

# Introducción práctica a las tecnologías del lenguaje

Isabel Segura Bedmar  
Máster de Lingüística y Tecnología  
Universidad Complutense de Madrid  
18 de septiembre, 2024

# Un poco de mi...

- 1992-1998: Licenciada en Matemáticas, Ciencias de la Computación.
- 1998-2001: Desarrollo SW para móviles en Telefónica I+D
- 2001-2004: Desarrollo SW banca en Banco Santander.
- 2004- ... : Profesora en el Departamento de Informática de la UC3M, Grupo HuLAT (Human language and Accessibility Technologies).
  - Doctora Europea, Tesis: [Application of information extraction techniques to the pharmacological domain](#). Premio Extraordinario de Doctorado UC3M 2010.
  - Premio Sociedad Español de Procesamiento de Lenguaje Natural. 2011.
  - [DDIExtraction 2011](#), [SemEval 2013 DDIExtraction](#).
  - Proyectos de Investigación: DeepEMR, NLP4Rare, TrendMiner, MultiMedica, etc.
  - Recursos: [Corpus DDI](#), [Ontología DINTO](#), [Corpur RareDis](#),
  - Junta Directiva Sociedad Española de Inteligencia Artificial en Biomedicina ([IABIOMED](#))

# Agenda

- **Introducción**
- Principales aplicaciones PLN.
- Retos en PLN
- Algunos recursos útiles para formación en PLN.

# Procesamiento de Lenguaje Natural (PLN)

- Campo interdisciplinar de la IA y la **lingüística computacional** que se enfoca en la interacción entre las computadoras y el lenguaje humano.
- Su objetivo es desarrollar programas (sw) capaces de comprender, interpretar, generar y responder al lenguaje natural de manera eficaz.

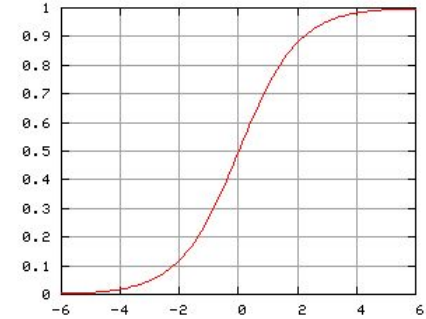
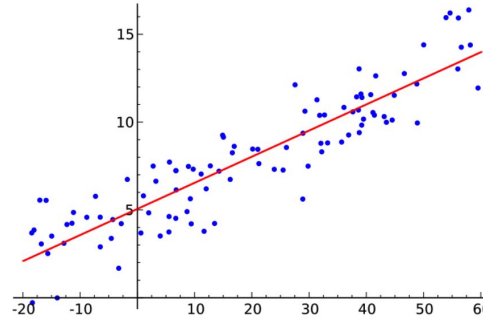
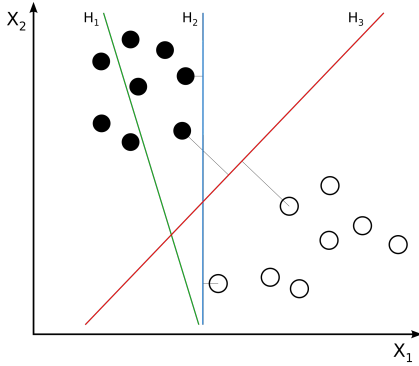
## **Inteligencia Artificial:**

programas con la capacidad de aprender y razonar como seres humanos.

### **Aprendizaje automático:**

algoritmos con la capacidad de aprender sin ser programados explícitamente

# Aprendizaje automático



## Support Vector Machines

Fuente: ZackWeinberg, [Wikimedia](#)

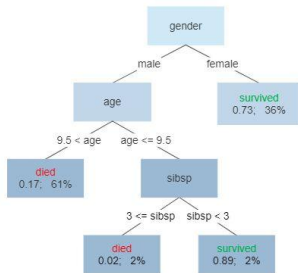
## Regresión Lineal

Fuente: Sewaqu, [Wikimedia](#)

## Logistic regression

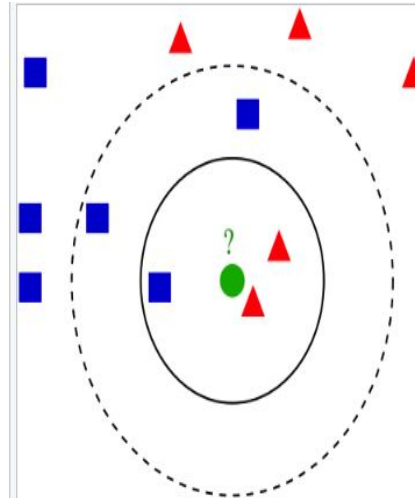
Fuente: [Wikimedia](#)

Survival of passengers on the Titanic



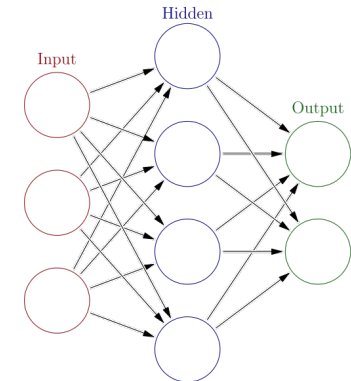
## Árboles de Decision

Fuente: Gilgoldm, [Wikimedia](#)



## k-nearest neighbors

Fuente: Antti Ajanki AnAj, [Wikimedia](#)



## Redes neuronales

Fuente: Glosser.ca, [Wikimedia](#)

## **Inteligencia Artificial:**

programas con la capacidad de aprender y razonar como seres humanos.

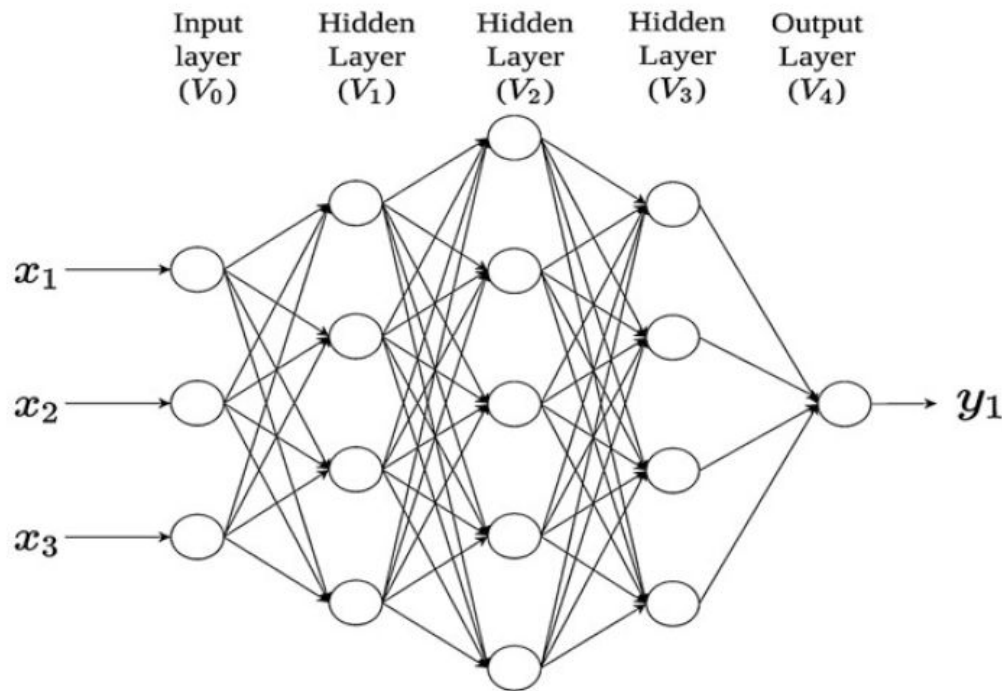
### **Aprendizaje automático:**

algoritmos con la capacidad de aprender sin ser programados explícitamente

### **Aprendizaje profundo:**

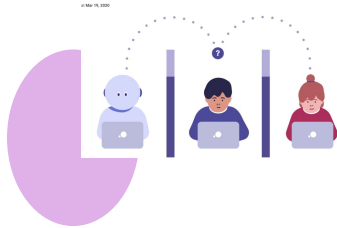
algoritmos de aprendizaje automático basados en redes neuronales con varias capas oculta

# Aprendizaje profundo: redes neuronales profundas



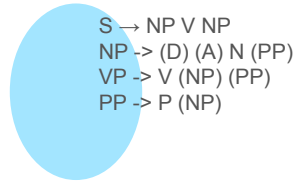


# Evolución PLN



**1950**

Test de Turing  
Primeros experimentos  
en traducción  
automática

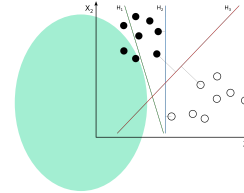


$S \rightarrow NP V NP$   
 $NP \rightarrow (D) (A) N (PP)$   
 $VP \rightarrow V (NP) (PP)$   
 $PP \rightarrow P (NP)$



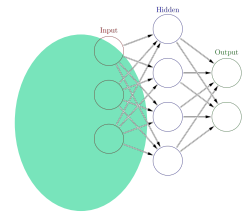
**1960-1980**

Primeros analizadores  
sintácticos basados en  
gramáticas generativas  
(Noam Chomsky)  
  
Primeros sistemas de  
diálogo: ELIZA, SHRDLU



**1990-2000**

Enfoques estadísticos y  
aprendizaje  
automático + corpora



**2010 -**

Enfoques Aprendizaje  
Profundo (word  
embeddings, redes  
recurrentes,  
transformers, etc)

# Agenda

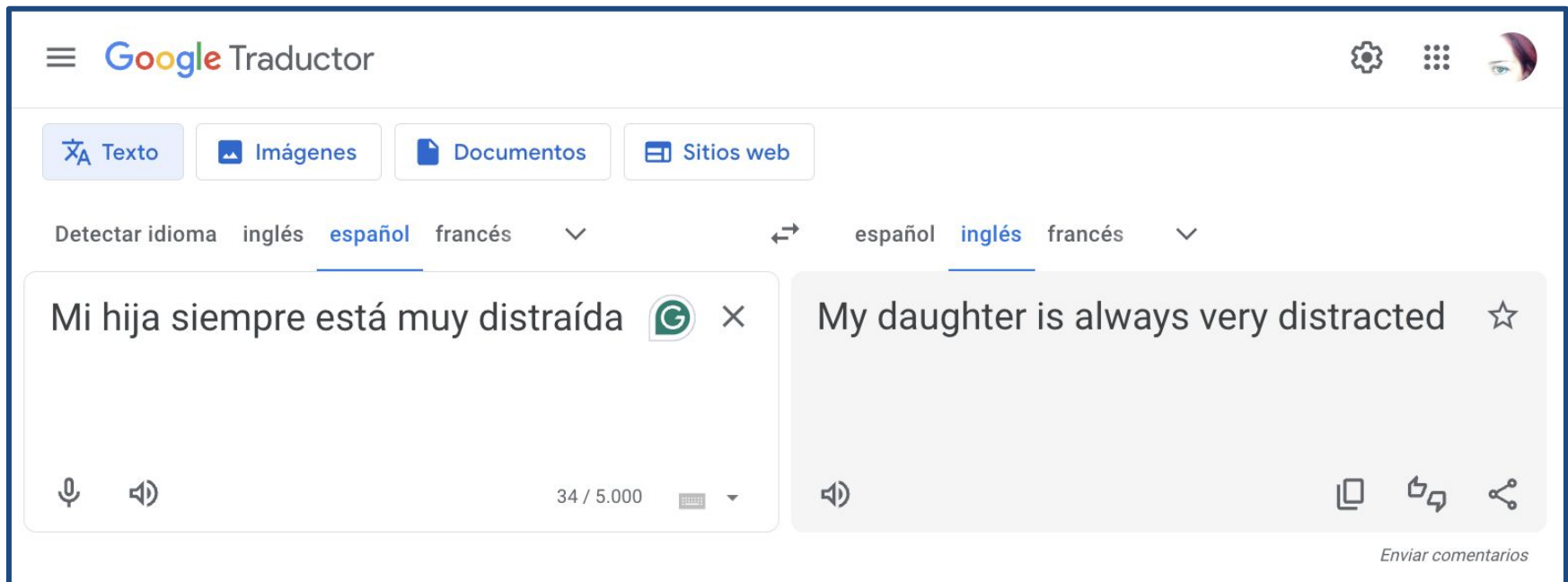
- Introducción
- **Principales aplicaciones PLN.**
- Retos en PLN
- Algunos recursos útiles para formación en PLN.

# Aplicaciones PLN


- **Traducción automática**
- Generación de resúmenes
- Recuperación y Extracción de Información
- Clasificación de textos
- Agentes conversacionales

# Traducción automática

- En inglés **Machine Translation**
- Proceso automático (software) para traducir un texto (o habla) de un idioma a otro.



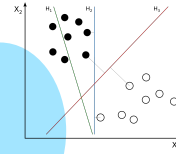
# Traducción automática (enfoques)



```
# texto_es det + nombre + adjetivo  
  
det = extraer_det(texto_es)  
nombre = extraer_nombre(texto_es)  
adjetivo = extraer_adj(texto_es)  
  
det_en = diccionario_es_en[det]  
nombre_en = diccionario_es_en[nombre]  
adjetivo_en = diccionario_es_en[adjetivo]  
texto_en = det_en + " " + adjetivo_en + " " + nombre_en
```

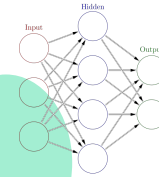
1950-1980

**Rule-based  
machine  
translation**



1980-2010


**Statistical  
machine  
translation**



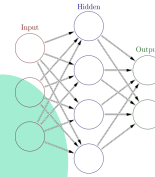
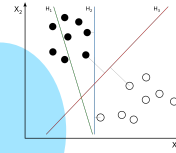
2010-...

**Neuronal  
machine  
translation**

# Traducción automática (enfoques)



```
# texto_es det + nombre + adjetivo  
  
det = extraer_det(texto_es)  
nombre = extraer_nombre(texto_es)  
adjetivo = extraer_adj(texto_es)  
  
det_en = diccionario_es_en[det]  
nombre_en = diccionario_es_en[nombre]  
adjetivo_en = diccionario_es_en[adjetivo]  
texto_en = det_en + " " + adjetivo_en + " " + nombre_en
```



1950-1980

**Rule-based  
machine  
translation**

1980-2010

**Statistical  
machine  
translation**

2010-...

**Neuronal  
machine  
translation**

# **Traducción automática basada en reglas (Rule-based machine translation)**

- Basados en reglas gramaticales (morfología, sintaxis y semántica) y en diccionarios.
- En la actualidad, se siguen utilizando porque ayudan a preservar la consistencia en ciertos contextos (por ejemplo, documentos técnicos y legales).

# Traducción automática basada en reglas (Rule-based machine translation)

## Ejemplos de reglas gramaticales:

Español:

*“Las casas bonitas”*

Traducción literal incorrecta:

*“The **houses** beautiful”*

Traducción correcta: “

*The beautiful houses”*



# Traducción automática basada en reglas (Rule-based machine translation)

## Ejemplos de reglas gramaticales:

Español:

*“Me gusta el chocolate”*

Traducción literal incorrecta:

*“I like the chocolate”*

Traducción correcta:

*“I like chocolate”*

# Traducción automática basada en reglas (Rule-based machine translation)

## Ejemplos de reglas gramaticales:

Español:

*“Había terminado cuando tú llegastes”*

Traducción literal incorrecta:

*“**Had finished** when you arrived”*

Traducción correcta:

*“**I had finished** when you arrived”*

# Traducción automática basada en reglas (Rule-based machine translation)

## Ejemplos de reglas gramaticales:

Español:

*“Pensar en algo”*

Traducción literal incorrecta:

*“Think **in** something”*

Traducción correcta:

*“Think **about** something”*

# Traducción automática basada en reglas (Rule-based machine translation)

## Ejemplos de reglas gramaticales:

Español:

*“Está lloviendo a cantaros”*

Traducción literal incorrecta:

*“It’s raining **jugs**”*

Traducción correcta:




*“It’s raining **cats and dogs**”*

# Traducción automática basada en reglas (Rule-based machine translation)

- Algunos ejemplos de diccionarios:
  - **Diccionarios bilingües generales:** Oxford dictionary, [Cambridge Dictionary](#), [WordReference](#), [Glosbe](#), etc
  - **Diccionarios especializados:** [IATE](#) (Inter-Active Terminology for Europe), [EuroVoc](#) (tesauro multilingüe con términos de la UE en los dominios de derecho y administración), etc.
  - **Diccionarios de expresiones idiomáticas:** Cambridge Idioms dictionary, HarperCollins Idioms Dictionary, [reverso](#) dictionary, etc.





# Traducción automática basada en reglas (Rule-based machine translation)

## Ventajas:


-  **Precisión alta:** manejo y traducción correcta de estructuras sintácticas complejas.
-  Enfoque **transparente y comprensible**; fácil de mantener y ajustar.
-  Aplicable a **idiomas con escasos recursos** y datasets para entrenar modelos estadísticos y redes neuronales.

# Traducción automática basada en reglas (Rule-based machine translation)

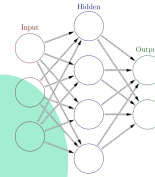
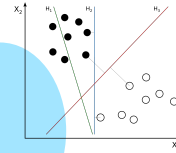
## Desventajas:

-  **Enfoque costoso y lento** para desarrollar y mantener las reglas y diccionarios
-  **Escalabilidad limitada** a otros idiomas: cada nuevo par de idiomas requiere sus propias reglas y diccionarios.
-  **Rigidez para adaptarse** a la naturaleza dinámica del lenguaje (nuevo vocabulario, etc).
-  **Dificultad para manejar excepciones y contextos específicos:** no se ajusta bien a textos creativos, etc

# Traducción automática (enfoques)



```
# texto_es det + nombre + adjetivo  
  
det = extraer_det(texto_es)  
nombre = extraer_nombre(texto_es)  
adjetivo = extraer_adj(texto_es)  
  
det_en = diccionario_es_en[det]  
nombre_en = diccionario_es_en[nombre]  
adjetivo_en = diccionario_es_en[adjetivo]  
texto_en = det_en + " " + adjetivo_en + " " + nombre_en
```



1950-1980

Rule-based  
machine  
translation

1980-2010

Statistical  
machine  
translation

2010-...

Neuronal  
machine  
translation



# **Traducción automática estadística (Statistical machine translation).**

- **Modelos estadísticos** contruidos a partir de **corpora paralelos** (grandes colecciones de textos paralelos)

# Traducción automática estadística (Statistical machine translation).

## ¿Qué es un corpus paralelo?

Ejemplo de alineamiento de oraciones (Europarl)

Inglés	Español
"The European Parliament is committed to ensuring that all citizens are represented."	"El Parlamento Europeo está comprometido a garantizar que todos los ciudadanos estén representados."
"We need to address the issue of climate change with urgent measures."	"Necesitamos abordar el problema del cambio climático con medidas urgentes."
"The new regulations will come into effect starting next year."	"Las nuevas regulaciones entrarán en vigor a partir del próximo año."
"Our goal is to achieve sustainable development across the continent."	"Nuestro objetivo es lograr el desarrollo sostenible en todo el continente."
"It is essential to enhance cooperation between member states."	"Es esencial fortalecer la cooperación entre los Estados miembros."

# Traducción automática estadística (Statistical machine translation).

Inglés	Español
"The European Parliament"	"El Parlamento Europeo"
"is committed to ensuring"	"está comprometido a garantizar"
"that all citizens are represented."	"que todos los ciudadanos estén representados."

Alineamiento de frases de la primera oración:

*"The European Parliament is committed to ensuring that all citizens are represented."*

# Traducción automática estadística (Statistical machine translation).

- Ejemplos de corpora paralelos:
  - [Europarl](#): transcripciones de debates en el Parlamento Europeo (21 idiomas).
  - [United Nations Parallel Corpus](#): documentos oficiales de la ONU, en Árabe, Chino, Inglés, Francés y Español.
  - [Opensubtitles](#): subtítulos de películas y series; excelente recurso para la traducción de lenguaje coloquial y expresiones idiomáticas.
  - [PaEns](#): inglés - español, incluye obras de ficción, novelas y cuentos, y no ficción (psicología, ensayos y textos de divulgación científica).

# Traducción automática estadística (Statistical machine translation).





- Gracias al corpus paralelo, es posible construir un **modelo estadístico** que calcule la probabilidad de que una frase se traduzca a otra en otro idioma.
- Por ejemplo, las posibles traducciones para “**en su casa**” podrían ser:
  - *at their house*
  - *at his house*
  - *at her house*
  - *in their house*
  - *in his house*
  - *in her house*

# Traducción automática estadística (Statistical machine translation).


- Se divide el texto en frases o n-gramas: *“Ana compró un libro interesante en la librería”* -> [*“Ana compró”*], [*“un libro interesante”*], [*“en la librería”*]
- El modelo estadístico (generado a partir del corpus paralelo) nos proporciona las traducciones más probables para cada frase
  - *“Ella compró”* ->
    - *“She bought”,*
  - *“un libro interesante”* ->
    - *“an interesting book”*
  - *“en la librería”* ->
    - *“in the bookstore”,*
    - *“at the bookstore”*
- Se forma el texto final:
  - *“She bought an interesting book at the bookstore.”*

# Traducción automática estadística (Statistical machine translation).

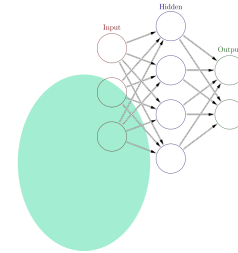
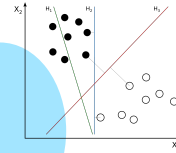
## Pros y contras:

-  Fácil adaptación a diferentes contextos y estilos. Traducciones más para manejar variaciones y adaptación a diferentes contextos.
-  Requiere grandes colecciones de textos paralelos.
-  Enfoque menos transparente que el basado en reglas.
-  Puede generar traducciones incorrectas o no naturales.

# Traducción automática (enfoques)



```
# texto_es det + nombre + adjetivo  
  
det = extraer_det(texto_es)  
nombre = extraer_nombre(texto_es)  
adjetivo = extraer_adj(texto_es)  
  
det_en = diccionario_es_en[det]  
nombre_en = diccionario_es_en[nombre]  
adjetivo_en = diccionario_es_en[adjetivo]  
texto_en = det_en + " " + adjetivo_en + " " + nombre_en
```



1950-1980

Rule-based  
machine  
translation

1980-2010

Statistical  
machine  
translation

2010-...

Neuronal  
machine  
translation

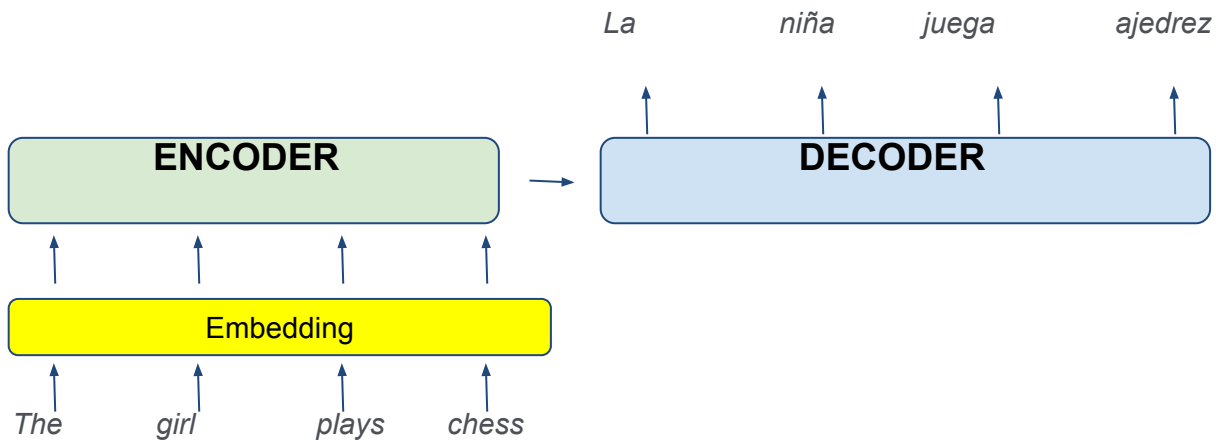


# Traducción automática basada en redes neuronales (Neural machine translation)

- Utiliza redes neuronales profundas para traducir de un idioma a otro.
- Requiere grandes colecciones de textos bilingües (texto original, y su traducción) para entrenar la red.
- Arquitectura de la red: **Seq2Seq** (secuencia a secuencia)

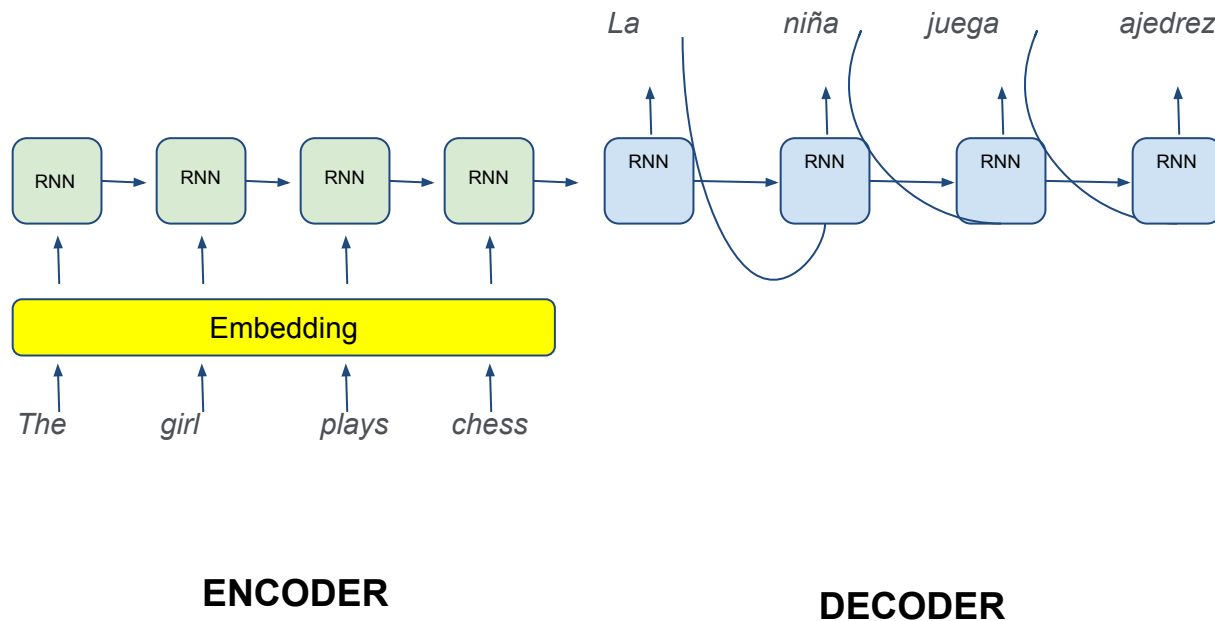
# Traducción automática basada en redes neuronales (Neural machine translation)

- ¿Qué es Seq2Seq? Recibe como entrada una secuencia, y genera como salida otra secuencia.



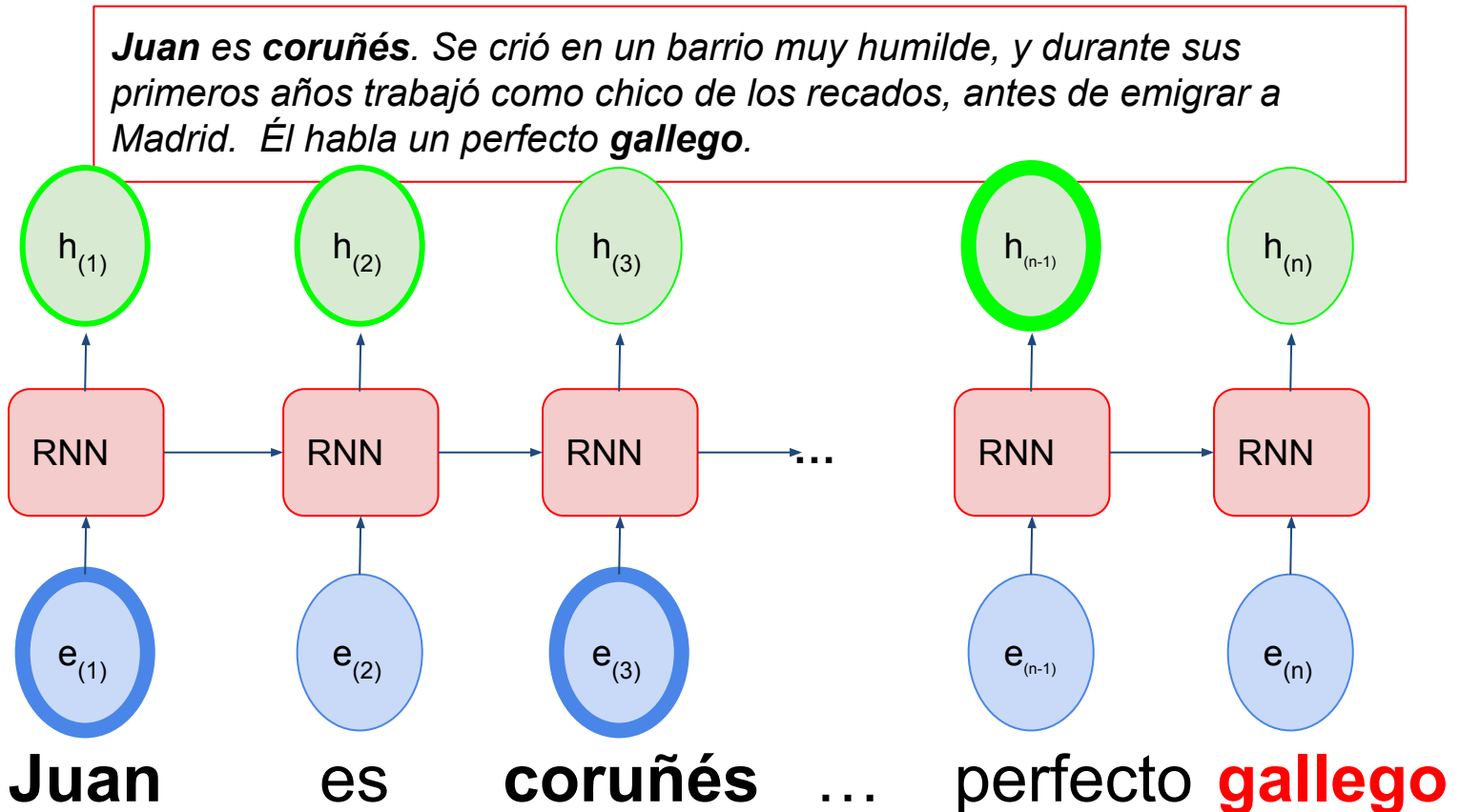
# Traducción automática basada en redes neuronales (Neural machine translation)

- Las **primeras** arquitecturas **Seq2Seq** estaban basadas en **redes recurrentes**



# Traducción automática basada en redes neuronales (Neural machine translation)

- **Limitaciones de las redes recurrentes:** 1) pérdidas de información en oraciones largas; 2) no permiten paralelización.

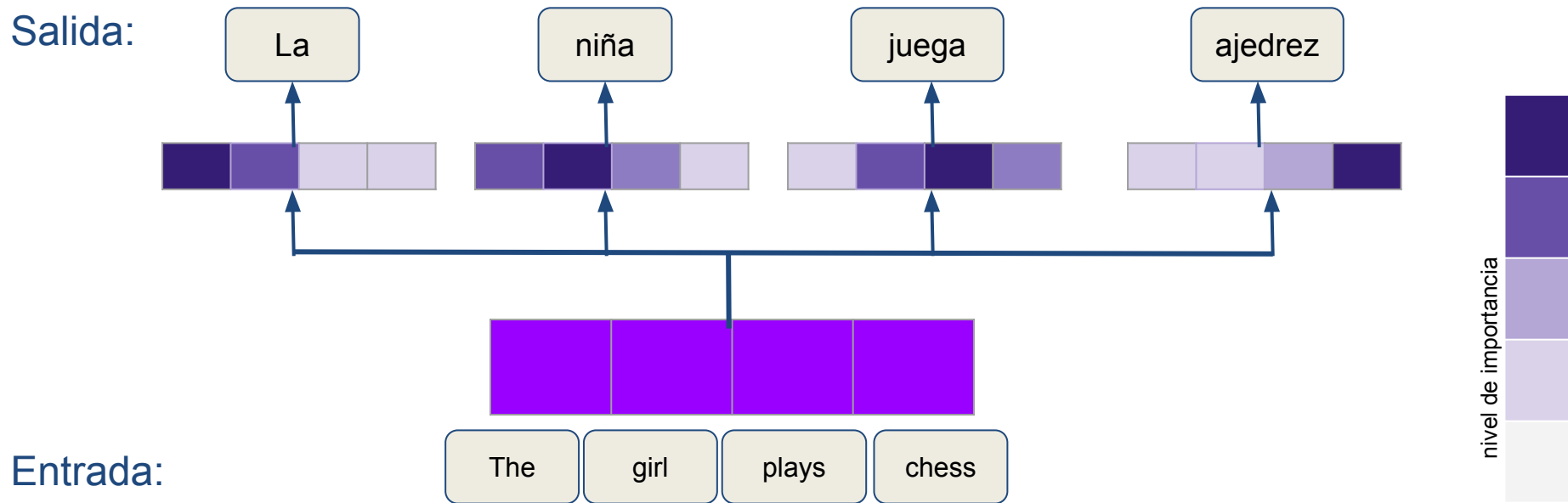


# Traducción automática basada en redes neuronales (Neural machine translation)

- En el artículo “**Attention is all you need**” (Vaswani et al, 2017) se presenta el mecanismo de atención.
- Supera las limitaciones de las redes recurrentes, y obtiene mejoras significativas en precisión y eficiencia en muchas tareas de PLN.

# Traducción automática basada en redes neuronales (Neural machine translation)

- ¿Qué es el mecanismo de atención?



# Traducción automática basada en redes neuronales (Neural machine translation)

- Mecanismos de atención => modelos **transformers**
- Las características principales de estos modelos son: **self-attention** y **multi-head** (permite paralelización)
- Muy eficientes porque únicamente utilizan funciones de activación (softmax) y sumas ponderadas.

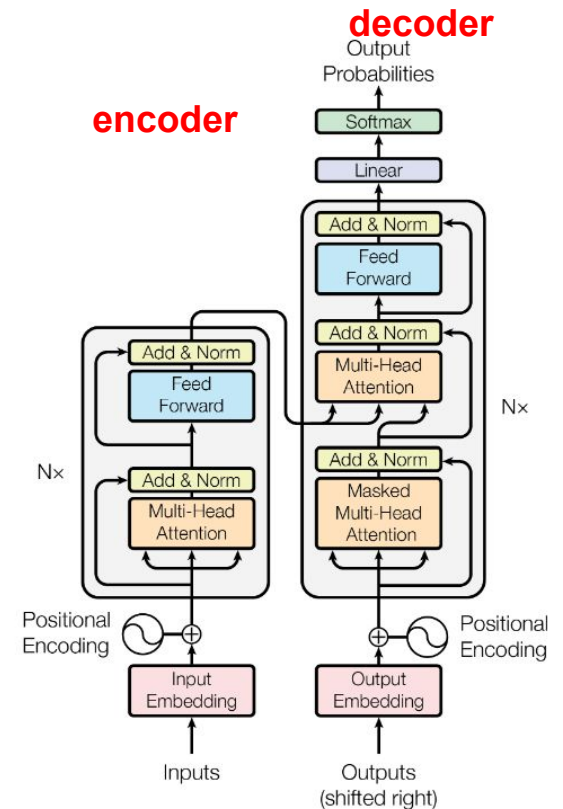




Figura tomada de Vaswani, A. et al. (2017). **Attention is all you need**. *Advances in neural information processing systems*, 30.

# Traducción automática neuronal

## Pros y contras:

 Mejor precisión y fluidez

 Requiere grandes colecciones de datos y recursos computacionales; dependen de la calidad de los datos.

 Caja negra (escasa explicabilidad de los resultados)



# APIs para traducción automática

- [Ejemplo](#)

# Traducción automática neuronal (Neural machine translation)

- Ejemplos de servicios de traducción basados en redes neuronales:
  - Google Translate
  - Microsoft Translate
  - DeepL Translator
  - Amazon Translate
  - Systran (reglas + modelos neuronales)
  - IBM Watson Language Translator
  - Baidu Translate

# Aplicaciones PLN

- Traducción automática
- **Generación de resúmenes**
- Recuperación y Extracción de Información
- Clasificación de textos
- Agentes conversacionales

# Generación de resúmenes (text summarization)

- Objetivo: Producir un resumen que sintetice la información más relevante de un texto o conjunto de textos de entrada.
- Ejemplos de aplicaciones:
  - Resumen de la historia clínica de un paciente.
  - Simplificación de un texto en una versión más sencilla.
  - Resumen de emails, artículos, etc...

# Generación de resúmenes (text summarization)

- **Tipos:**
  - **Extractiva:** el resumen es creado a partir de palabras y oraciones que aparecen en el texto de entrada.
  - **Abstractiva:** el resumen generado es un texto nuevo que sintetiza la idea del texto original.

# Generación de resúmenes (text summarization)

- Ejemplo Extractiva vs Abstractiva:
  - **Texto original:** *Yolanda Díaz en un acto de Sumar ha afirmado que existe una elite mundial preparando planes alternativos para poder huir del mundo en el caso de que nos vayamos al “carajo”.*

# Generación de resúmenes (text summarization)

- Ejemplo Extractiva vs Abstractiva:
  - **Texto original:** *Yolanda Díaz en un acto de Sumar ha afirmado que existe una elite mundial preparando planes alternativos para poder huir del mundo en el caso de que nos vayamos al “carajo”.*
  - **Resumen extractivo:** *Yolanda Díaz ha afirmado que existe una elite mundial preparando planes para poder huir del mundo.*

# Generación de resúmenes (text summarization)

- Ejemplo Extractiva vs Abstractiva:
  - **Texto original:** *Yolanda Díaz en un acto de Sumar ha afirmado que existe una elite mundial preparando planes alternativos para poder huir del mundo en el caso de que que nos vayamos al “carajo”.*
  - **Resumen extractivo:** *Yolanda Díaz ha afirmado que existe una elite mundial preparando planes para poder huir del mundo.*
  - **Resumen abstracto:** *Según Yolanda Díaz, existe una elite mundial que planea huir si el mundo se va al carajo.*



# Generación de resúmenes (text summarization)

- Principales enfoques extractivos:
  - **reglas** para seleccionar las oraciones más importantes:
    - *incluir la primera oración,*
    - *excluir oraciones demasiado cortas,*
    - *identificar el o temas del texto, y seleccionar oraciones que los contengan.*
    - *etc*
  - **estadísticos**: basados en modelos BoW o tf-idf; se seleccionan las oraciones con las puntuaciones más altas. La puntuación de una oración se obtiene sumando las frecuencias de sus palabras.
  - **grafos** (TextRank): las oraciones son representadas como nodos; las aristas del grafo son las relaciones de similitud entre oraciones. Se seleccionan las oraciones más “centrales”.
  - **algoritmos de clasificación** (ej. **SVM**) entrenados con **corpus paralelos**.

# Generación de resúmenes (text summarization)

- Enfoques utilizados
  - **Extractiva:**
    - **grafos:** oraciones (nodos) relaciones basadas en la similitud entre oraciones (TextRank),. El algoritmo selecciona las oraciones más “centrales”.
      - Representa la estructura global del texto y se enfoca en cómo las oraciones están relacionadas semánticamente.
      - Son sencillos de implementar y aplicables a cualquier tipo de texto.
      - No requiere grandes colecciones de textos
      - Las oraciones más “centrales” suelen coincidir con las más representativas.
      - Pueden generar resúmenes redundantes (oraciones muy similares pueden ser seleccionadas) y pérdida de información importante.

# Generación de resúmenes (text summarization)

- Enfoques utilizados
  - **Extractiva:**
    - **supervisados:** algoritmos (ej SVM) son entrenados con un corpus paralelo (texto original + resumen) para aprender a identificar si una oración del texto original es relevante o no para el resumen.
      - Buenos resultados si se cuenta con un buen corpus (cantidad y calidad).
      - Elaboración de estos datasets es muy costosa.
      - Difícil de aplicar a otros dominios

# Generación de resúmenes (text summarization)

- Principales enfoques abstractivos:
  - Suelen estar basados en arquitecturas **Seq2Seq** con **redes neuronales profundas**:
    - redes recurrentes
    - **transformers**:
      - muy efectivos (enfoque extractivo como abstractivo), generando resúmenes más coherentes y precisos.

# Algunas APIs para generación de resúmenes

- *Ejemplo*

# Aplicaciones PLN

- Traducción automática
- Generación de resúmenes
- **Recuperación y Extracción de Información**
- Clasificación de textos
- Agentes conversacionales

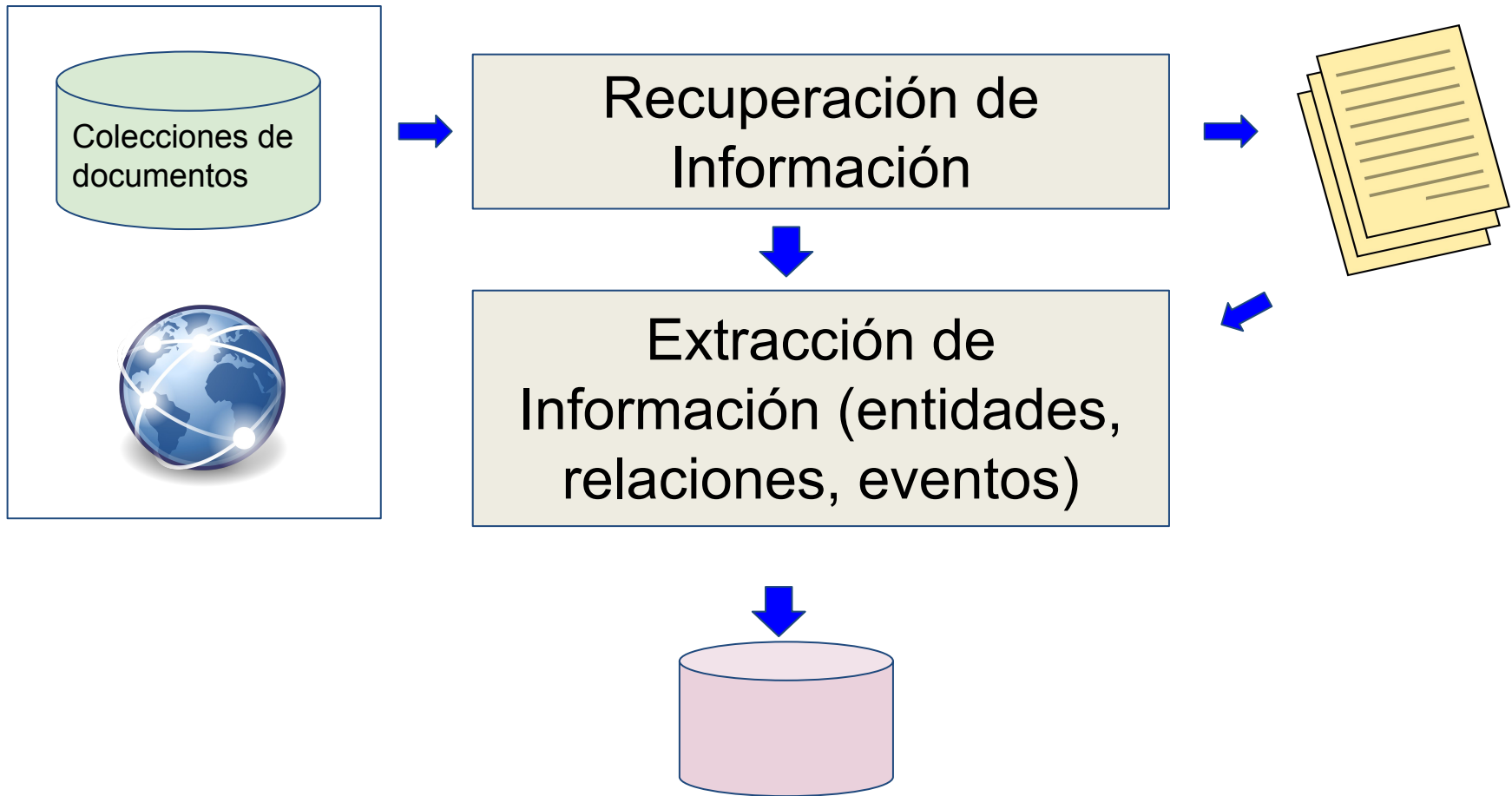
# Recuperación de Información

Dada una consulta o necesidad de información, el objetivo es encontrar los documentos más relevantes para dicha consulta



# Extracción de Información

Información no estructurado (textos) -> Estructurado (bases de datos, ontologías, etc)



Bases de datos, ontologías, etc



# Extracción de Información

*Isabel Segura-Bedmar: HULAT at SemEval-2023 Task 10: Data Augmentation for Pre-trained Transformers Applied to the Detection of Sexism in Social Media. SemEval@ACL 2023: 184-192. 13-14 July 2023 Toronto.*



**Autor:** Isabel Segura-Bedmar.

**Título:** HULAT at SemEval-2023 Task 10: Data Augmentation for Pre-trained Transformers Applied to the Detection of Sexism in Social Media.

**Conferencia:** SemEval@ACL 2023

**Páginas:** 184-192

**Fechas:** 13-14 July 2023

**Lugar:** Toronto

# Extracción de Información

1 El asma afecta las vías respiratorias.

```
graph LR; C1[Concept: asma] -- Subject --> A1[Action: afecta]; A1 -- Target --> C2[Concept: vías respiratorias]; C1 -- is-a --> C2;
```

2 Los pulmones se hinchan.

```
graph LR; C1[Concept: pulmones] -- Subject --> A1[Action: hinchan];
```

3 El asma es una enfermedad.

```
graph LR; C1[Concept: asma] -- is-a --> C2[Concept: enfermedad];
```

4 Un ataque de asma se produce cuando los síntomas empeoran.

```
graph LR; C1[Concept: ataque de asma] -- Subject --> A1[Action: se produce]; A1 -- Target --> C2[Concept: síntomas]; C1 -- is-a --> C2;
```

5 Los síntomas y el tratamiento dependen del tipo de cáncer y de lo avanzada que esté la enfermedad.

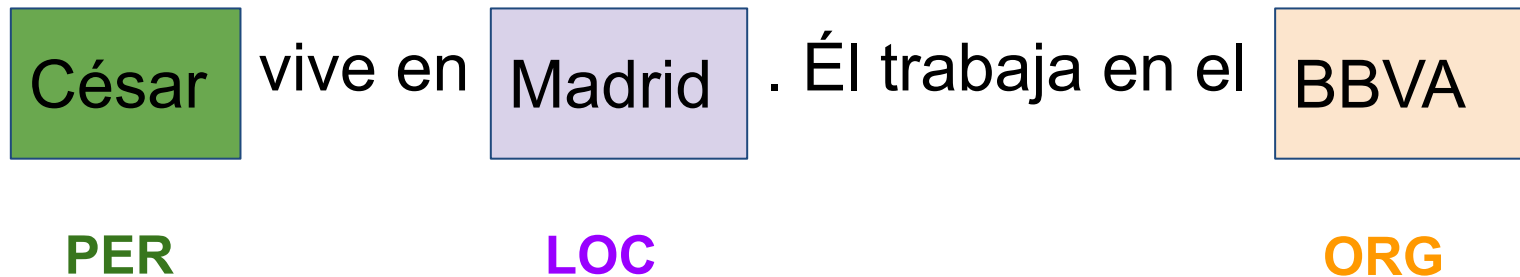
```
graph LR; C1[Concept: síntomas] -- Subject --> A1[Action: dependen]; C2[Concept: tratamiento] -- Subject --> A1; A1 -- Target --> C3[Concept: tipo]; C1 -- is-a --> C2;
```

# Extracción de Información

- **Principales aplicaciones:**
  - Población de bases de datos, diccionarios, ontologías, taxonomías, etc.
  - Mejorar los sistemas de recuperación de información
  - Ayudan en tareas como la traducción automática o generación de resúmenes
  - Facilitan la implementación de sistemas de búsqueda de respuestas, y asistentes virtuales.

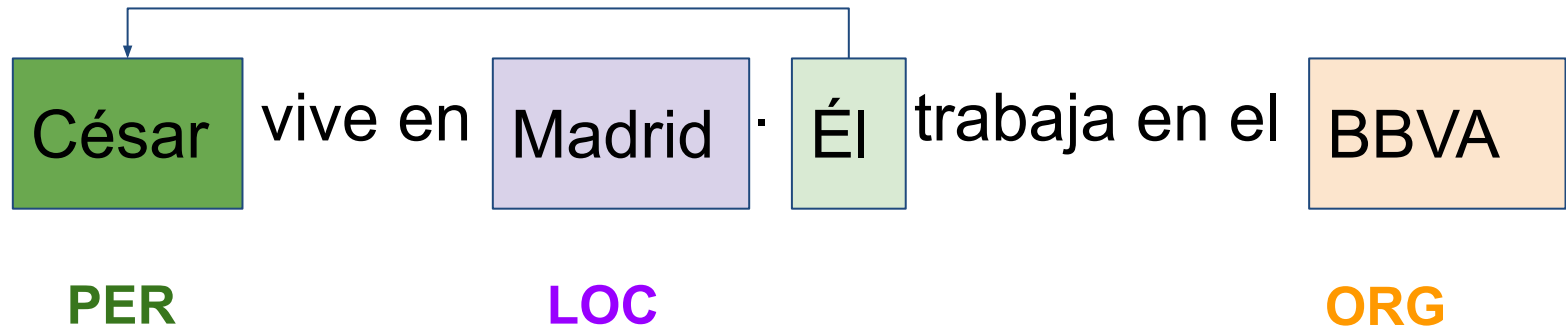
# Extracción de Información. Tareas:

**Reconocimiento de Entidades Nombradas.  
(Named Entity Recognition (NER))**



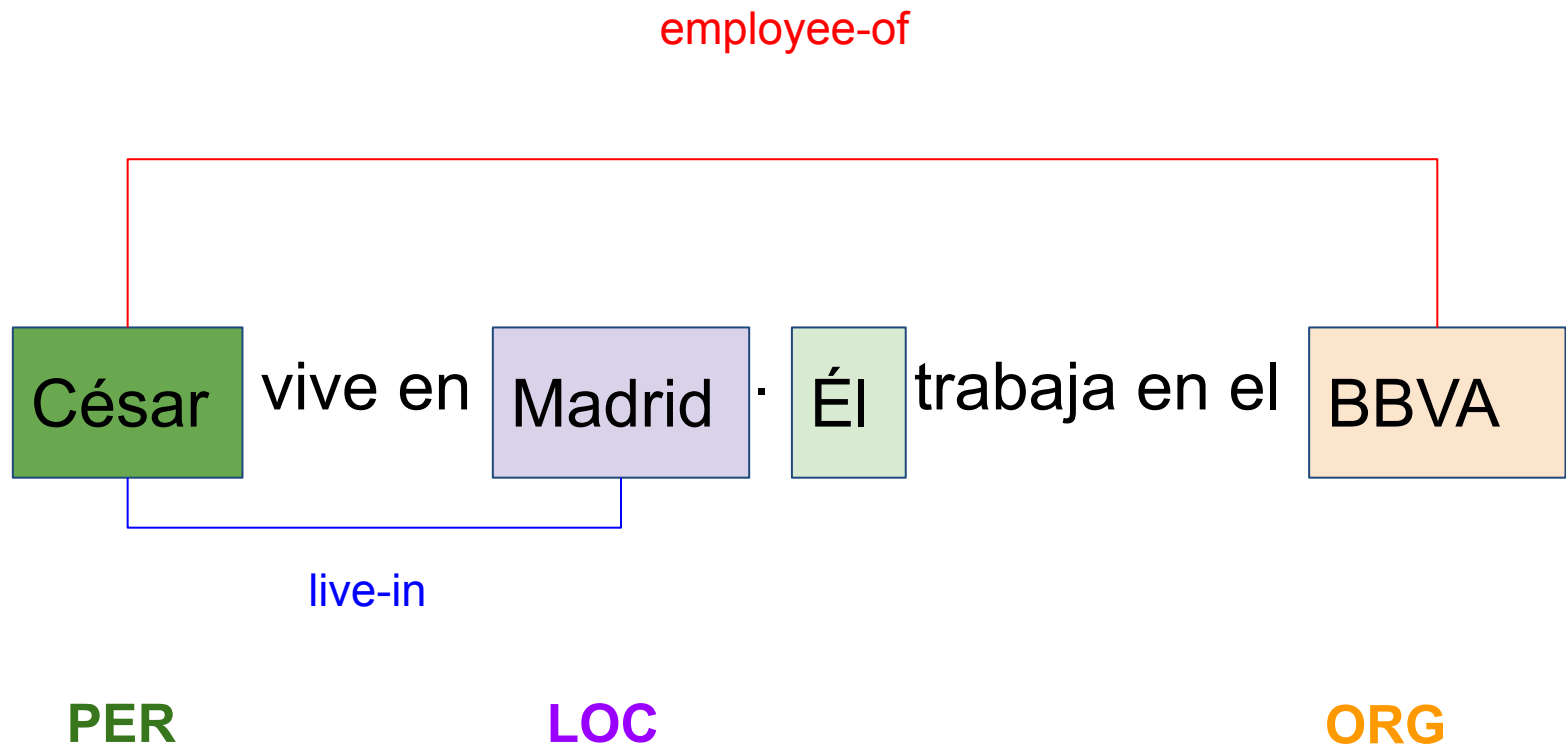
# Extracción de Información. Tareas:

## Resolución de la Correferencia



# Extracción de Información. Tareas:

## Extracción de relaciones



# Extracción de Información

- **Enfoques:**
  - Reglas y diccionarios
  - Aprendizaje automático:
    - algoritmos clásicos: CRF para reconocimiento de entidades y SVM para extracción de relaciones. Requiere definir las características para representar cada instancia.
    - redes profundas: BiLSTM, transformers.

# Ejemplo NER con Spacy

- *Ejemplo NER con Spacy*
- *Otras funcionalidades de Spacy*



# Aplicaciones PLN

- Traducción automática
- Generación de resúmenes
- Recuperación y Extracción de Información
- **Clasificación de textos**
- Agentes conversacionales

# Clasificación de textos

- Asignar una o más categorías de un conjunto predefinido a un texto dado.
  - **binaria**: dos clases.
    - Ejemplo: detectar si un correo es spam o no.
  - **multi-clasificación**: tres o más clases; se asigna una única clase a cada texto.
    - Ejemplo: identificar la opinión de un usuario sobre un determinado producto como buena, neutra, o negativa.
  - **multi-etiquetado**: cada documento puede ser clasificado con varias etiquetas.
    - Ejemplo: clasificación de noticias (una noticia podría estar en una o más categorías de las siguientes: Política, Economía, Salud, Deportes, y Cultura).

# Clasificación de textos

- **Principales aplicaciones:**
  - Filtrado de Spam
  - Análisis de sentimiento
  - Clasificación de noticias
  - Detección de mensajes de odio
  - Clasificación de textos médicos  
(definición de cohortes de pacientes,  
etiquetado con ICD-10, etc)
  - Detección de plagio
  - etc

# Clasificación de textos

- Enfoques
  - reglas (por ejemplo, basadas en palabras claves “oferta”, “gratis”, “premio” etc, pueden ayudar a identificar correo spam.
    - Fácil implementación, pero con pobres resultados (precisión y recall).
  - aprendizaje automático:
    - algoritmos clásicos: por ejemplo, SVM.
    - redes neuronales profundas: CNN, BiLSTM, transformers.

# Un caso práctico para clasificación de textos

- *Pipelines con transformers*
- *Fine-tuning de un modelo transformer para análisis de sentimiento*

# Retos en PLN

- Comprender contextos complejos: Ambigüedad, ironías, sarcasmos o sutilezas del lenguaje.
- Transferencia de conocimiento a dominios especializados y con pocos recursos.
- Modelos Multilingües y Multimodales.
- Baja Explicabilidad de los modelos.
- Sesgos en datos => resultados discriminatorios en aplicaciones del mundo real.

# Algunos recursos interesantes

- [Curso OCW UC3M sobre PLN con aprendizaje profundo](#)
- [NLP Course en Hugging Face](#)
- [Coursera NLP DeepLearningIA](#)
- [NLPprogress](#)

# Gracias por vuestra atención!!!

Isabel Segura Bedmar  
Máster de Lingüística y Tecnología  
Universidad Complutense de Madrid  
18 de septiembre, 2024