Programa de formación
MACHINE LEARNING AND DATA SCIENCE MLDS

Facultad de
INGENIERÍA

UNIVERSIDAD
NACIONAL
DE COLOMBIA

Módulo 4

Procesamiento y Entendimiento del Lenguaje Natural

Unidad 2

Preprocesamiento

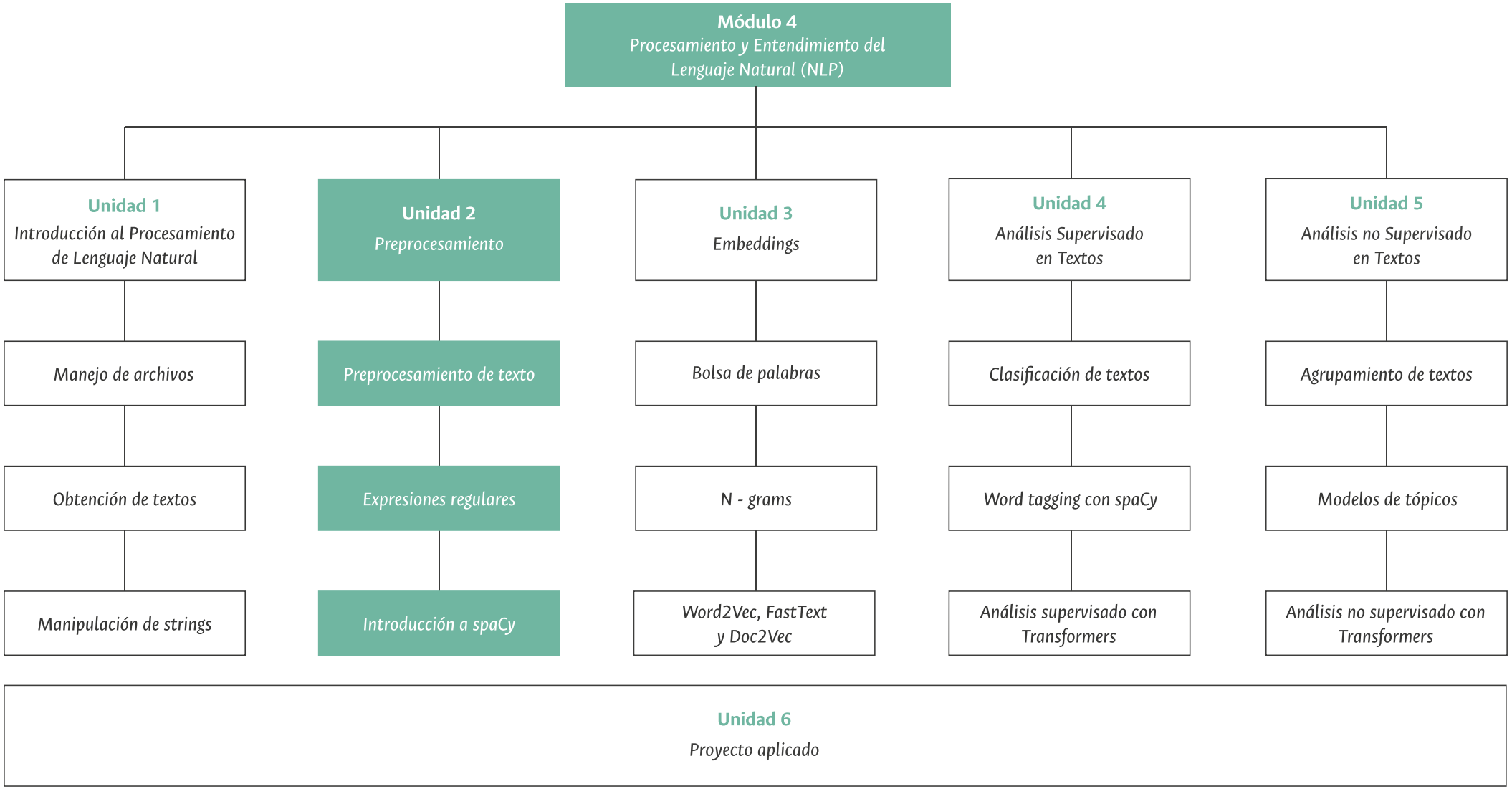
Clase sincrónica

Felipe Restrepo Calle, PhD.

Facultad de INGENIERÍA

UNIVERSIDAD NACIONAL DE COLOMBIA

Mapa de contenidos



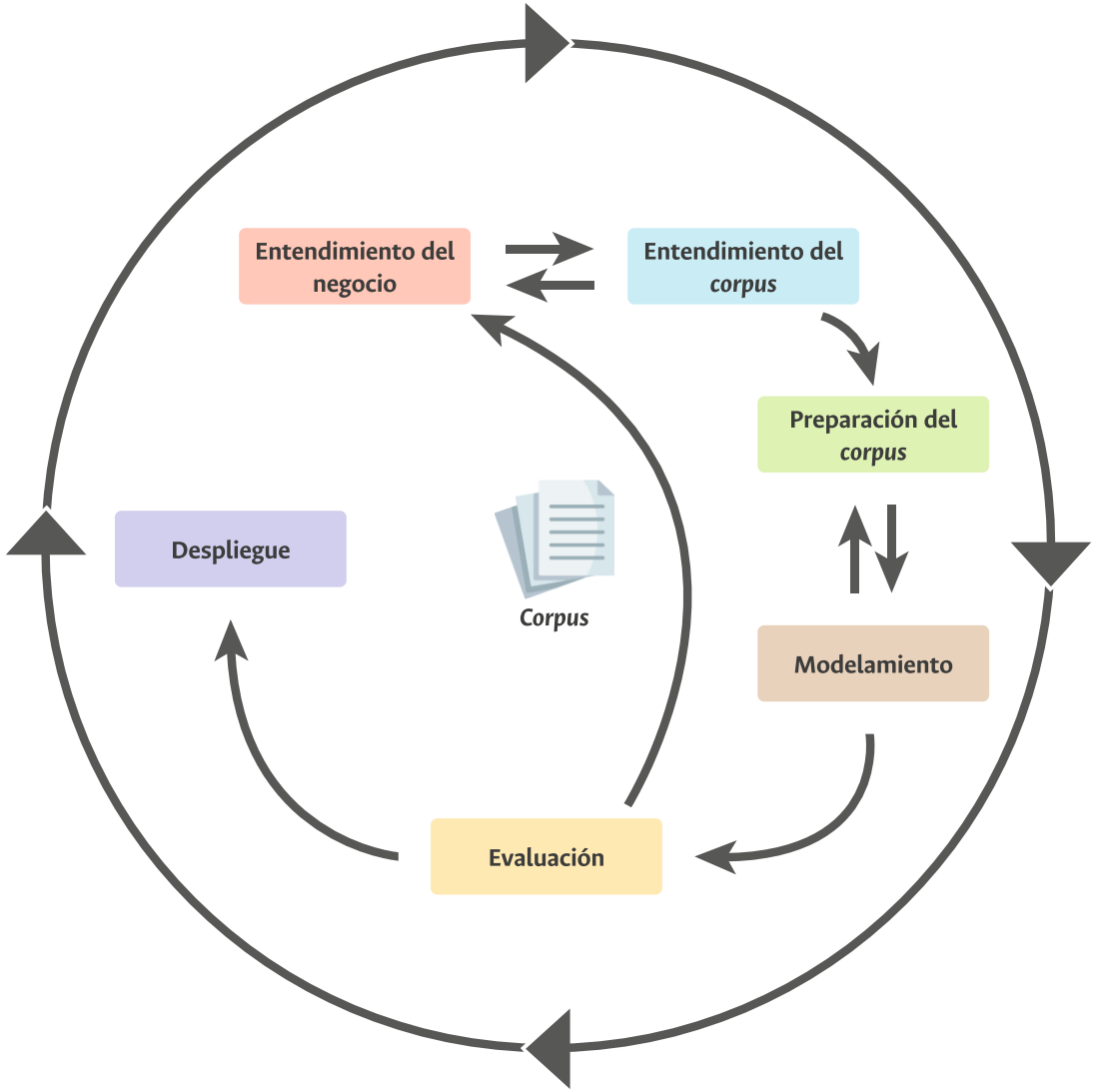
Agenda

1. Ciclo de vida de proyectos de NLP
2. Preparación del *corpus*
   - 2.1 Tokenización
   - 2.2 Filtrado de palabras
   - 2.3 Lematización
   - 2.4 Normalización de textos
   - 2.5 Modificación de grafía
   - 2.6 Expresiones regulares
3. Herramientas para preprocesamiento

Ciclo de Vida en NLP

- El desarrollo de aplicaciones de NLP sigue el mismo enfoque de *Cross-Industry Standard Process for Data Mining* (CRISP-DM)

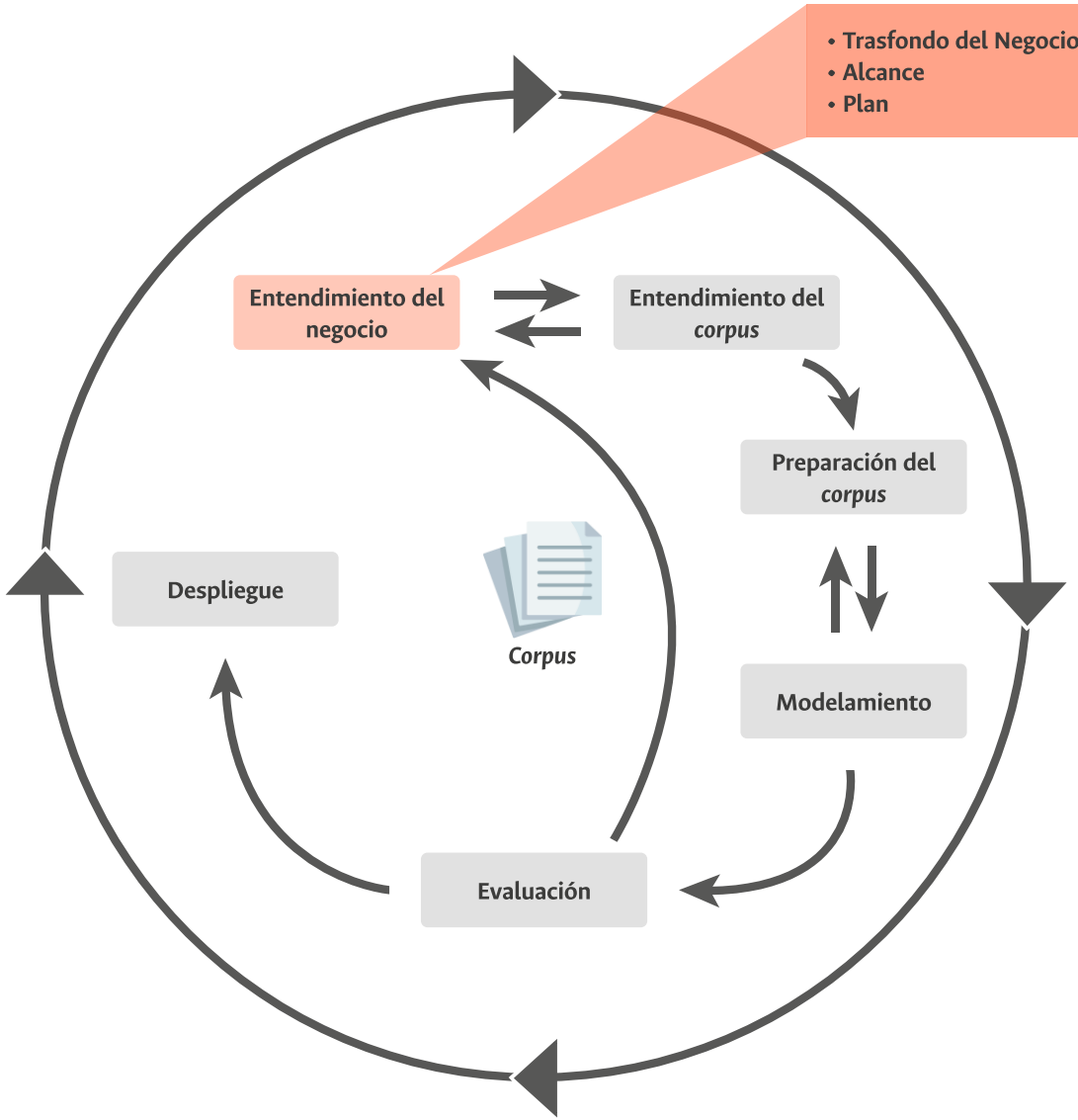
- Esto involucra:
  - Entendimiento del negocio
  - Entendimiento del *corpus*
  - Preparación del *corpus*
  - Modelamiento
  - Evaluación
  - Despliegue



Ciclo de Vida en NLP

Entendimiento del negocio

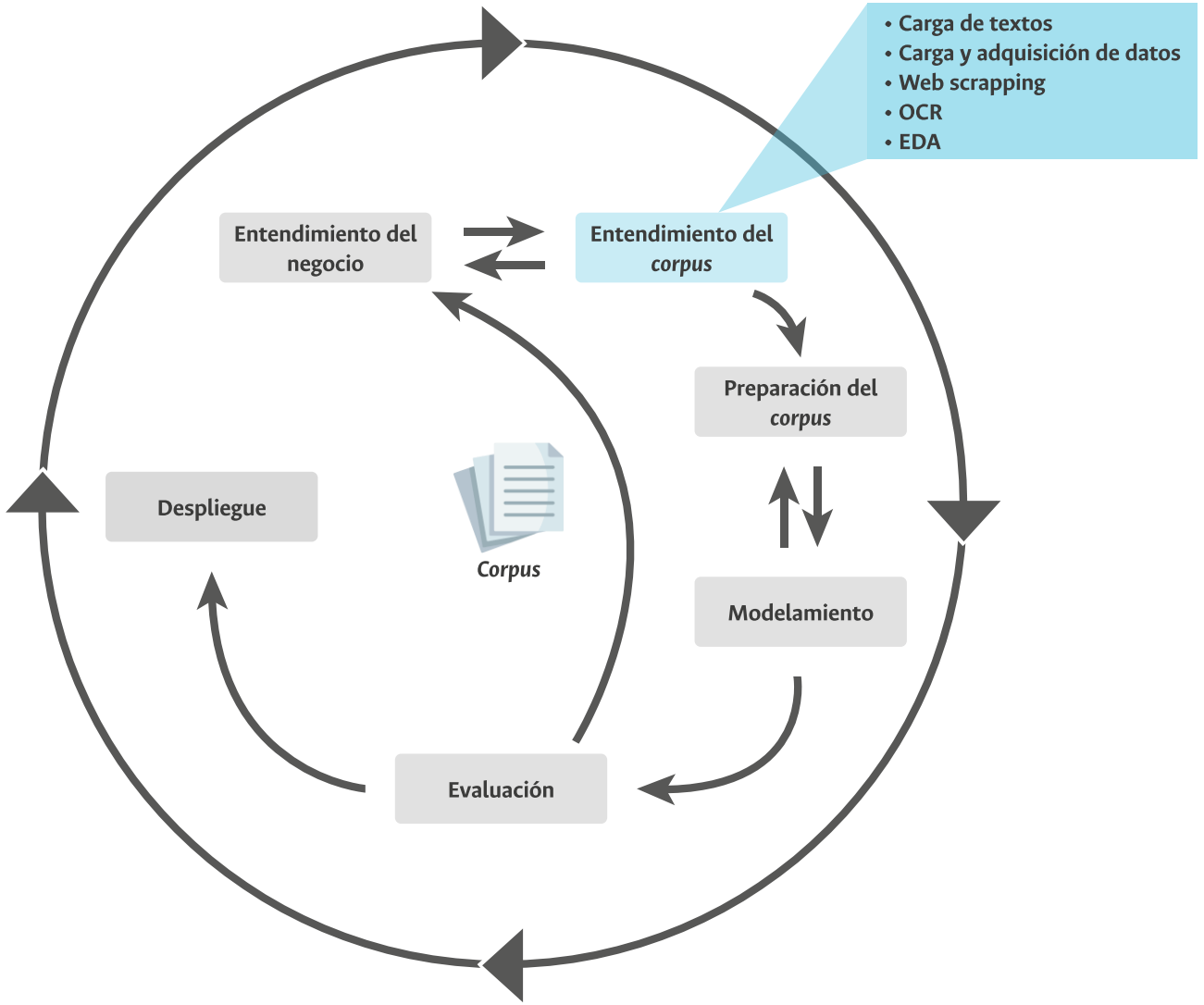
- Comprensión de objetivos y requisitos.
- Identificación de las necesidades empresariales en relación a un proyecto NLP.
- Garantía de éxito y cumplimiento de expectativas en proyectos de NLP.



Ciclo de Vida en NLP

Entendimiento del *corpus*

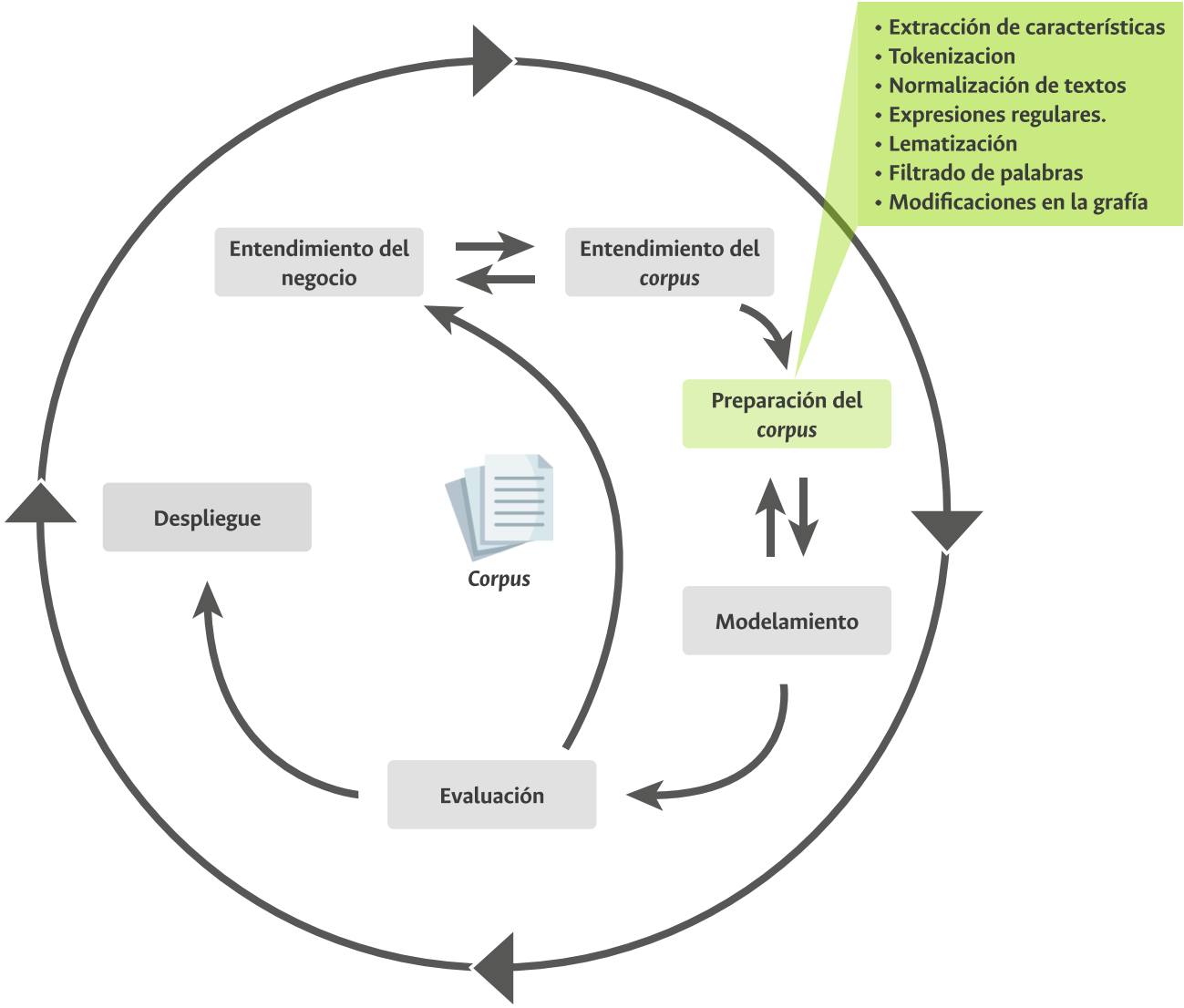
- Conocimiento de calidad, cantidad y relevancia de datos NLP.
- Algoritmos NLP precisos con datos adecuados.
- Identificación y solución de problemas de integridad de datos.



Ciclo de Vida en NLP

Preparación del *corpus*

- La preparación de datos en *NLP* implica la limpieza, normalización y transformación de los datos para su uso en modelos.
- Es un paso clave en el procesamiento de datos para mejorar la precisión y eficiencia de los modelos *NLP*.
- Incluye técnicas como la normalización de textos, la *tokenización*, eliminación de palabras, entre otras.



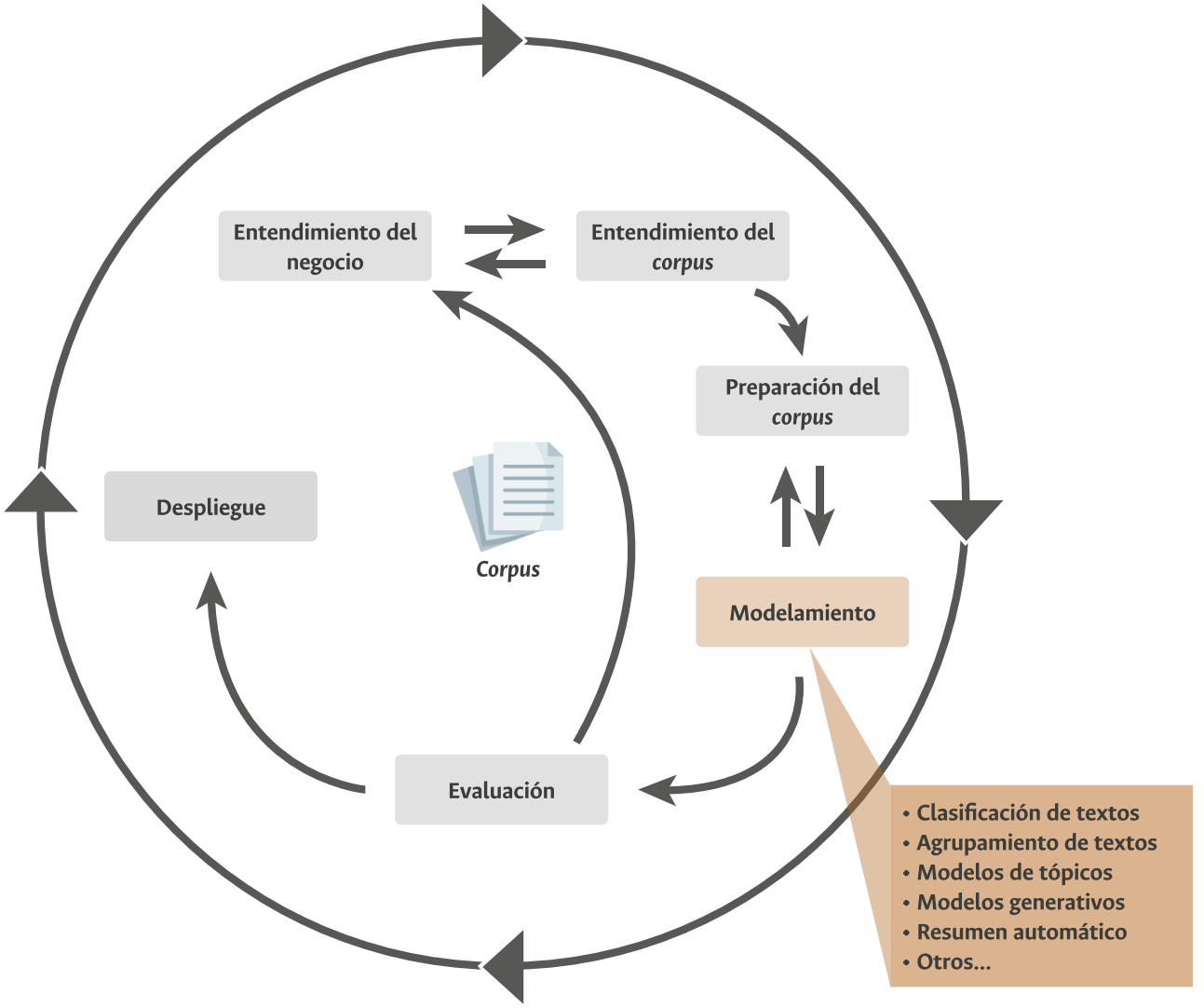
Ciclo de Vida en NLP

Modelamiento

- El modelamiento en *NLP* implica la selección y entrenamiento de modelos para resolver tareas específicas de *NLP* a partir de los datos previamente preprocesados.

- **Tareas de aprendizaje supervisado:** clasificación de tokens, clasificación de textos, traducción automática, llenado de máscaras, resumen abstractivo, *question answering*, ...

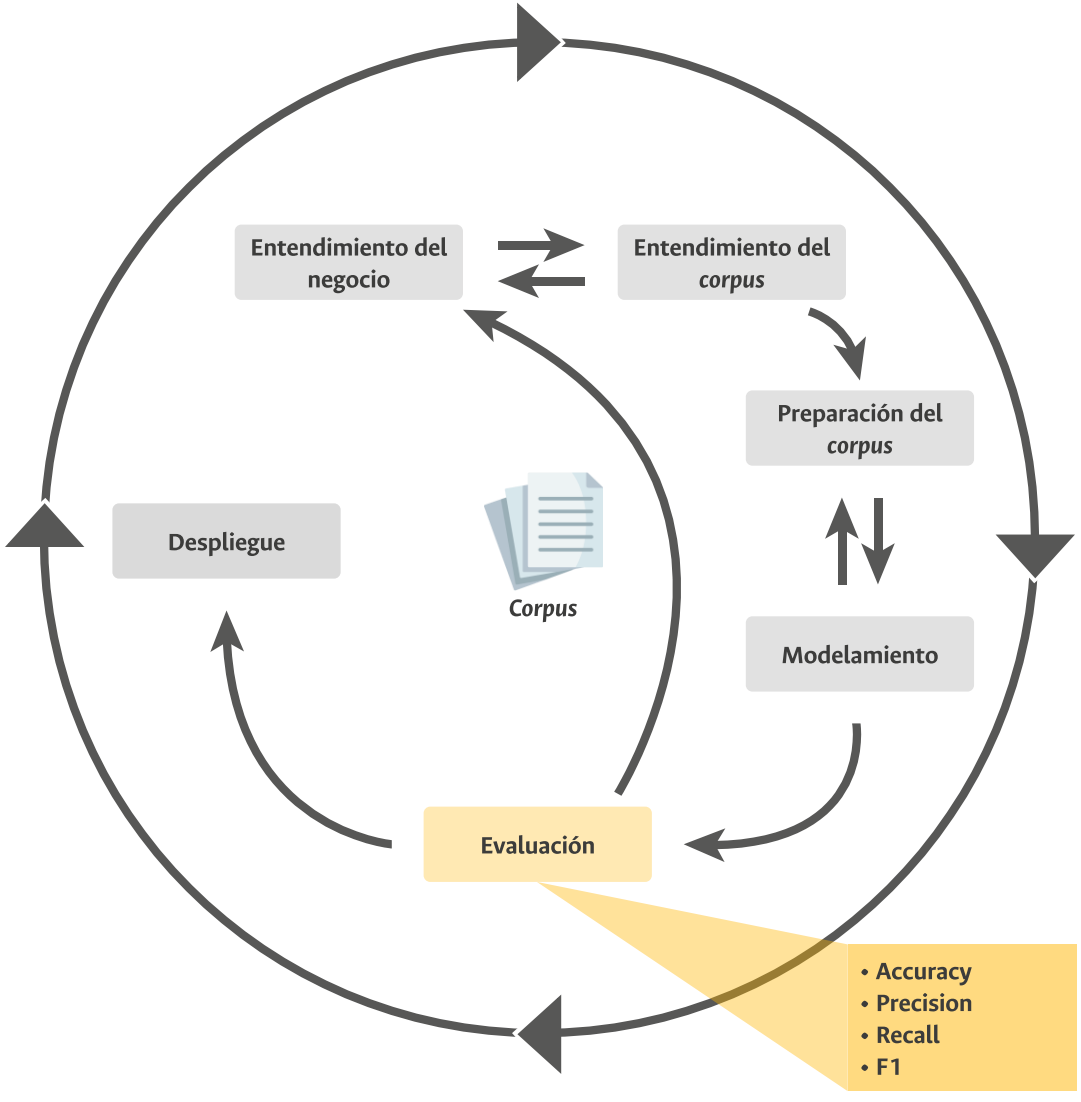
- **Tareas de aprendizaje no supervisado:** similitud semántica, *zero-shot classification*, generación de texto, agrupamiento de textos, modelado de tópicos, ...



Ciclo de Vida en NLP

Evaluación

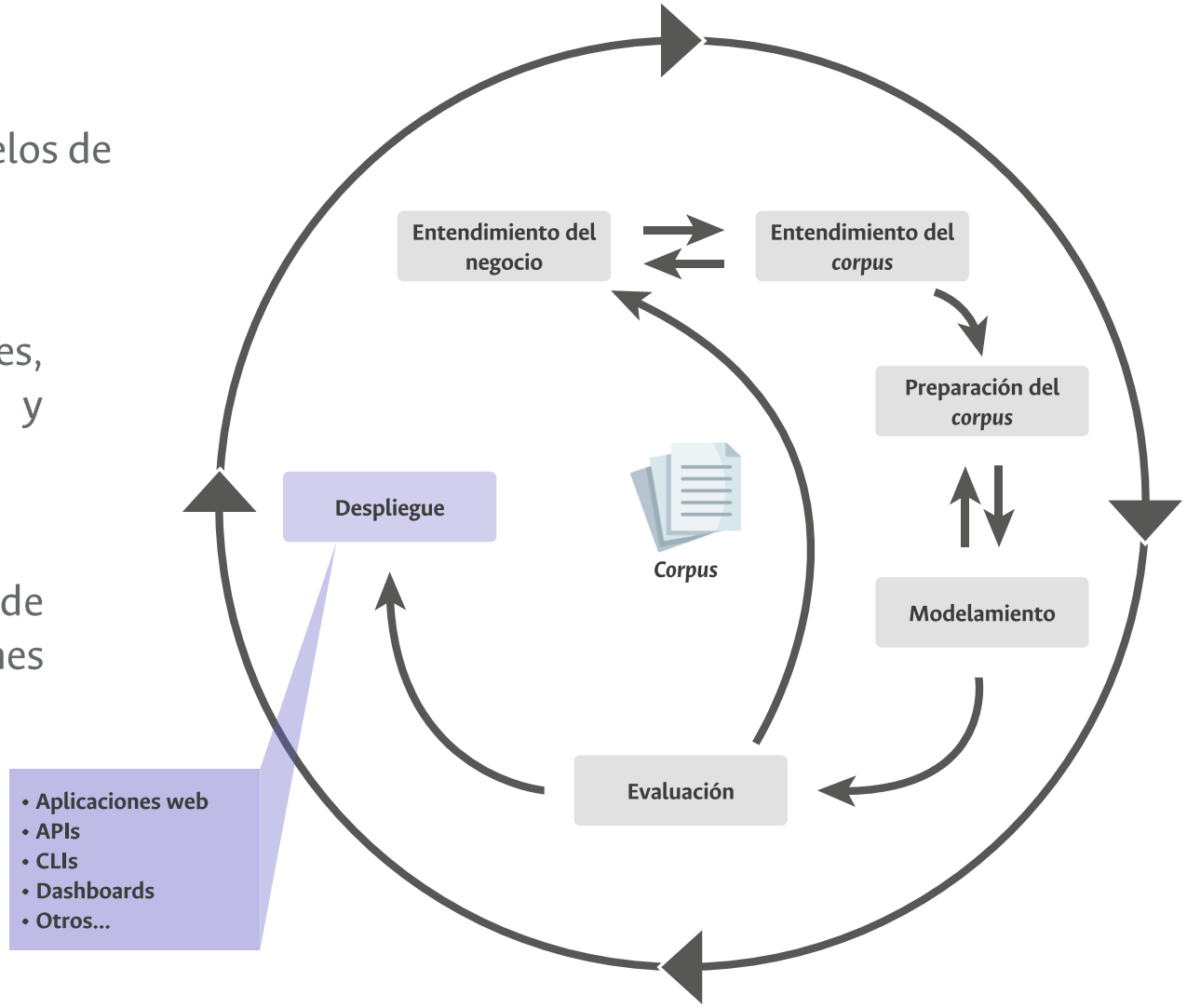
- Permite evaluar el rendimiento de los modelos en tareas específicas de NLP.
- Comparación con datos de referencia y uso de métricas apropiadas.
- Mejora continua de precisión y eficiencia.



Ciclo de Vida en NLP

Despliegue

- Integración y puesta en producción de modelos de NLP.
- Incluye la integración con sistemas existentes, pruebas en ambientes de producción y monitoreo continuo.
- Garantiza la disponibilidad y escalabilidad de los modelos NLP para su uso en aplicaciones empresariales y de usuario final.

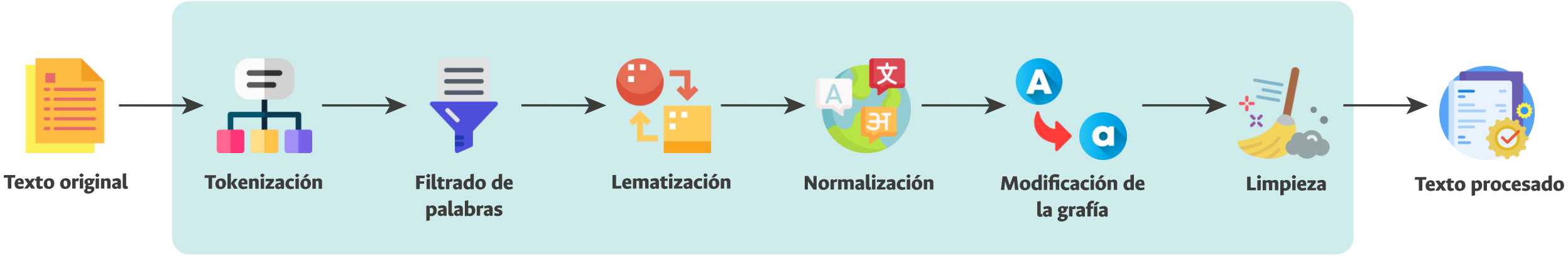


Agenda

1. Ciclo de vida de proyectos de NLP
2. **Preparación del *corpus***
   - 2.1 Tokenización
   - 2.2 Filtrado de palabras
   - 2.3 Lematización
   - 2.4 Normalización de textos
   - 2.5 Modificación de grafía
   - 2.6 Expresiones regulares
3. Herramientas para preprocesamiento

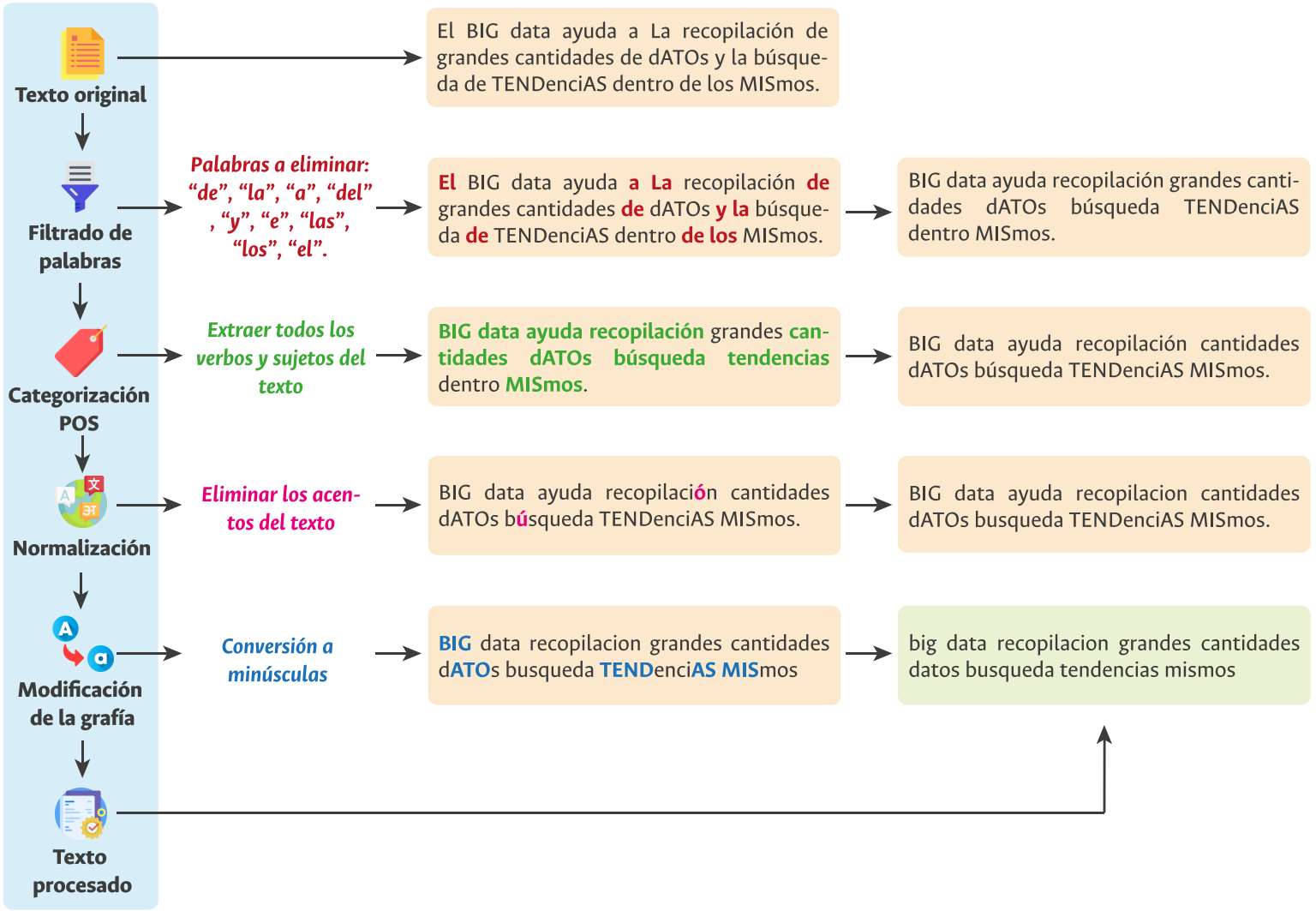
Preparación del *corpus*

Preprocesamiento



- La preparación de los datos en NLP busca dejar los textos en un formato más simple.
- Se aplican técnicas de lingüística computacional y modelos de NLP para la limpieza de datos.

Preparación del *corpus*



Preparación del corpus

Tokenización

- La tokenización en NLP es el proceso de dividir un texto en fragmentos (tokens) significativos.
- Los tokens pueden ser palabras y símbolos del lenguaje, e incluso frases completas, según el objetivo de la aplicación.

1. La RAM almacena los datos de forma temporal.

↓

Lista de tokens de caracter

L, a, R, A, M, a, l, m, a, c, e, n, a, l, o, s, d, a, t, o, s, d, e, f, o, r, m, a, t, e, m, p, o, r, a, l, .

2. El Grey Hat es un tipo de hacker. Actúa ilegalmente, aunque con buenas intenciones. Estos prestan sus servicios a agencias de inteligencia.

↓

Lista de tokens de oración

El Grey Hat es un tipo de hacker, Actúa ilegalmente, aunque con buenas intenciones, Estos prestan sus servicios a agencias de inteligencia

3. El procesador es el cerebro de todo el funcionamiento del sistema y se encarga de ejecutar el código de los diferentes programas.

↓

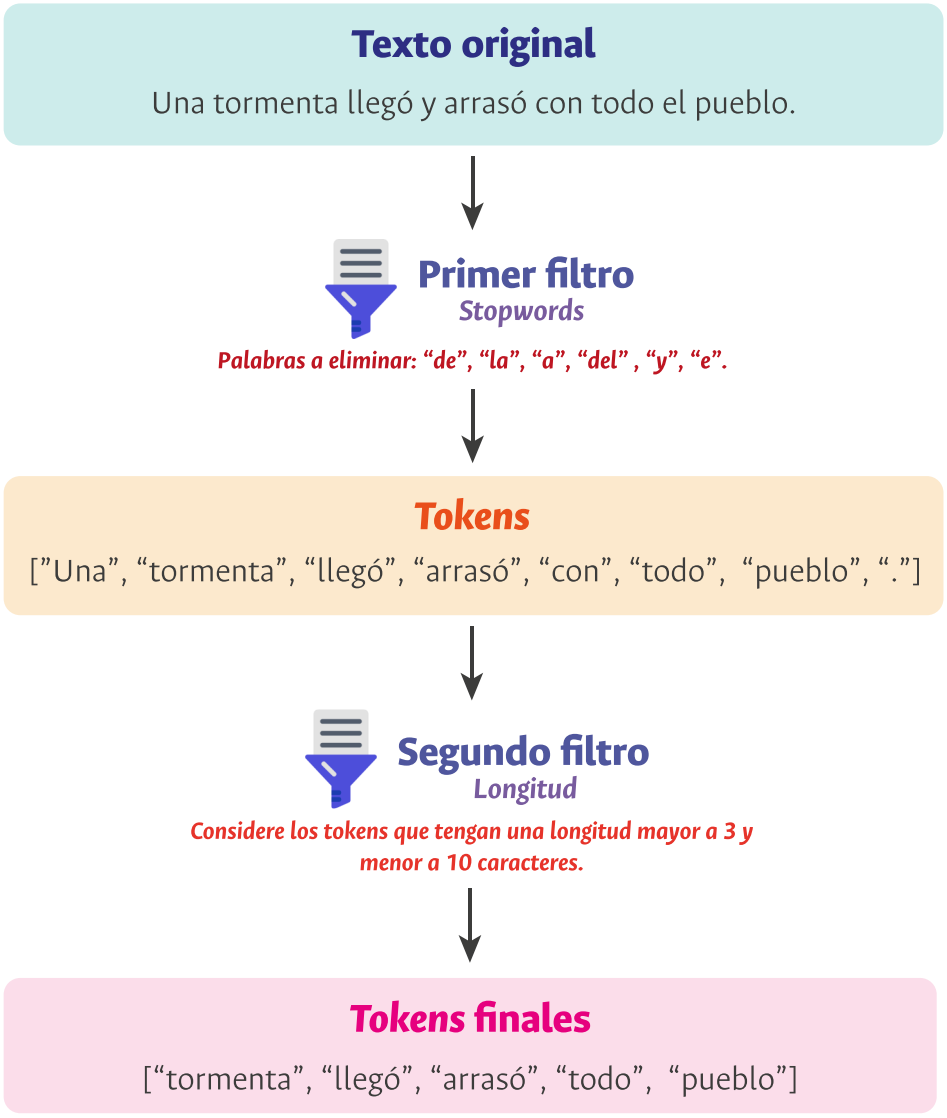
Lista de tokens de palabra

El, procesador, es, el, cerebro, de, todo, el, funcionamiento, del, sistema, y, se, encarga, de, ejecutar, el, código, de, los, diferentes, programas, .

Preparación del corpus

Filtrado de palabras

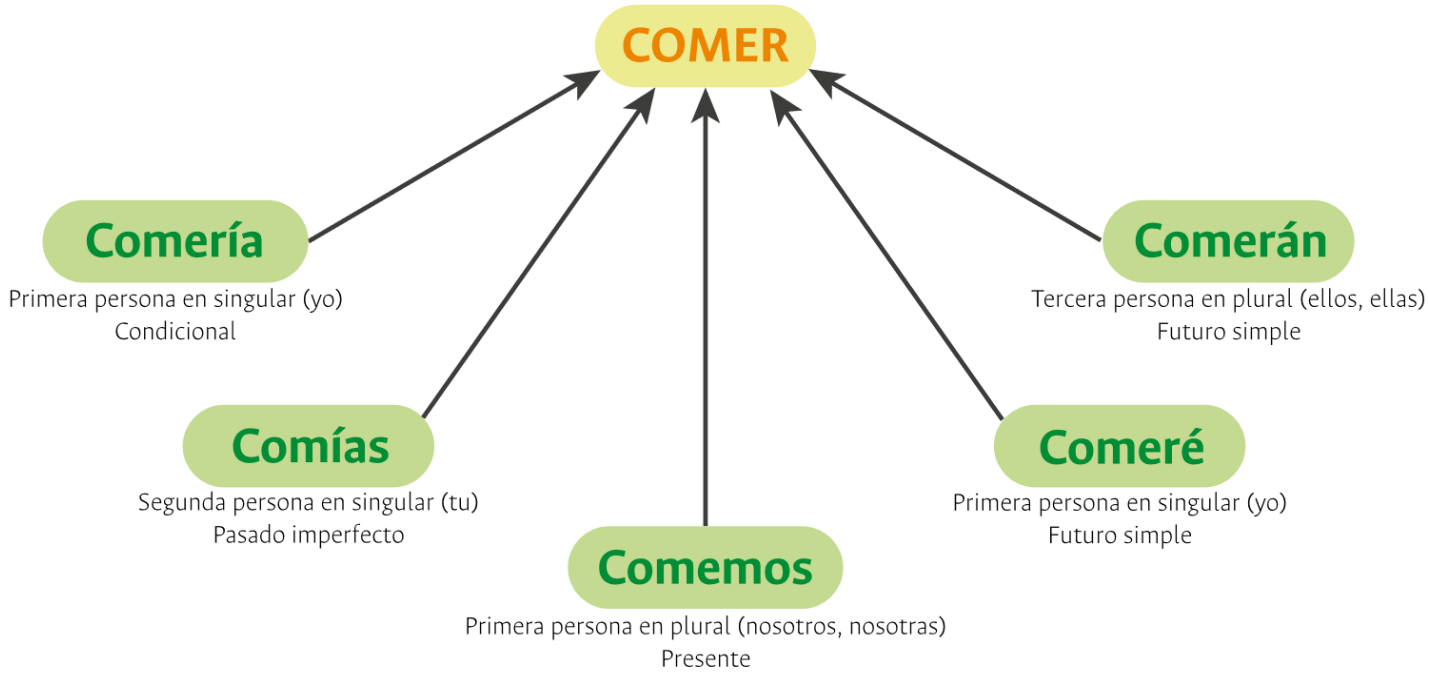
- El filtrado de palabras es el proceso de eliminar o reducir la frecuencia de palabras no relevantes o irrelevantes.
- Mejora la eficiencia y precisión de los modelos NLP al reducir la dimensionalidad y eliminar la interferencia de las palabras irrelevantes.



Preparación del corpus

Lematización

La *lematización* es el proceso de convertir una palabra en su forma base o **lema**.



- Se puede ver como una generalización del proceso de *stemming* (obtener prefijos de palabras), ya que considera estructuras gramaticales (tiempos verbales, conjugaciones y demás) y no únicamente los sufijos.
- Mejora la eficiencia y precisión de los modelos NLP al reducir la dimensionalidad y eliminar la redundancia de los datos.

Preparación del corpus

Normalización de Textos

La *normalización* de texto es el proceso de transformación de texto para la obtención de una forma canónica con el fin de mejorar la eficiencia y precisión de los modelos NLP.

- Incluye acciones como la eliminación de puntuación, eliminación de acentos y la corrección de ortografía.

El perro ladró, la vaca mugió, el gato maulló y el pato graznó y corrió todo lo que pudo. Fue así como transcurrió otro apacible día en la granja. Bueno, quizás, debí decir un día raro, y es que el león, que se había escapado de la selva.

Texto original

Normalización

el perro ladro la vaca mugio el gato maullo y el pato grazno y corrio todo lo que pudo fue asi como transcurrio otro apacible dia en la granja bueno quizas debi decir un dia raro y es que el leon que se habia escapado de la selva

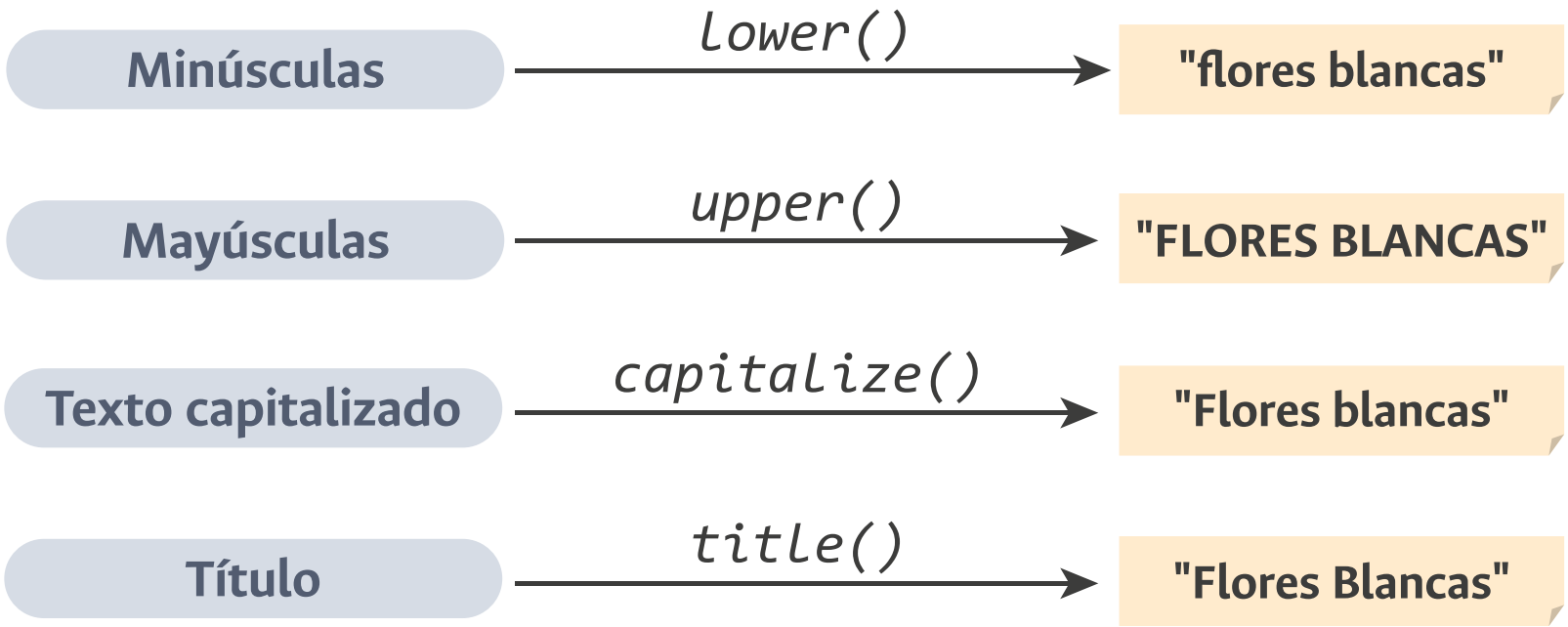
Texto normalizado

Preparación del corpus

Modificación de la Grafía

La modificación de la grafía permite estandarizar textos al eliminar mayúsculas o minúsculas.

- Esto permite eliminar efectos de signos de puntuación y algunos patrones de escritura.

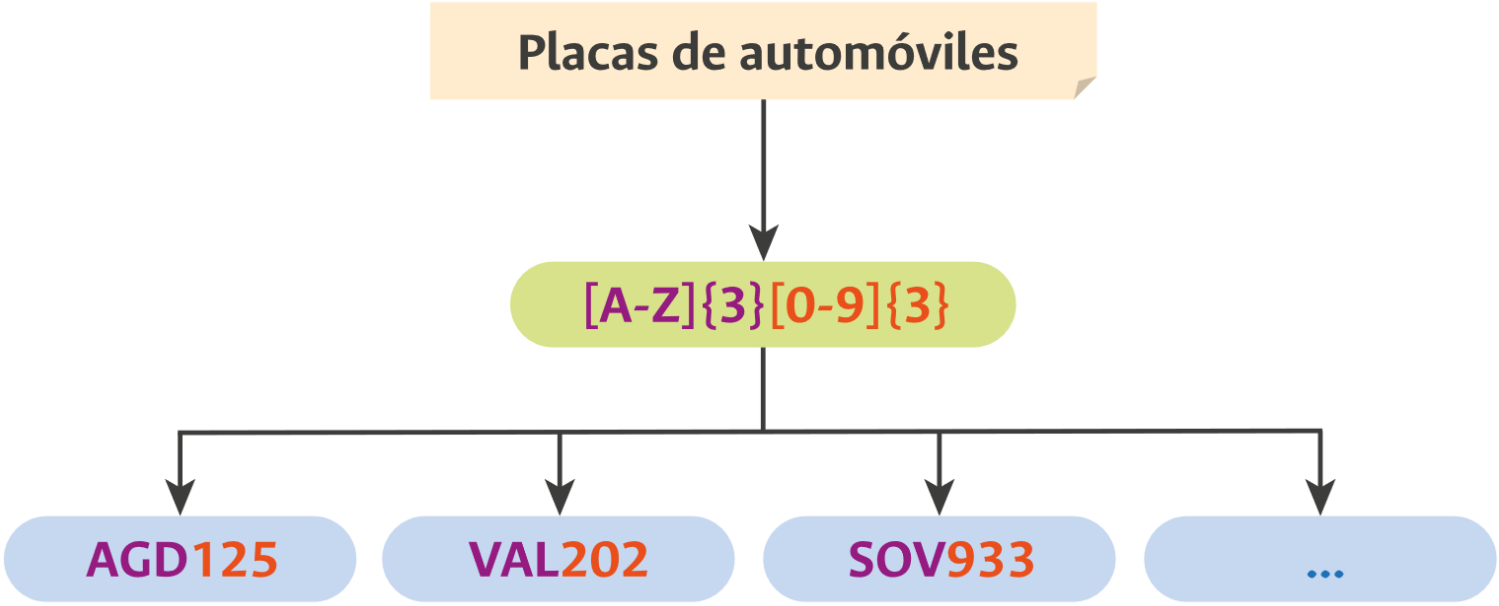


Preparación del corpus

Expresiones Regulares

Las expresiones regulares son un lenguaje de patrones que permite buscar y manipular texto.

- Se utilizan para tareas como la extracción de información, la validación de formato y la limpieza de texto en NLP.



Expresiones Regulares

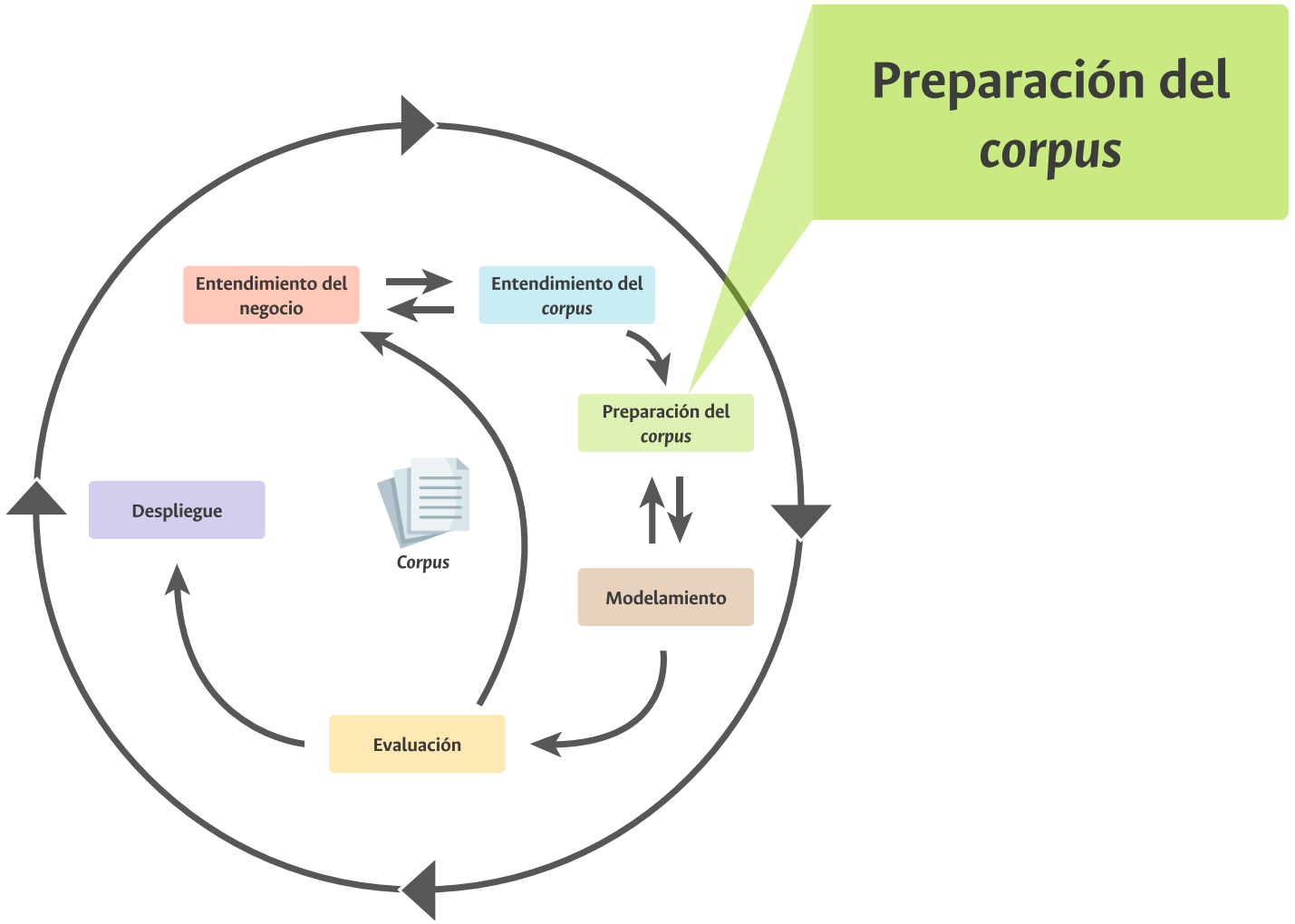
Expresión	Descripción	Ejemplos
`[a-zA-Z]+`	Cadenas que estén compuestas por una o más letras mayúsculas y/o minúsculas.	a, H, l, m, Hola, hola, mAr, flor, ESTRella, safFlsPS, mCnjUD, ...
`[a-z]+[0-9]{2}` `[a-z]+\d{2}`	Cadenas que estén compuestas por una o más letras minúsculas seguidas de dos números del 0 al 9.	base10, planeta52, escuela63, cancion27, iron14, piano99, sfgrd02, ifkos31, ...
`A[a-z]*`	Cadenas que empiecen por una A mayúscula seguida de 0 o más letras minúsculas.	Ala, Ardilla, Ambulancia, Arroyo, Anillo, Alma, Ancdop, Ayrgf, A, ...
`[AEIOU]*[a-z]+[sS]`	Cadenas que empiecen por cero o más vocales mayúsculas y que terminen con la letra *s* minúscula o mayúscula.	Almohadas, Urbanos, Esmeraldas, Ostras, IlusionesssS, AkodckS, pdclss, akodckS, OmskcSS, ...

- Las expresiones regulares son bastante generales, permiten definir patrones sobre *strings*.
- Aunque son muy comunes en pre-procesamiento, también se pueden usar para lematizacion, extracción de frases, entre otras.

Etapa actual del ciclo de vida en NLP

El preprocesamiento de textos es importante en el ciclo de NLP ya que da una versión estandarizada de los datos y posibilita el entrenamiento de modelos.

Hace parte de la etapa de **preparación del *corpus*.**



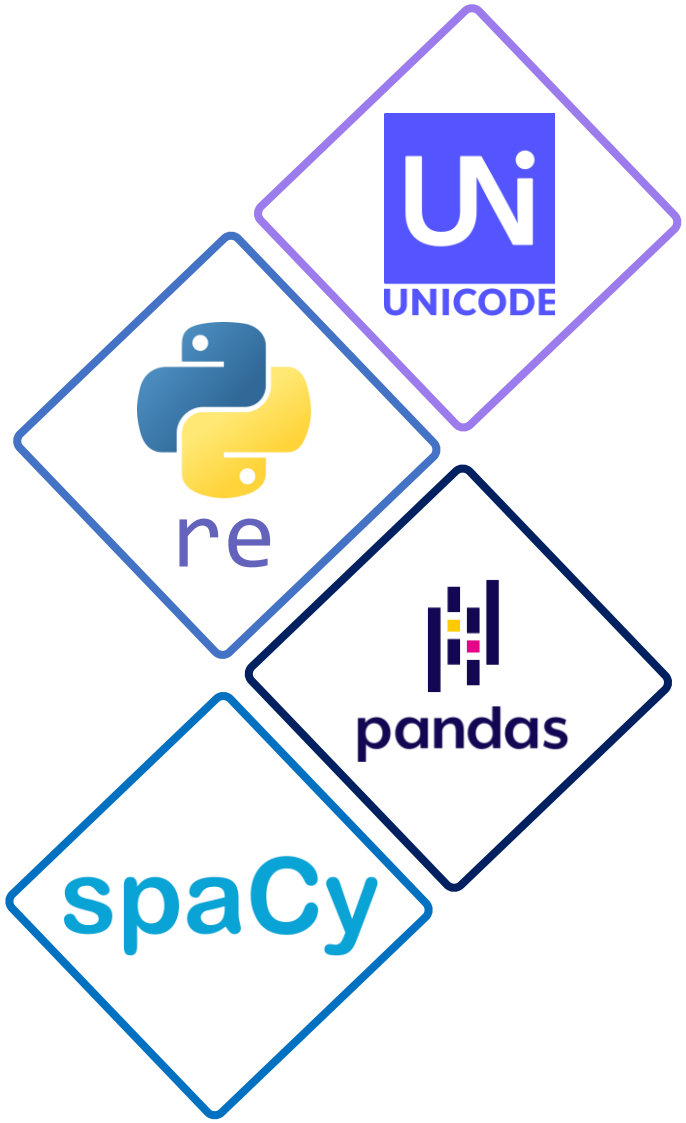
Agenda

1. Ciclo de vida de proyectos de NLP
2. Preparación del *corpus*
   - 2.1 Tokenización
   - 2.2 Filtrado de palabras
   - 2.3 Lematización
   - 2.4 Normalización de textos
   - 2.5 Modificación de grafía
   - 2.6 Expresiones regulares
3. **Herramientas para preprocesamiento**

Herramientas para Preprocesamiento

- Existen distintas librerías para preprocesamiento de textos en Python.

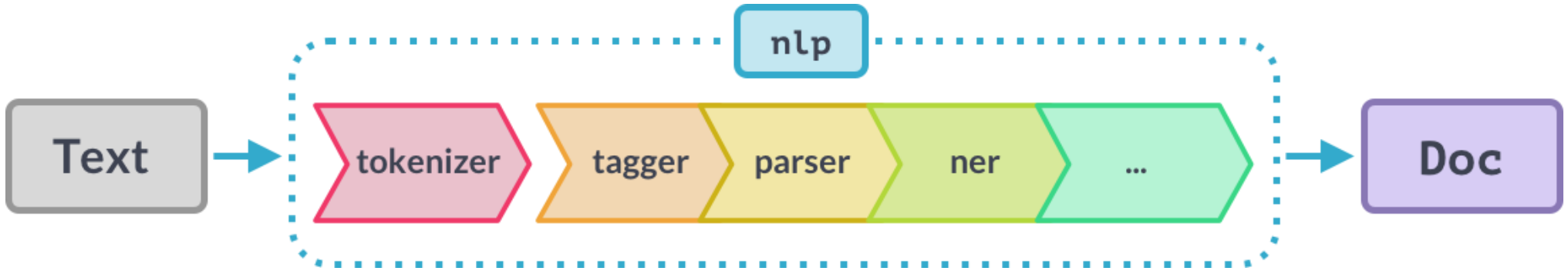
- Entre las más comunes encontramos *spaCy*, re, *unidecode*, *pandas*.



Herramientas para Preprocesamiento

spaCy

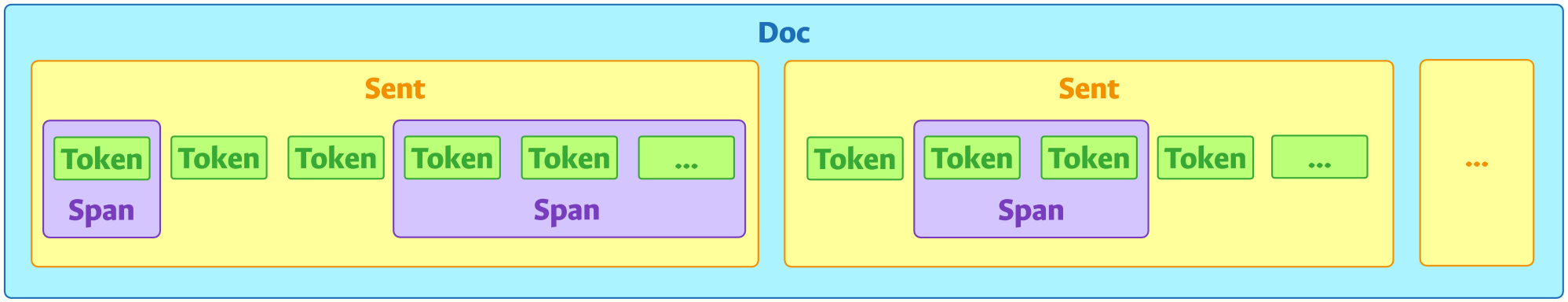
- Ofrece una amplia gama de herramientas para el análisis de texto, incluyendo tokenización, lematización, identificación de *stopwords*, entre otras.
- *spaCy* es una biblioteca de procesamiento de lenguaje natural para Python.



Pipeline de spaCy [Imagen]. Extraída de https://course.spacy.io/en/chapter3

Herramientas para Preprocesamiento / *spaCy*

spaCy nos ofrece una forma sencilla de aplicar distintas técnicas típicas de NLP por medio de un *Pipeline*.



spaCy es una librería de software para procesamiento de lenguajes naturales. spaCy proporciona modelos preentrenados de diferentes lenguajes. Además, junto a una sintaxis clara, hace que sea ideal para principiantes en el campo de la NLP.

- Por medio de clases ***Doc*** y ***Span*** podemos extraer información relevante de todo un documento, o de una secuencia del mismo (entidades nombradas, *part-of-speech*, *tokens*, entre otros).

Herramientas para Preprocesamiento — re

La biblioteca ***re*** (*regular expressions*) en Python es un módulo para trabajar con expresiones regulares. Permite buscar y manipular texto, lo que es útil en tareas como: extracción de información, validación de formato y limpieza de texto.

```
text = "La Guayana Francesa se ubica en América del Sur
pero es territorio de Francia"
```

Expresión regular para buscar **todas las palabras** que empiecen por una mayúscula seguida de una o más minúsculas (incluyendo los acentos)

```
regex = r"[A-Z][a-záéíóú]+"
compile_regex = re.compile(regex)
```

Método *findall()*

```
results = re.findall(compile_regex, text)
>>> ['La', 'Guayana', 'Francesa', 'América',
'Sur', 'Francia']
```

Expresión regular para **sustituir** un *substring* con la preposición "de" seguido de una palabra que comience por una mayúscula seguida por una o más minúsculas **por la palabra "francés"**

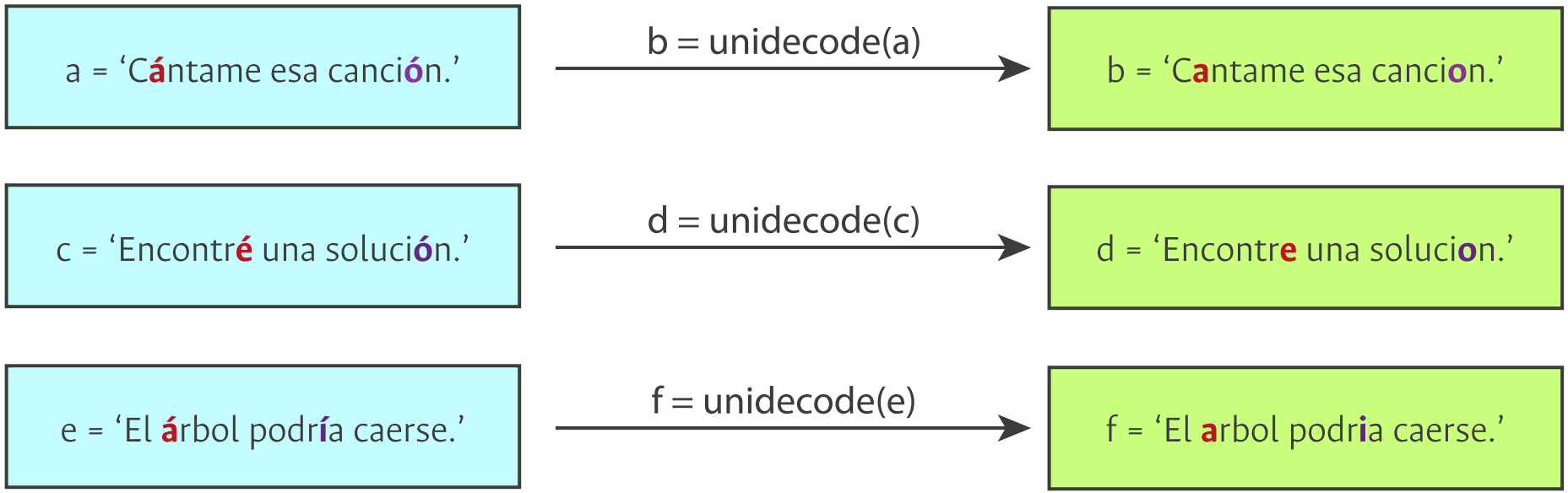
```
to_replace = "francés"
regex = "de [A-Z][a-z]+"
```

Método *sub()*

```
results = re.sub(regex, to_replace, text)
>>> La Guayana Francesa se ubica en América
del Sur pero es territorio francés
```

Herramientas para Preprocesamiento

unidecode



- La biblioteca unidecode en Python es un módulo que proporciona funciones para convertir texto con caracteres Unicode a ASCII.

- Es útil para normalizar y limpiar texto en aplicaciones de NLP, especialmente cuando se trabaja con texto en varios idiomas y alfabetos.

Herramientas para Preprocesamiento — *pandas*

- *pandas* es una biblioteca de análisis de datos en Python que facilita el manejo de datos en formato tabular.
- En NLP, se utiliza para cargar, manipular y preparar los datos.
- *pandas* ofrece funciones de manipulación de strings que son muy útiles para manipular corpus completos.

Helado	Color
MAraCuYÁ	AmarILlO
FRESa	ROjo
Mora	MorAdO
LImÓN	VERDE
MandaRIna	ANaranJaDo

```
df.assign(Helado = df.Helado.str.lower())
```

Helado	Color
maracuyá	AmarILlO
fresa	ROjo
mora	MorAdO
limón	VERDE
mandarina	ANaranJaDo

Referencias

- Bird, S., Klein, E., Loper, E. (2019). *Natural Language Processing with Python. Capítulo 3: Processing Raw Text.* (Primera edición). *O'Reilly Media.* Recuperado de https://www.nltk.org/book/
- Clark, A., Fox, C., Lappin, S. (2010). *The Handbook of Computational Linguistics and Natural Language Processing.* (Primera edición). *Blackwell Publishing Ltd.* Recuperado de http://santini.se/teaching/sais/ClarkEtAl2010_HandbookNLP.pdf

---

- Linguistic Features. (7 de noviembre de 2016). *spaCy*. https://spacy.io/usage/linguistic-features
- Rule-based matching. (1 de noviembre de 2016). *spaCy*. https://spacy.io/usage/rule-based-matching
- Language Processing Pipelines. (1 de noviembre de 2016). *spaCy*. https://spacy.io/usage/processing-pipelines

Derechos de imágenes

- Freepik. (s. f.). File free icon. [Icono]. https://es.wikipedia.org/wiki/Archivo:Python.svg
- The Go Authors. (26 de abril de 2018). Pandas Logo [Vector]. https://en.wikipedia.org/wiki/File:New_Unicode_logo.svg
- s.A. (19 de febrero de2016). SpaCy Logo [Vector]. https://es.wikipedia.org/wiki/Archivo:SpaCy_logo.svg
- Marc Garcia. (22 de octubre de 2019). https://en.wikipedia.org/wiki/File:Pandas_logo.svg
- Flaticon. (s.f.). Paper free icon. [Icono]. https://www.flaticon.com/free-icon/paper_2541984
- Flaticon. (s.f.). Number 3 free icon. [Icono]. https://www.flaticon.com/free-icon/number-3_3840739
- Flaticon. (s.f.). Number 1 free icon. [Icono]. https://www.flaticon.com/free-icon/number-one_3840653
- Flaticon. (s.f.). Number 2 free icon. [Icono]. https://www.flaticon.com/free-icon/number-2_3840738

Créditos

Facultad de
INGENIERÍA

Profesor
Felipe Restrepo Calle, PhD

Asistente docente
Juan Sebastián Lara Ramírez

Coordinador de virtualización
Edder Hernández Forero

Diagramadora PPT
Rosa Alejandra Superlano Esquibel

Diseño gráfico
Clara Valeria Suárez Caballero
Milton R. Pachón Pinzón

2024