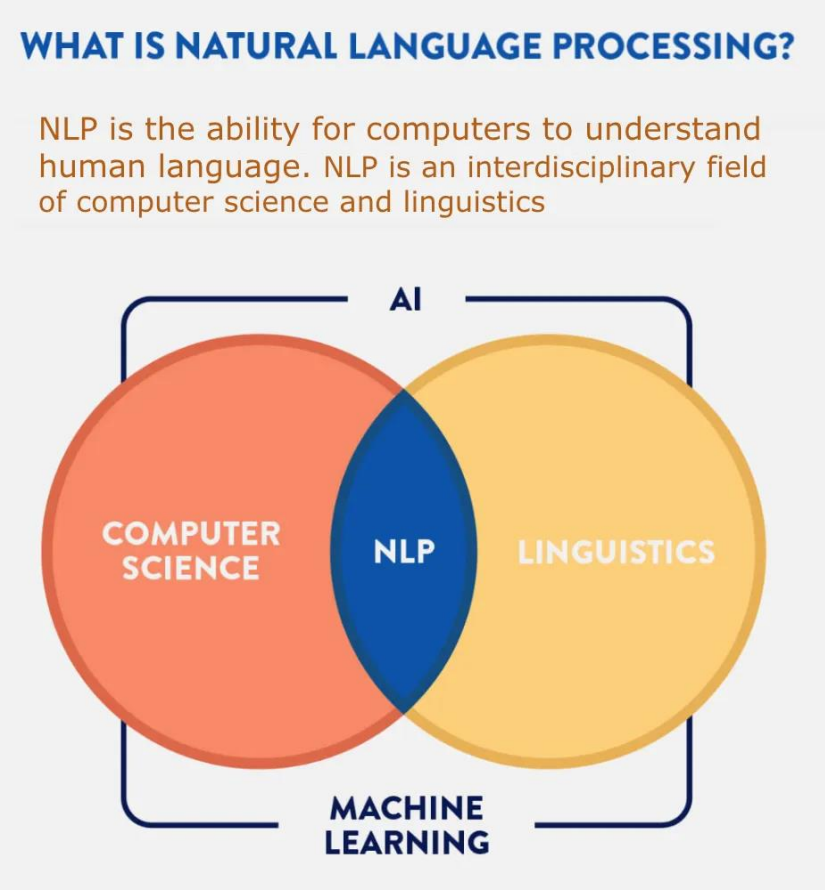
Algunos elementos del lenguaje importantes en el análisis de texto

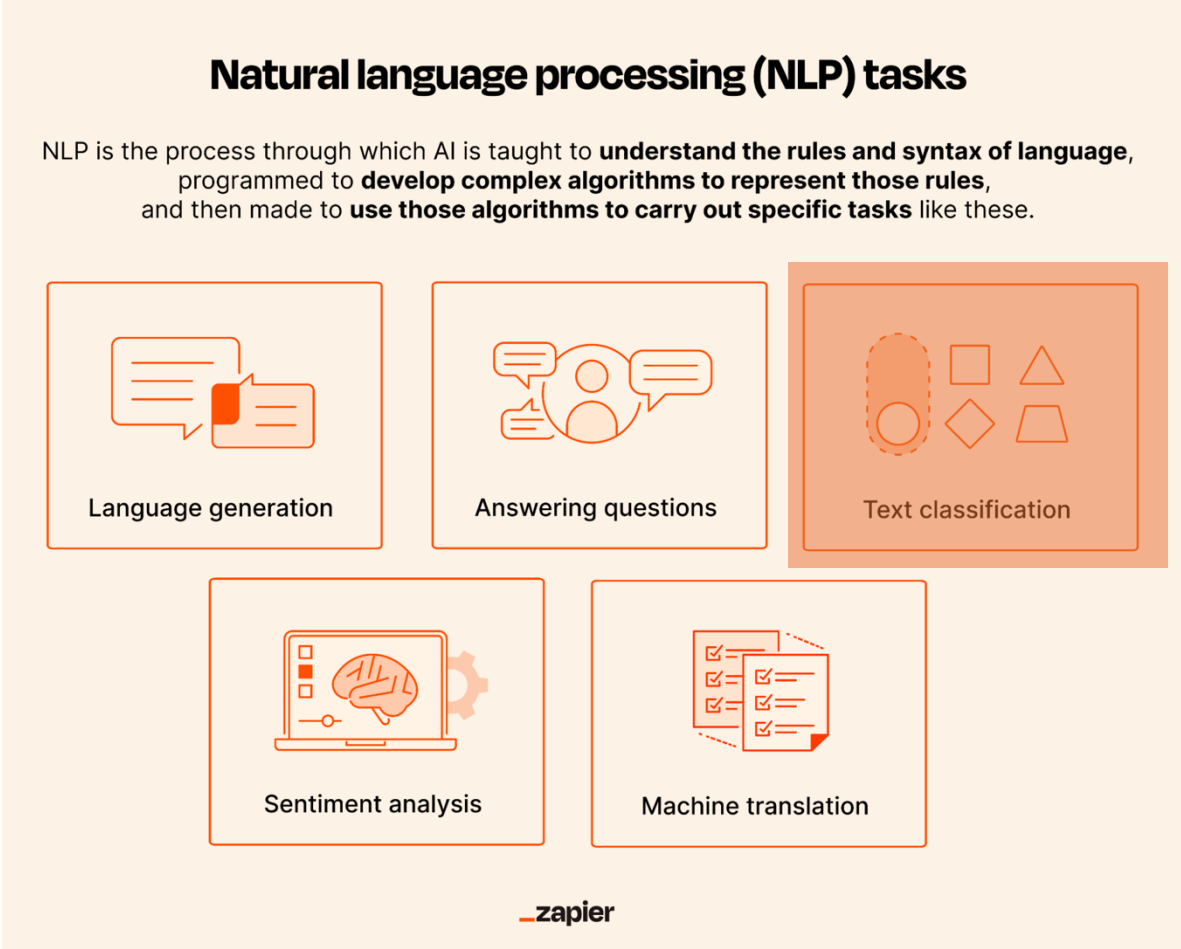
- **Morfología:**
  - Es el estudio de la **estructura de las palabras** y de cómo se forman. Se centra en los morfemas, las unidades significativas más pequeñas de un idioma, como las raíces, los prefijos y los sufijos
  - Técnicas como ***stemming*** y ***lemmatization*** reducen palabrasa su forma base y es parte de la normalización de texto en NLP
- **Sintaxis:**
  - Se ocupa de la **estructura de las oraciones** y de las reglas que rigen la disposición de palabras y frases para crear oraciones significativas
  - Técnicas como ***parsing*** (descomponer oraciones en sus componentes gramaticales), **etiquetado de partes del discurso y análisis de dependencia**, que ayuda a las máquinas a comprender cómo se relacionan entre sí las diferentes partes de una oración
- **Semántica:**
  - Se centra en el **significado de las palabras, frases y oraciones**. Trata de cómo se interpretan las palabras y cómo se deriva el significado de las combinaciones de palabras
  - Técnicas como **traducción automática, preguntas y respuestas, análisis de sentimientos** y **desambiguación del sentido** de las palabras se centran en este tema. Ayuda a los sistemas de procesamiento del lenguaje natural a interpretar el contexto y el significado pretendido detrás de las palabras.

Análisis de texto en el contexto de NLP

- Procesamiento del lenguaje natural (NLP) es un subcampo de la inteligencia artificial (IA) y la lingüística computacional que se centra en la **interacción entre las computadoras y el lenguaje humano**
- El objetivo principal es permitir que las **computadoras comprendan, interpreten y generen lenguaje humano** de una manera que sea significativa y útil
- Hay diferentes componentes en NLP:
  - Procesamiento de texto (limpieza, tonenización, etc)
  - Entendimiento de texto (semántica, sintaxis, etc)
  - Generación de texto (resumen, traducción, generación de dialogo, etc)

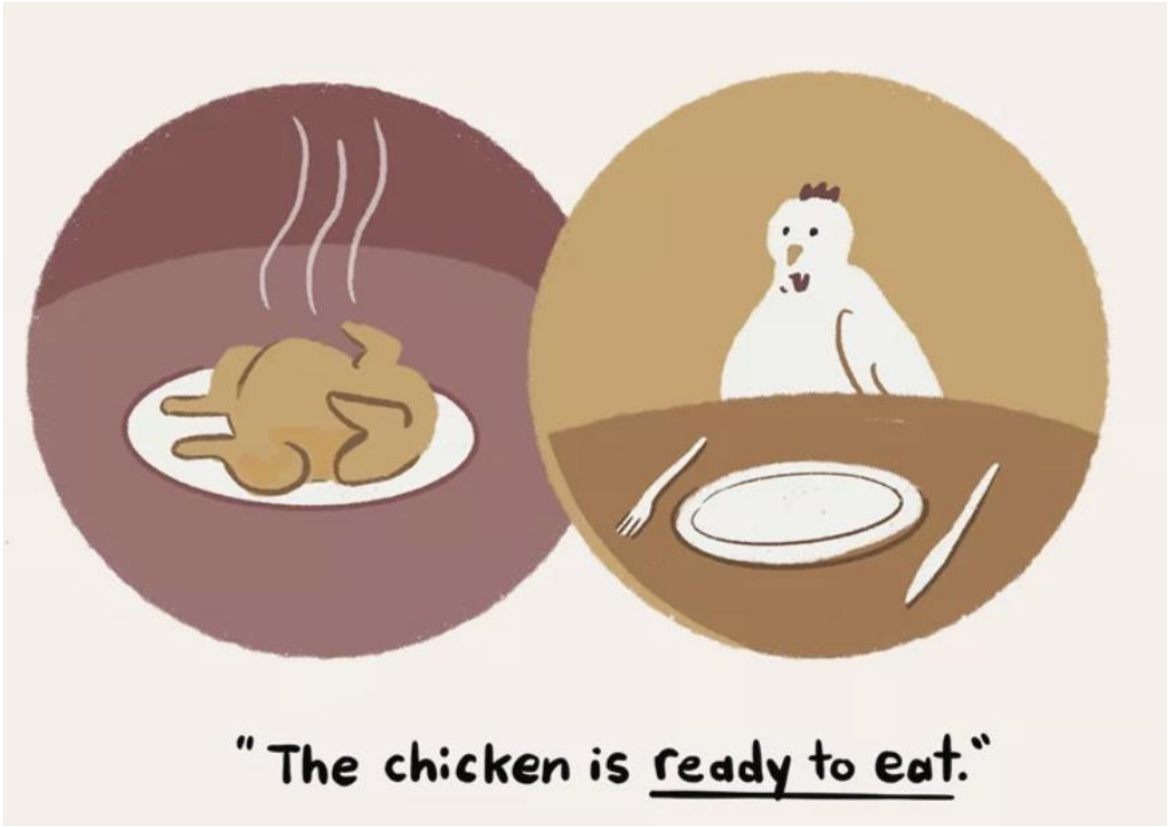
¿Qué es procesamiento de lenguaje natural?





¿Por qué es complejo analizar texto?

AMBIGÜEDAD



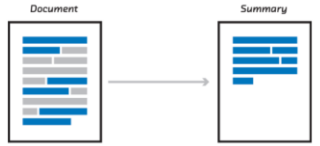
El pollo está listo para comer

Complejidades de analizar texto

JERGA TECNICA

Algunas de las areas/aplicaciones de NLP

Resumen



Pregunta-respuesta

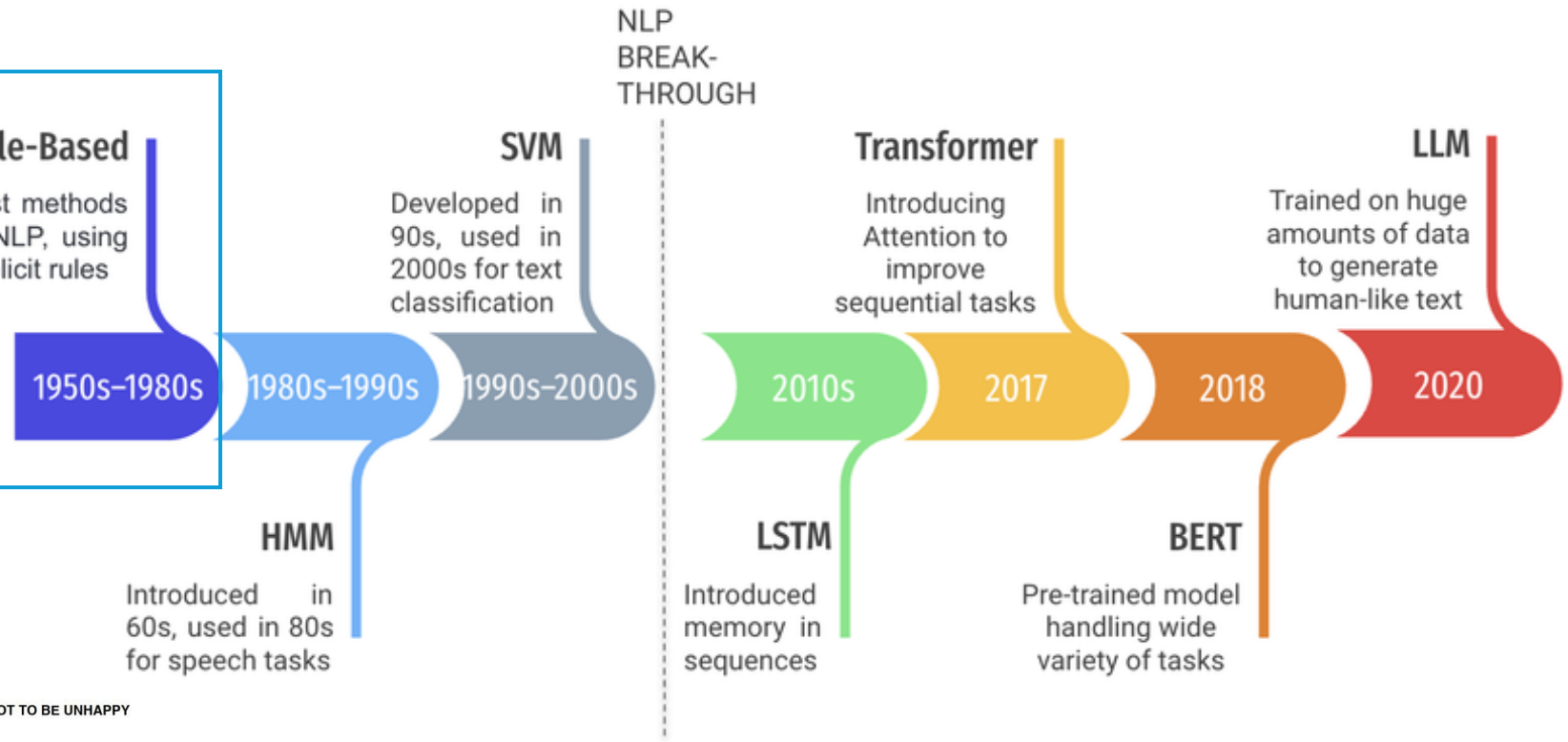
Traducción

Clasificación de sentimientos

Extracción de información

Línea de tiempo de progreso de NLP

Basos en reglas implícitas



ELIZA (1966)

User: I am unhappy.

ELIZA: DO YOU THINK COMING HERE WILL HELP YOU NOT TO BE UNHAPPY

User: I need some help, that much seems certain.

ELIZA: WHAT WOULD IT MEAN TO YOU IF YOU GOT SOME HELP

User: Perhaps I could learn to get along with my mother.

ELIZA: TELL ME MORE ABOUT YOUR FAMILY

User: My mother takes care of me.

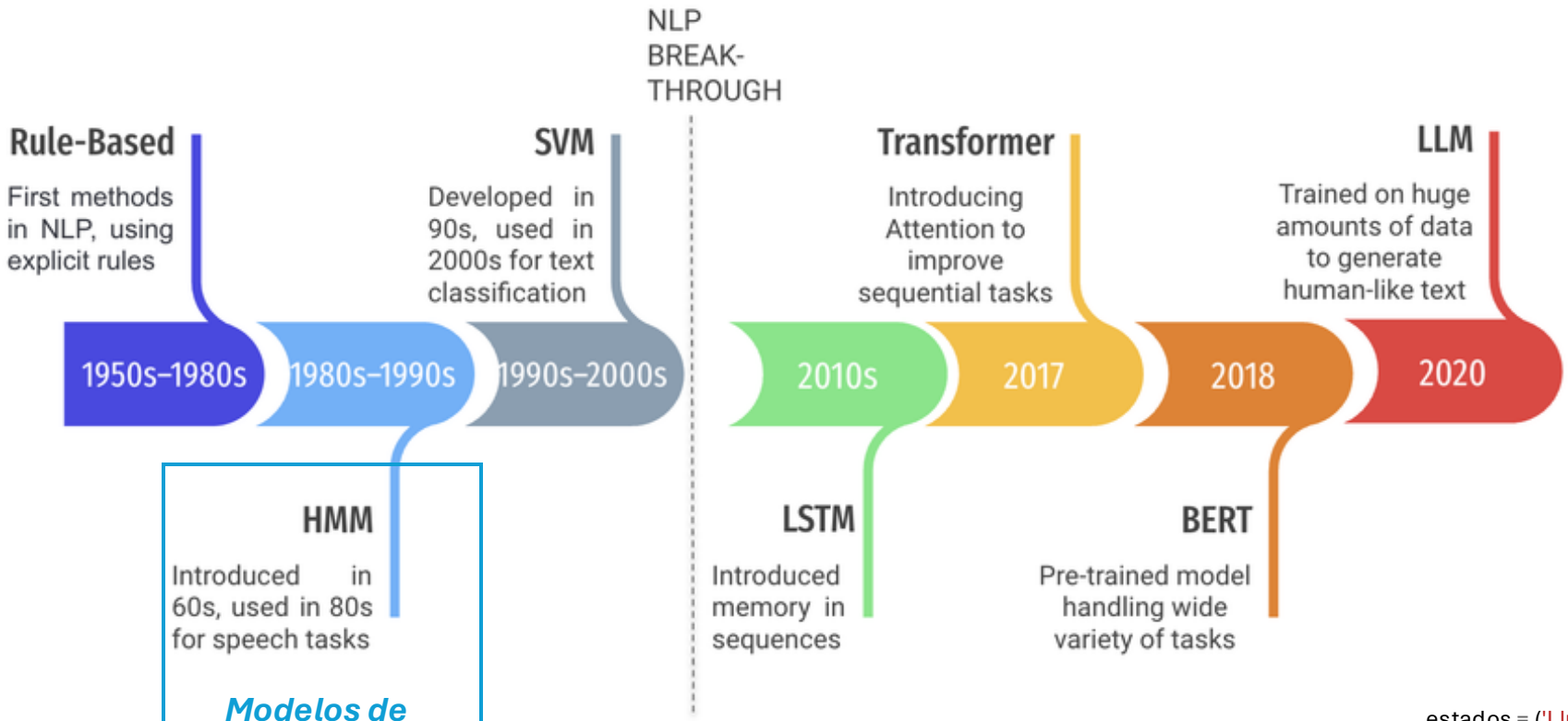
ELIZA: WHO ELSE IN YOU FAMILY TAKES CARE OF YOU

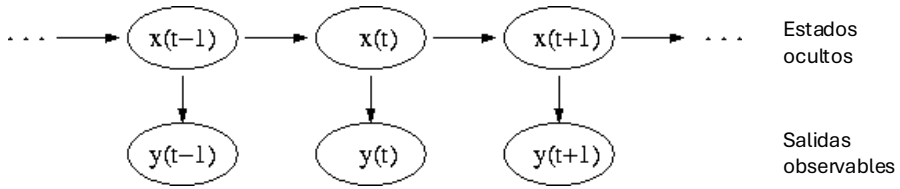
User: My father.

ELIZA: YOUR FATHER

User: You are like my father in some ways.

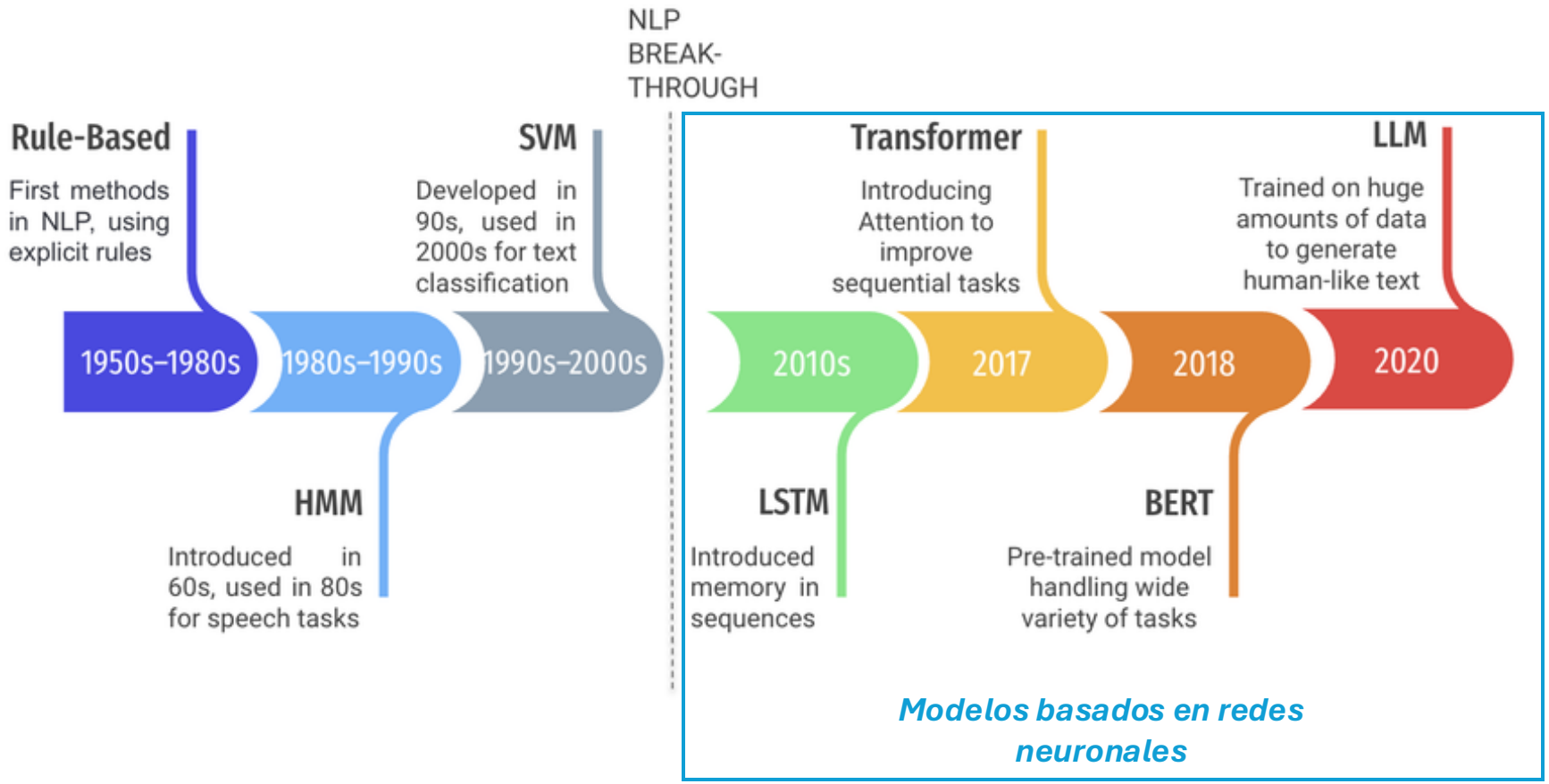
Línea de tiempo de progreso de NLP



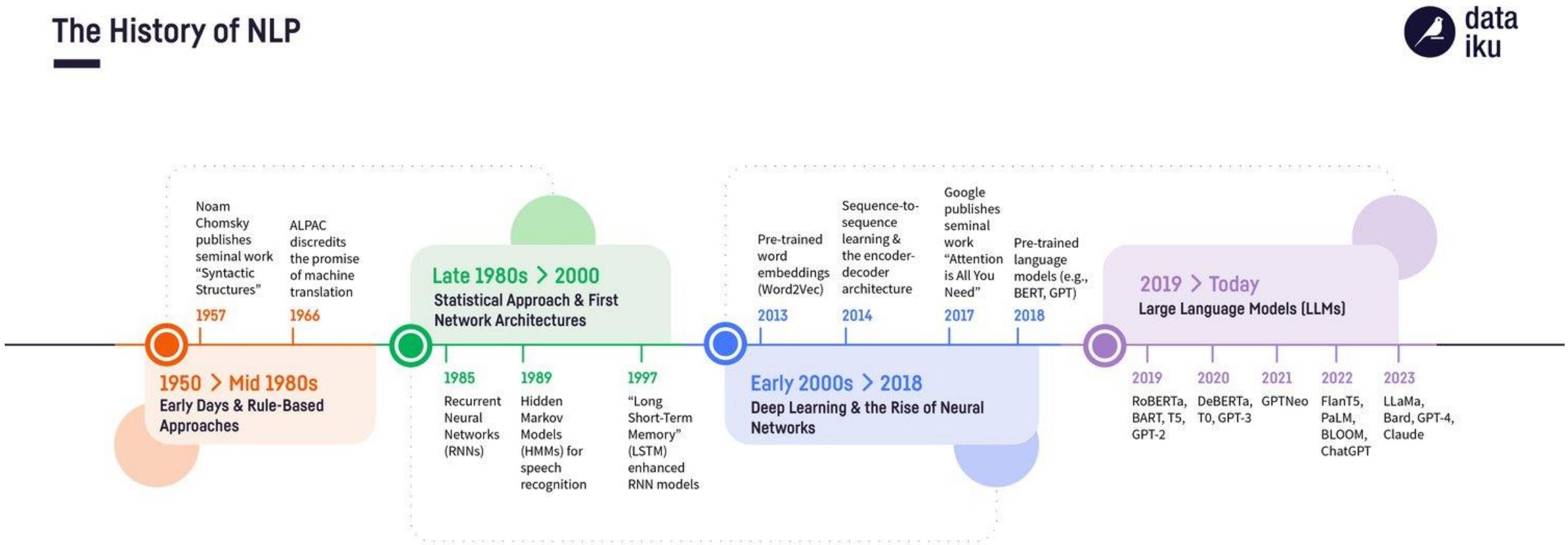



```
estados = ('Lluvioso', 'Soleado')
observaciones = ('caminar', 'comprar', 'limpiar')
probabilidad_inicial = {'Lluvioso': 0.6, 'Soleado': 0.4}
probabilidad_transicion = { 'Lluvioso' : {'Lluvioso': 0.7, 'Soleado': 0.3}, 'Soleado' : {'Lluvioso': 0.4, 'Soleado': 0.6}, }
probabilidad_emision = { 'Lluvioso' : {'caminar': 0.1, 'comprar': 0.4, 'limpiar': 0.5}, 'Soleado' : {'caminar': 0.6, 'comprar': 0.3, 'limpiar': 0.1}, }
```

Línea de tiempo de progreso de NLP



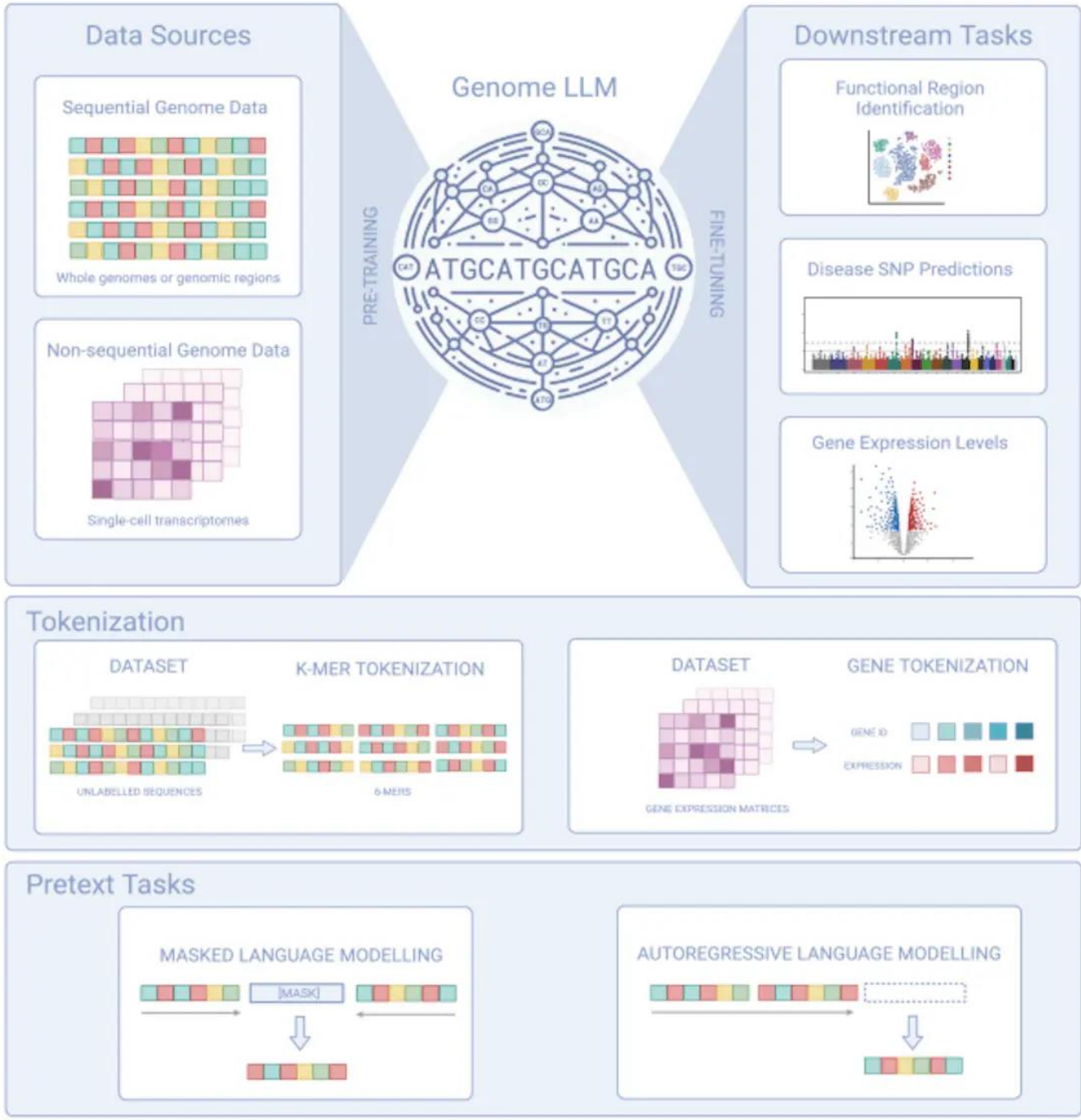
Línea de tiempo más detallada



Casos de uso de NLP

Ejemplos

- Filtros de correos – evitar SPAM, autocorrección y completado de oraciones
- Análisis de comentarios y publicaciones en redes sociales
- Asistentes virtuales (Siri, Alexa, etc)
- Motores de búsqueda (Google, Bing)
- Traducción de idiomas
- Subtítulos
- Resumen de documentos
- Análisis de datos genéticos



Conceptos de lenguaje y técnicas de análisis de texto

Conceptos de lenguaje

Corpus: un conjunto grande y estructurado de textos utilizados para el análisis lingüístico y el aprendizaje automático (Set de datos)

Vocabulario: Conjunto de **palabras únicas** que se reconocen y utilizan en un conjunto de datos específico

Tipos: Se refieren a las palabras **o términos únicos** en un texto determinado. (Sin contar repeticiones)

Tokens: Son **piezas individuales de un texto** que se han segmentado durante la etapa de preprocesamiento (Cuenta repeticiones)

"El perro ladra. El perro corre."

Tipos: 4 - [“el”, “perro”, “ladra”, “.” ,“corre”]

Tokens: 8 - [“el”, “perro”, “ladra”, “.” , “el”, “perro”, “corre”, “.”]

Ejemplo de set de datos para clasificación

Tokens

1	just	just
2	plain	plain
3	boring	boring
4	entirely	entirely
5	predictable	predictable
6	and	and
7	lacks	lacks
8	energy	energy
9	no	no
0	surprises	surprises
11	and	~~and~~
12	very	very
13	few	few
14	laughs	laughs
15	very	~~very~~
16	powerful	powerful
17	the	the
18	most	most
19	fun	fun
20	film	film
21	of	of
22	the	~~the~~
23	summer	summer

23 tokens

Types

1	just
2	plain
3	boring
4	entirely
5	predictable
6	and
7	lacks
8	energy
9	no
0	surprises
11	very
12	few
13	laughs
14	powerful
15	the
16	most
17	fun
18	film
19	of
20	summer

20 types

	Cat	Documents [Jurafsky&Martin]
Training	-	just plain boring
	-	entirely predictable and lacks energy
	-	no surprises and very few laughs
	+	very powerful
	+	the most fun film of the summer
Test	?	predictable with no fun

Vocabulary size:
$|V| = 20$

Partes del discurso

Tag	Description	Example	Tag	Description	Example	Tag	Description	Example
CC	coordinating conjunction	*and, but, or*	PDT	predeterminer	*all, both*	VBP	verb non-3sg present	*eat*
CD	cardinal number	*one, two*	POS	possessive ending	*'s*	VBZ	verb 3sg pres	*eats*
DT	determiner	*a, the*	PRP	personal pronoun	*I, you, he*	WDT	wh-determ.	*which, that*
EX	existential 'there'	*there*	PRP$	possess. pronoun	*your, one's*	WP	wh-pronoun	*what, who*
FW	foreign word	*mea culpa*	RB	adverb	*quickly*	WP$	wh-possess.	*whose*
IN	preposition/ subordin-conj	*of, in, by*	RBR	comparative adverb	*faster*	WRB	wh-adverb	*how, where*
JJ	adjective	*yellow*	RBS	superlatv. adverb	*fastest*	$	dollar sign	*$*
JJR	comparative adj	*bigger*	RP	particle	*up, off*	#	pound sign	*#*
JJS	superlative adj	*wildest*	SYM	symbol	+,%, &	"	left quote	' or "
LS	list item marker	*1, 2, One*	TO	"to"	*to*	"	right quote	' or "
MD	modal	*can, should*	UH	interjection	*ah, oops*	(	left paren	[, (, {, <
NN	sing or mass noun	*llama*	VB	verb base form	*eat*	)	right paren	],), }, >
NNS	noun, plural	*llamas*	VBD	verb past tense	*ate*	,	comma	,
NNP	proper noun, sing.	*IBM*	VBG	verb gerund	*eating*	.	sent-end punc	. ! ?
NNPS	proper noun, plu.	*Carolinas*	VBN	verb past part.	*eaten*	:	sent-mid punc	: ; ... – -

[Jurafsky, 2019]

Palabras tienen una categoría dependiendo de qué papel toman en el discurso

Tipos de pre-procesamiento para modelos de Deep learning

- Limpieza:
  - Estandarización a minúsculas
  - Eliminar puntuación
  - Eliminar caracteres especiales

Depende del caso de uso

- Tokenización: división del texto en palabras
- Eliminación de stop words: palabras sin significado relevante
- Stemming y lemmatization: llevar palabras a su base/raíz
- Palabras fuera del vocabulary (OOV)
- Vectorización: Conversión de texto a vectores numéricos (BoW, embeddings)
- Padding y truncación: asegurar uniformidad en datos de entrada (agrega ceros o corte secuencias)
- Otros generales: aumentación de datos, división en train/validación/test, tratamiento para clases desbalanceadas

Word	Stemming	Lemmatization
information	inform	information
informative	inform	informative
computers	comput	computer
feet	feet	foot

Palabras fuera del vocabulario

- Qué pasa cuando una palabra no fue vista en el documento de entrenamiento?

	Cat	Documents
Training	-	just plain boring
	-	entirely predictable and lacks energy
	-	no surprises and very few laughs
	+	very powerful
	+	the most fun film of the summer
Test	?	predictable with no fun

[Jurafsky&Martin]

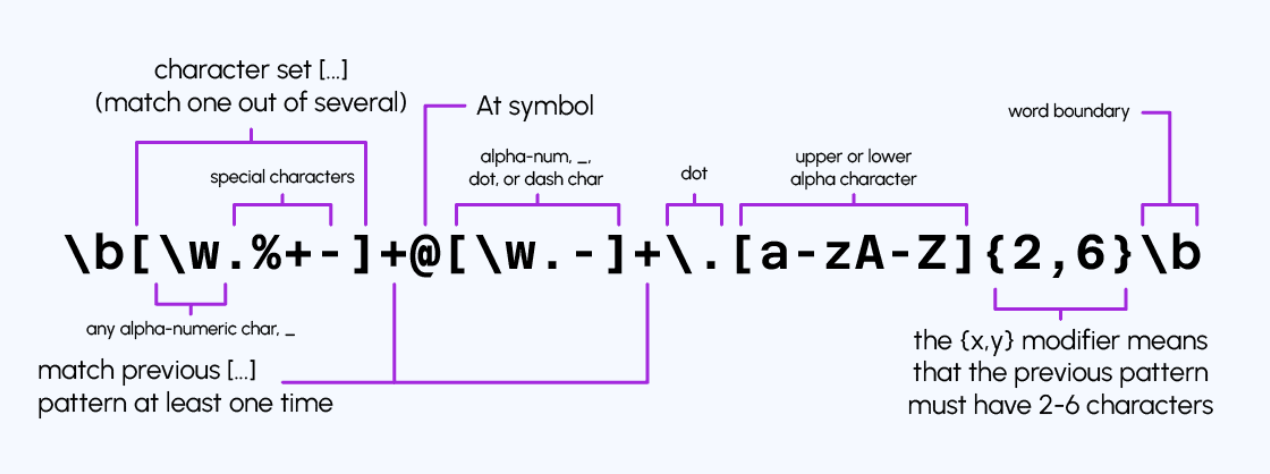
Expresiones regulares (RegEx)

Es una secuencia de caracteres que especifica un patrón de coincidencia en un texto.

Por lo general, los algoritmos de búsqueda de cadenas utilizan dichos patrones para operaciones de "buscar" o "buscar y reemplazar" en cadenas, o para la validación de entradas.

Se usan distintos términos

- **Caracter literal:** Por ejemplo, “r” para buscar la aparición de “r” en el texto
- **Meta-carácter:** Caracter con un significado especial. Pueden hacer cosas como indicar el comienzo de una línea, el final de una línea o hacer coincidir cualquier caracter individual.
- **Clase de caracter:** Conjunto de caracteres, le indica al motor que busque uno de una lista de caracteres. Se indica con y con los caracteres que está buscando en el medio de los **corchetes**.
- **Grupo de captura:** un grupo de captura se indica con **paréntesis redondos** de apertura y cierre. Le permiten agrupar expresiones regulares para aplicar otras características de expresiones regulares como cuantificadores



Ejemplos de operadores

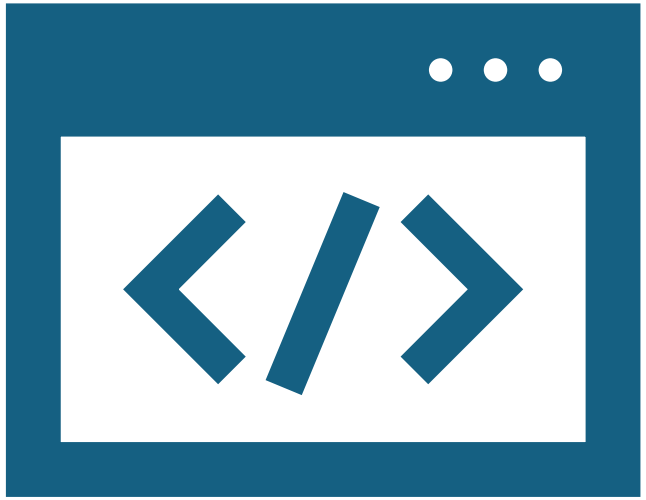
Quantifiers	
`a|b`	Match either “a” or “b”
`?`	Match either “a” or “b”
`+`	One or more
`*`	Zero or more
`*?`	Zero or more, but stop after first match
`{N}`	Exactly N number of times (Where N is number)
`{N, M}`	From N to M number of times (Where N and M are numbers)

Pattern Collections	
`[A-Z]`	Match any uppercase character from “A” to “Z”
`[a-z]`	Match any lowercase character from “a” to “z”
`[0-9]`	Match any number
`[asdf]`	Match any character that’s either “a”, “s”, “d”, or “f”
`[^asdf]`	Match any character that’s not any of the following: “a”, “s”, “d”, or “f”

General Tokens	
`.`	Any character
`\n`	Newline character
`\t`	Tab character
`\s`	Any whitespace character (Including \t, \n, etc)
`\S`	Any non-whitespace character
`\w`	Any word character (Upper/lowercase letters, 0-9, _)
`\W`	Any non-word character
`\b`	Word boundary (Matches between characters)
`\B`	Non-word boundary
`^`	The start of a line
`$`	The end of a line
`\\`	The literal character “\”

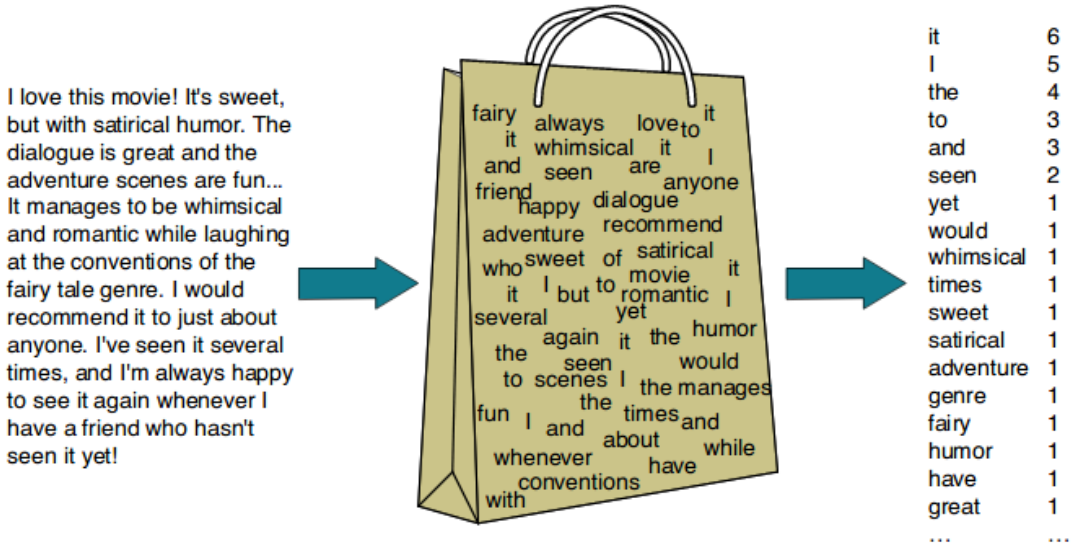
Vectorización

- La vectorización es el proceso de convertir datos de texto en vectores numéricos, lo que permite utilizarlos como entrada para algoritmos de aprendizaje automático y modelos de aprendizaje profundo.
- Dado que las computadoras operan con datos numéricos, transformar el texto en una representación numérica es esencial para cualquier tarea de procesamiento del lenguaje natural.
- Una vectorización eficaz **captura el significado semántico y las relaciones dentro del texto**, lo que ayuda a los modelos a aprender y hacer predicciones con mayor precisión.
- Los dos tipos más populares: **Bag of Words (BoW) y embeddings**



Qué es Bag–of-Words (BoW)

- Uno de los métodos más simples y más utilizados para la vectorización de texto. **En este enfoque, un documento se representa como una "bolsa" de palabras, sin tener en cuenta el orden en que aparecen las palabras.**
- Los pasos principales involucrados en el modelo BoW incluyen:
  - **Creación de vocabulario**
  - **Representación vectorial**
- Si bien BoW es sencillo y eficaz para muchas tareas, **tiene limitaciones**:
  - Ignora el contexto y el orden de las palabras, lo que puede provocar una **pérdida de información**
  - Además, BoW puede generar vectores de **alta dimensión,** especialmente con vocabularios grandes.



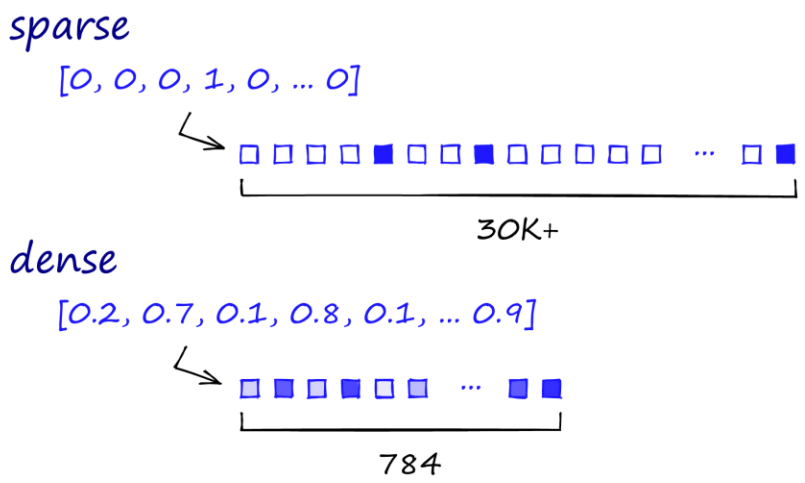
Ejemplo de BoW

	about	bird	heard	is	the	word	you
About the bird, the bird, bird bird bird	1	5	0	0	2	0	0
You heard about the bird	1	1	1	0	1	0	1
The bird is the word	0	1	0	1	2	1	0

Otro tipo de vectorización: Embeddings

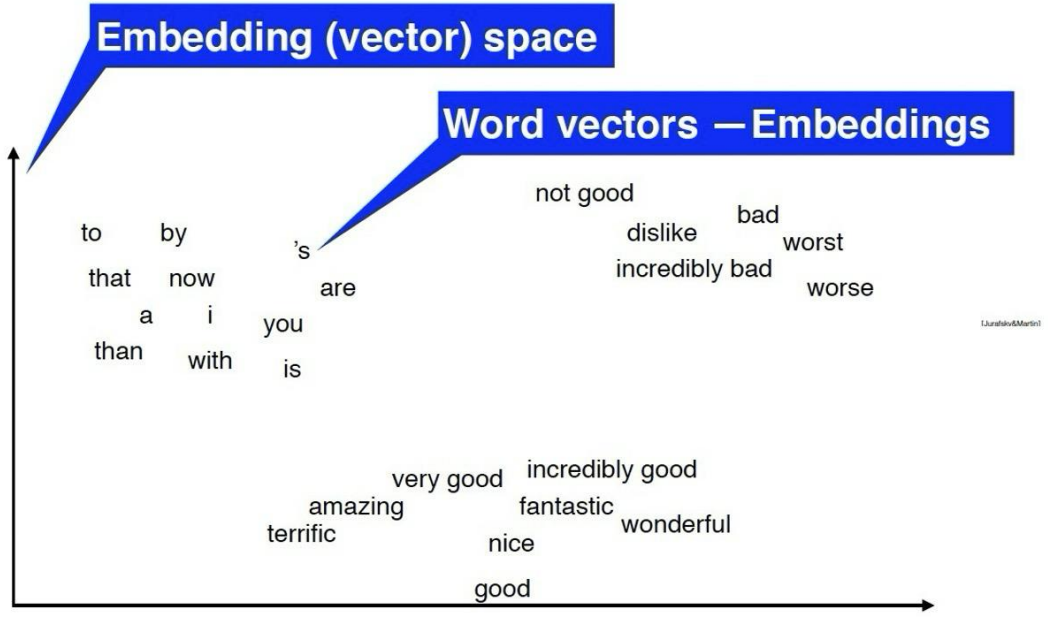
- Existe el concepto de vectores de palabras disperses y densos:

Disperso	Densos
Alta dimensionalidad, la mayoría de sus valores son ceros Ejemplo: BoW	Relativamente menor dimensionalidad, la mayoría de sus valores NO son ceros Ejemplo: embeddings (word2vec, GloVe, etc)

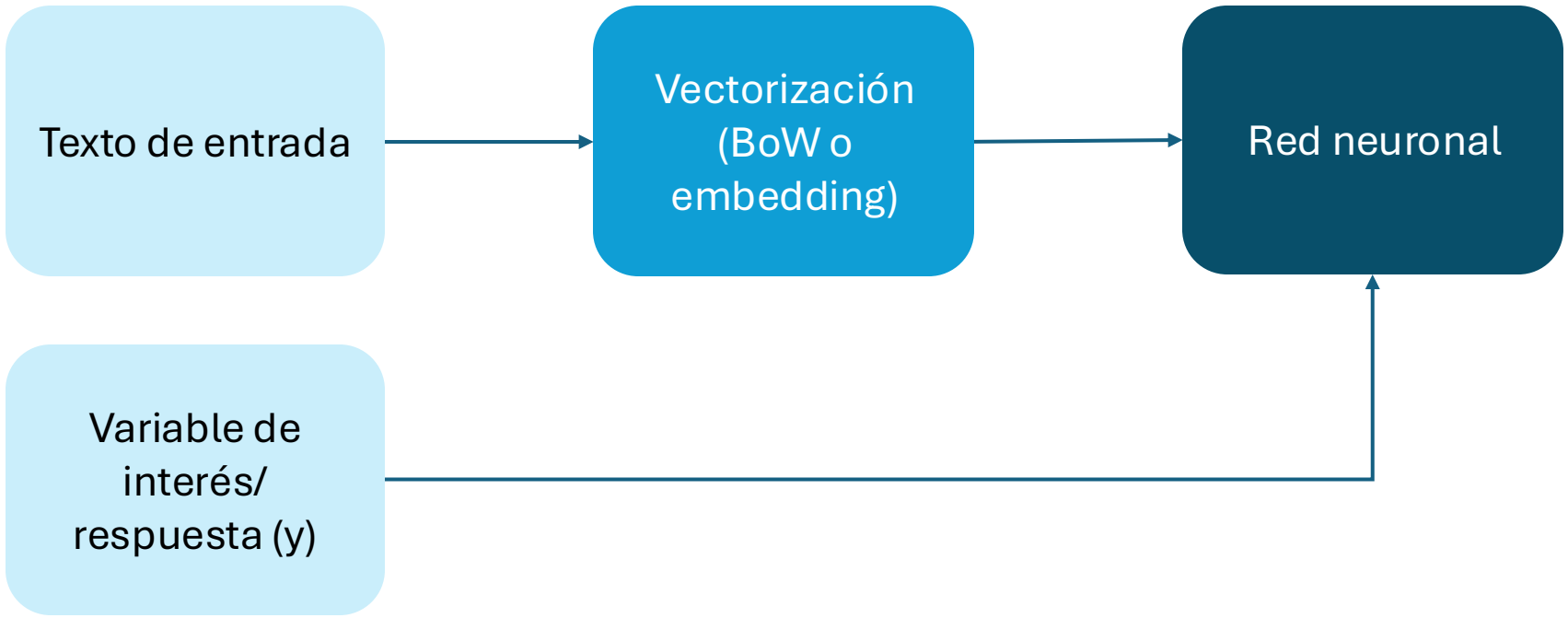


Embeddings

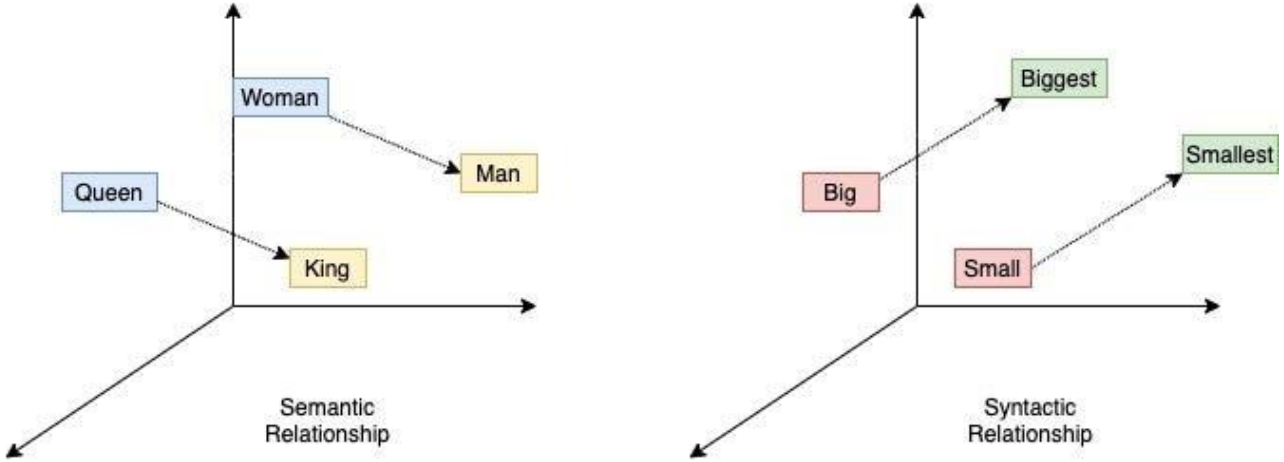
- Para abordar las limitaciones del modelo BoW, los embeddings de palabras proporcionan un enfoque más sofisticado para la vectorización
- Los embeddings representan las palabras como **vectores densos en un espacio vectorial continuo**, donde las **distancias** geométricas entre los vectores **reflejan las relaciones semánticas** entre las palabras
- Hay varios tipos: Word2Vec, GloVe, Fasttext y otros más avanzados generados como parte de LLMs



Cómo funcionan la vectorización en las redes neuronales



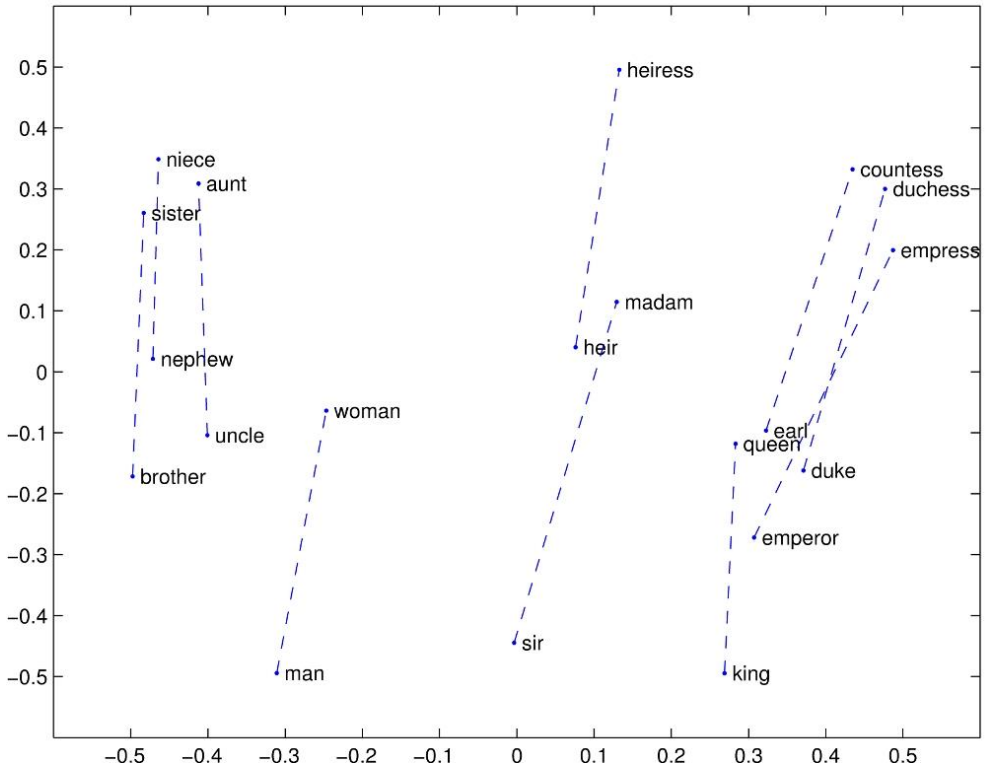
Word2Vec



- Desarrollado por Google en el 2013
- Utiliza una red neuronal pequeña para aprender embeddings de palabras en función de su co-ocurrencia en un corpus grande
- Ofrece dos arquitecturas de entrenamiento:
  - Continuous Bag of Words (CBOW): predice una palabra objetivo dadas sus palabras de contexto
  - Skip-Gram: predice las palabras de contexto dada una palabra objetivo. **Esto es más eficaz para capturar relaciones semánticas**, especialmente para palabras poco comunes
- Una vez generados, los embeddings son independientes del contexto La representación vectorial de una palabra es la misma sin importar dónde aparezca en una oración o documento.
- Capta similitudes sintácticas y semánticas entre las palabras.

GloVe (Global Vectors for Word Representation)

- Desarrollado en Stanford en el 2014
- GloVe es un método basado en recuentos que construye vectores de palabras en función de las estadísticas de co-ocurrencia de palabras en todo el corpus. Construye una matriz de coocurrencia de palabras y la factoriza para producir embeddings.
- Captura tanto el contexto local (palabras vecinas) como las estadísticas globales (coocurrencias de palabras en el corpus).
- Existen embeddings de GloVe entrenadas previamente (por ejemplo, entrenadas en Common Crawl).



Embeddings contextualmente dinámicos

- Se refieren a representaciones de palabras que cambian según el contexto específico en el que aparece la palabra.
- Generan diferentes vectores para la misma palabra según las palabras que la rodean en una oración
- Esta naturaleza dinámica ayuda a capturar los múltiples significados (polisemia) de una palabra y sus relaciones matizadas dentro de diferentes contextos, lo que hace que sean más potentes para tareas complejas de comprensión del lenguaje natural
- Usualmente utilizan arquitecturas Transformers
- Ejemplos: ELMo (Embeddings from Language Models), BERT (Bidirectional Encoder Representations from Transformers), GPT (Generative Pretrained Transformer)

*Voy al **banco** para hacer un depósito.*

*Nos sentamos en el **banco** del parque.*

En uno fijo, ambos bancos tendrían el mismo embedding, en uno contextual son dos distintos

Uso de embeddings pre-entrenados

- Hay embeddings ya entrenados en grandos cuerpos de texto
- Beneficios:
  - Tienden a capturar una amplia gama de relaciones semánticas que serían difíciles de lograr a partir de un conjunto de datos más pequeño y específico del dominio
  - Ahorra tiempo y recursos computacionales en este cálculo
  - Útiles para entendimiento general del lenguaje
- Desventajas:
  - Puede no capturar conceptos específicos del dominio
  - Vienen con un vocabulario fijo

Por qué MLP o CNN no son las ideales para este tipo de datos?

- **MLP:** carecen de memoria y conciencia de secuencias, lo que las hace inadecuadas para tareas de texto donde el orden de las palabras y el contexto son fundamentales.
- **CNN:** pueden capturar patrones locales (como n-gramas), pero tienen dificultades con las dependencias de largo alcance y el contexto global, lo que limita su eficacia para tareas que requieren una comprensión más profunda de la estructura o el significado de las oraciones
- Solución: RNN y Transformers

Más en la próxima clase

Preguntas?