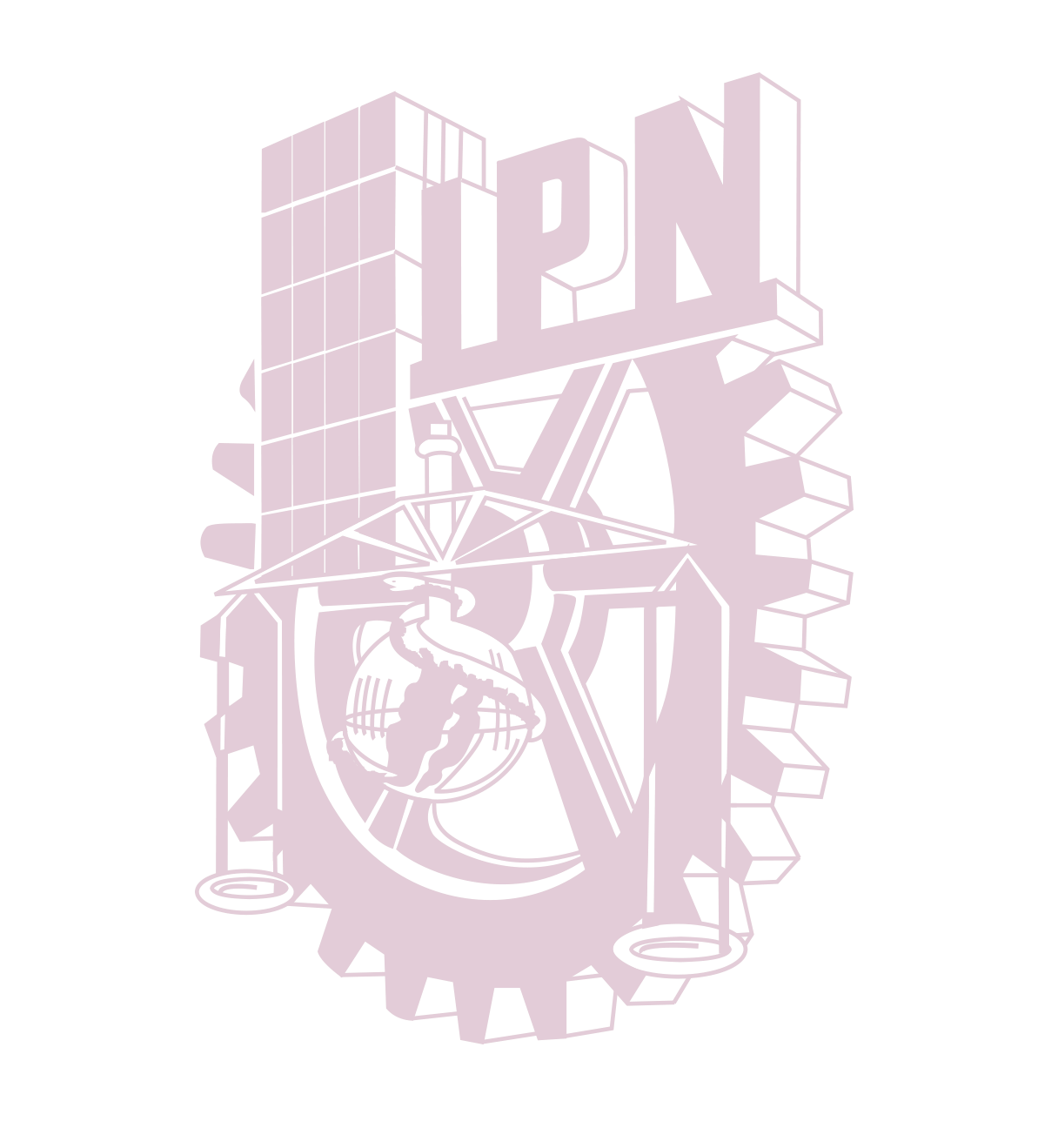


**Instituto Politécnico Nacional**

Escuela Superior de Cómputo

Practica 1

**Materia: TEORÍA DE LA COMPUTACIÓN**

**Grupo:** 5BV1

**Alumno:**

Calderon Sabbagh Juan Alberto

**Fecha:** 28/09/2023

**Nombre del profesor:** **CAMACHO VAZQUEZ VANESSA ALEJANDRA**

Contenido

[Procesamiento para el texto en español. 2](#_Toc146843695)

[Investigación. 2](#_Toc146843696)

[NLTK 2](#_Toc146843697)

[Spacy 2](#_Toc146843698)

[TextBlob 2](#_Toc146843699)

[Analisis exploratorio de textos. 3](#_Toc146843700)

[Identificar el numero total de tokens dentro del texto 3](#_Toc146843701)

[Identificar el numero total de tokens únicos dentro del texto 3](#_Toc146843702)

[Histograma de tokens únicos dentro del texto 4](#_Toc146843703)

[10 tokens más comunes 5](#_Toc146843704)

[10 tokens menos comunes 5](#_Toc146843705)

[Normalizacion del texto 6](#_Toc146843706)

[Pasar a minúsculas 6](#_Toc146843707)

[Eliminación de caracteres especiales, signos de puntuación y abreviaturas (solo en inglés). 6](#_Toc146843708)

[Pos-Tagging para mantener tokens con etiquetas de mayor relevancia. 7](#_Toc146843709)

[Eliminación de stopwords 7](#_Toc146843710)

[Lematización 7](#_Toc146843711)

[Stemming 7](#_Toc146843712)

[Exploración de texto a posterior normalización 8](#_Toc146843713)

[Numero total de tokens en el texto 8](#_Toc146843714)

[Numero de tokens únicos en el texto 8](#_Toc146843715)

[Histograma de tokens únicos dentro del texto 9](#_Toc146843716)

[Lista 10 tokens más comunes 10](#_Toc146843717)

[Lista 10 tokens menos comunes 10](#_Toc146843718)

[Analisis y conclusions 10](#_Toc146843719)

# Procesamiento para el texto en español.

## Investigación.

### NLTK

* Idiomas: NLTK es una biblioteca de procesamiento de lenguaje natural ampliamente utilizada que admite varios idiomas, incluyendo inglés y español.
* Facilidad de uso: NLTK es conocida por ser fácil de usar y es una excelente opción para quienes están aprendiendo procesamiento de lenguaje natural.
* Recursos lingüísticos: NLTK proporciona una amplia variedad de recursos lingüísticos predefinidos, como listas de palabras en varios idiomas, lemas y etiquetadores POS (Part-of-Speech) para inglés y español.
* Limpieza de texto: NLTK incluye funciones para tokenizar, eliminar stopwords, lematizar y etiquetar el texto, lo que facilita la preparación de los datos antes del análisis.

### Spacy

* Idiomas: spaCy es otra biblioteca de procesamiento de lenguaje natural popular que también admite varios idiomas, incluyendo inglés y español.
* Eficiencia: spaCy está diseñada para ser altamente eficiente y rápida, lo que la hace ideal para procesar grandes cantidades de texto de manera rápida y precisa.
* Modelos pre-entrenados: spaCy ofrece modelos pre-entrenados de alta calidad para varios idiomas, incluyendo modelos específicos para el procesamiento de español y inglés. Estos modelos incluyen etiquetadores POS, análisis sintáctico, reconocimiento de entidades nombradas y más.
* Facilidad de extensión: spaCy es altamente modular y permite la extensión con componentes personalizados para adaptarse a tareas específicas de procesamiento de texto.

### TextBlob

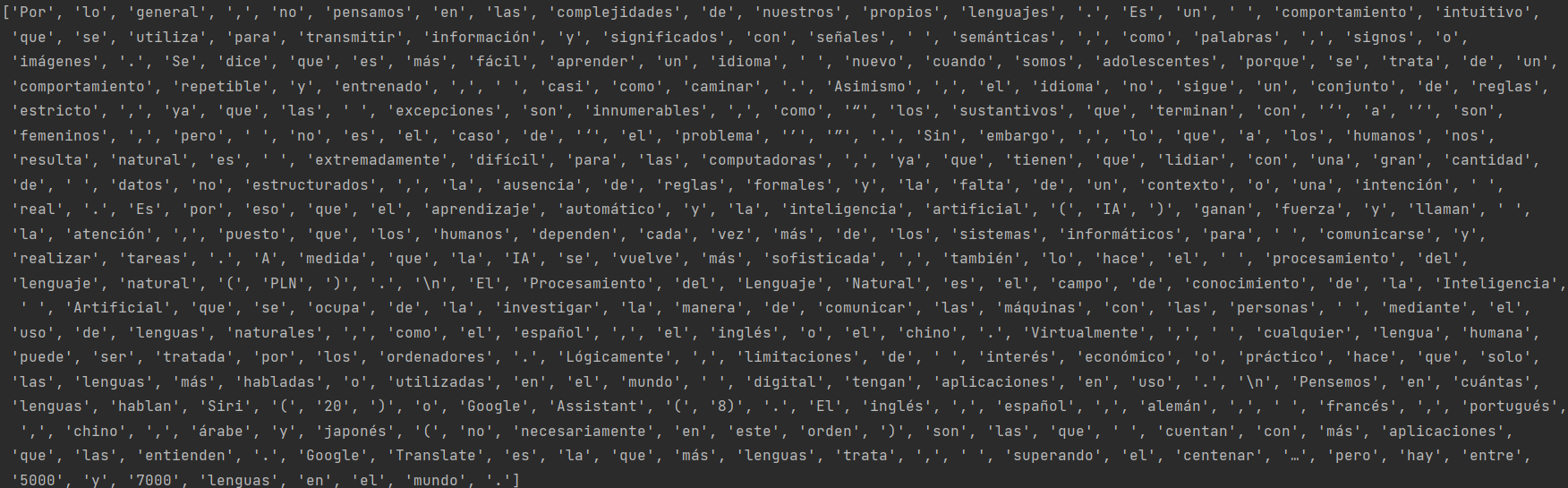
* Idiomas: TextBlob es una biblioteca que simplifica el procesamiento de texto en varios idiomas, incluyendo español e inglés.
* Facilidad de uso: Es conocida por su facilidad de uso y su simplicidad. Ofrece una API sencilla para tareas comunes de procesamiento de texto como tokenización, análisis de sentimientos, extracción de frases clave y corrección ortográfica.
* Análisis de sentimientos: TextBlob incluye un módulo para el análisis de sentimientos que permite determinar si un texto tiene una connotación positiva o negativa.
* Integración con WordNet: Ofrece integración con WordNet para la lematización y búsqueda de sinónimos en inglés.

El módulo que yo voy a utilizar es Spacy ya que me ofrece bastantes funciones y tiene un buen soporte tanto para inglés como para español, además el uso de esta librería me servirá para a la larga poder desarrollar proyectos de mayor complejidad. Igual se usará NLTK para la tarea de la de stemming ya que Spacy no cuenta con dicha función.

## Analisis exploratorio de textos.

En esta seccion se explorara el texto en cuestión y nos familiarizaremos con el texto mediante la obtención de sus tokens, identificación de los tokens mas y menos comunes dentro del texto y además vamos a ver cuales son los tokens menos comunes dentro del texto

### Identificar el numero total de tokens dentro del texto

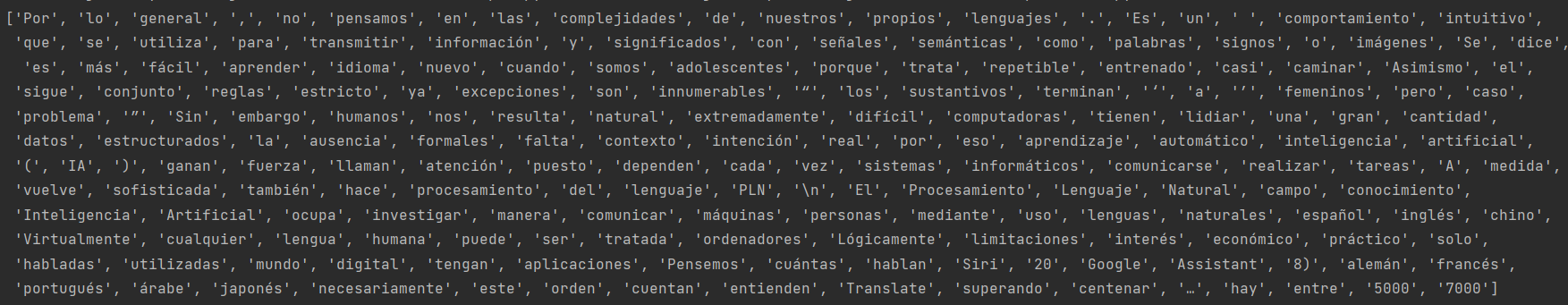
Al realizar la tokenizacion se obtienen 396 tokens, entre estos están caracteres especiales y signos de puntuación.

*1.1 Lista de tokens encontrados dentro del texto*

*1.2 Numero de tokens obtenido mediante len(nombre\_Array)*

### Identificar el numero total de tokens únicos dentro del texto

Para identificar los tokens únicos, se va recorriendo la lista de todos los tokens y en cada iteración se meten a un array y se va comprobando que estos tokens no existan dentro de este array para ser metidos. Finalmente se obtienen 197 tokens únicos.



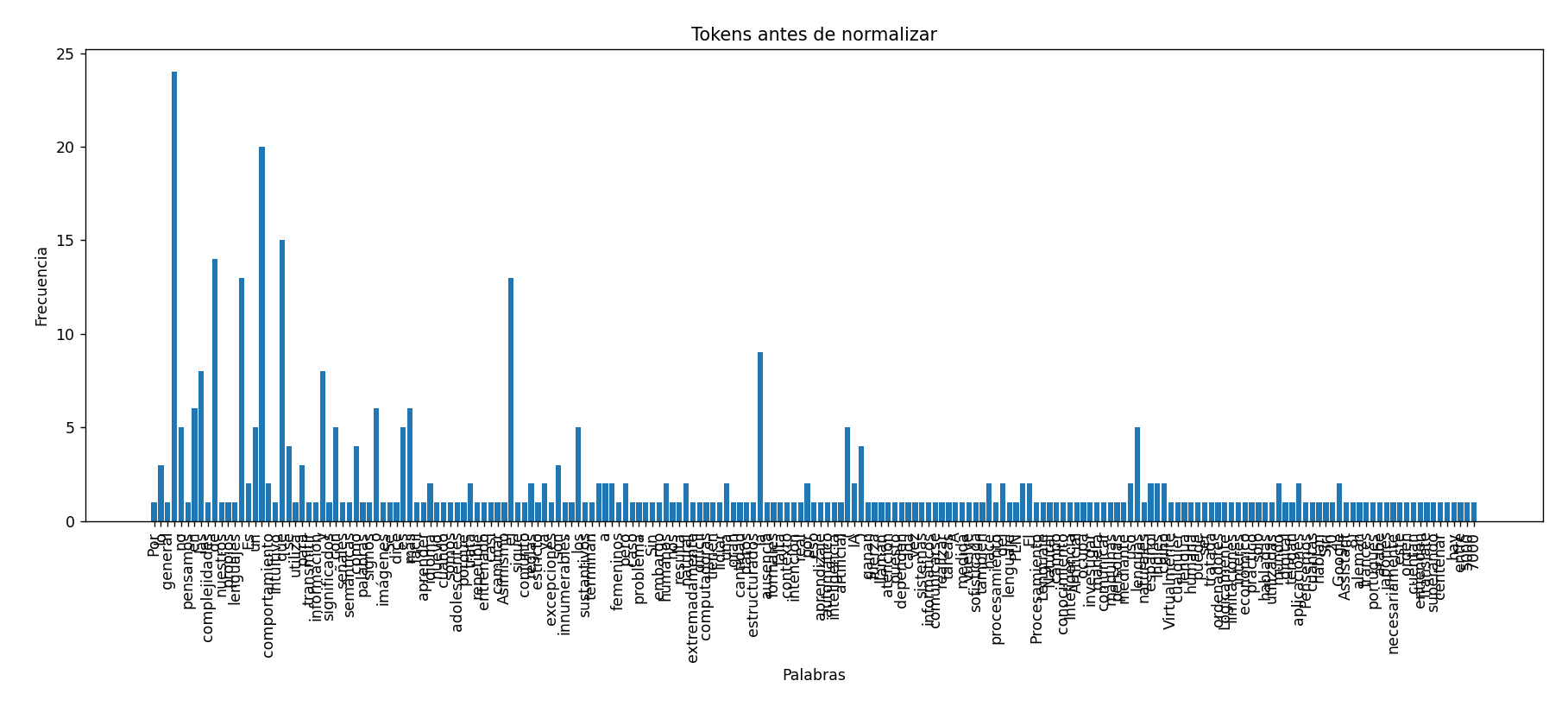
*1.3 Lista de tokens únicos*

**

*1.4 Numero de tokens obtenido mediante len(nombre\_Array)*

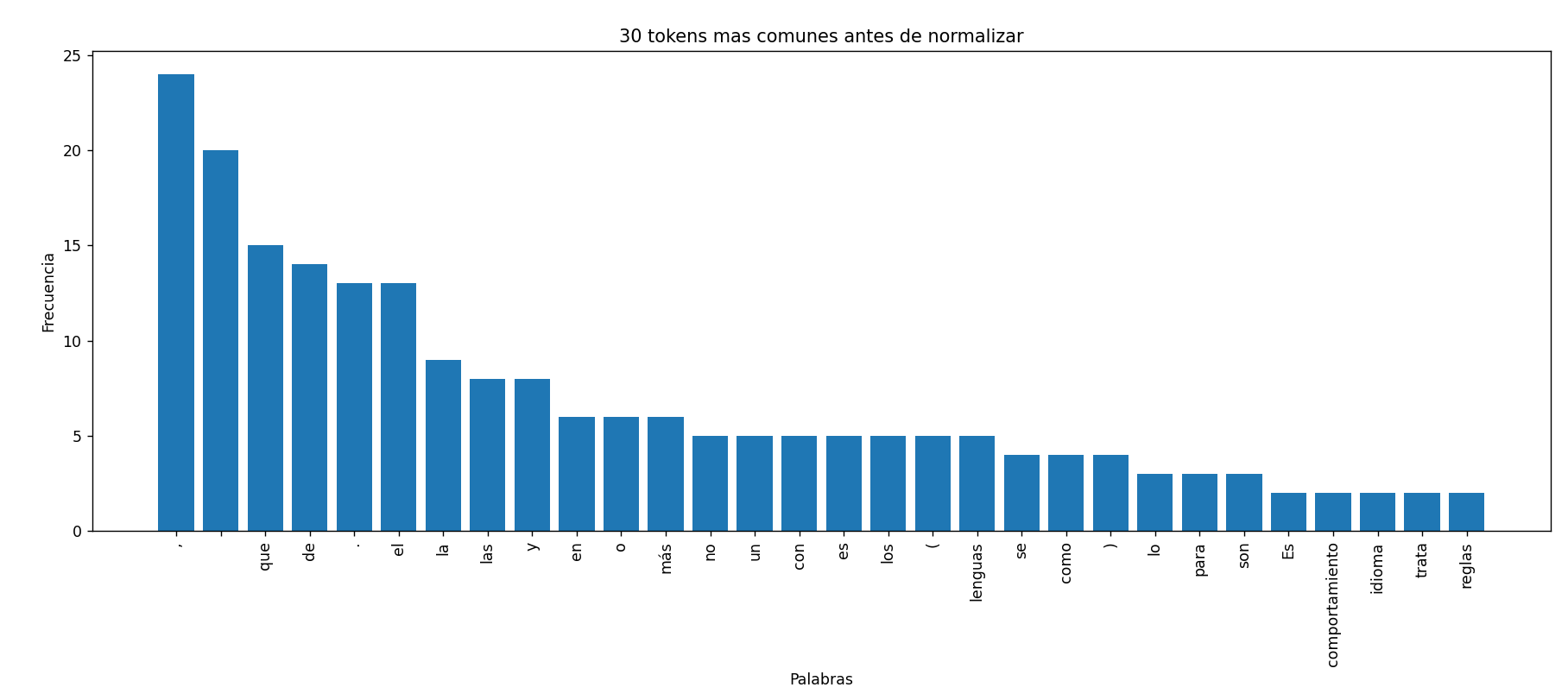
### Histograma de tokens únicos dentro del texto

Por medio de la la librería matplotlib se grafica el histograma de cada token, igual se obtiene la frecuencia de cada uno.



*1.5 Histograma de tokens únicos dentro del texto*

Como se puede observar en la imagen, el histograma obtenido esta en un completo caos, así que solo se graficaran los 30 tokens mas comunes.

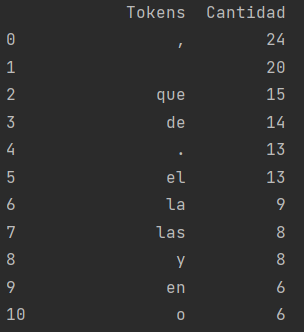


*1.6 Histograma de los 30 tokens mas comunes*

De esta manera se puede apreciar de mejor manera cuales son los tokens mas comunes. En la siguiente sección se hablara en mas detalle sobre los resultados obtenidos en este punto.

### 10 tokens más comunes

A partir de la lista de los 30 tokens mas comunes podemos obtener los 10 mas comunes.

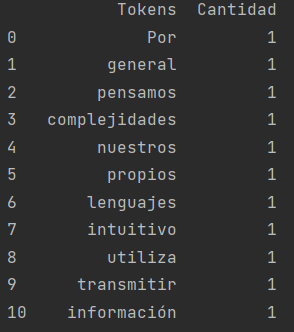


*1.7 Lista de los 10 tokens mas comunes*

Con esta lista podemos observar mas claramente los resultados obtenidos en los histogramas de la sección anterior. El token con mayor frecuencia es una coma, el segundo un carácter que no es visible y los demás son tokens que no nos ofrecen ningún tipo de información relevante.

### 10 tokens menos comunes

Similar al proceso anterior para obtener los tokens mas comunes se realizo algo parecido pero para los menos comunes.



*1.8 Lista de los 10 tokens menos comunes*

Para el caso de los tokens menos comunes no hay mucho que agregar ya que hay demasiados que aparecen una sola vez y la lista se hizo conforme fueron apareciendo los tokens que tenían solo una aparición.

## Normalizacion del texto

El proceso de normalización por el que opte es primero pasar todos los tokens a minúsculas, después se quitan caracteres especiales, signos de puntuación y en el caso del inglés se remueven las abreviaciones, se sigue con el posTagging donde solo nos vamos a quedar con los tokens que contengan la etiqueta de ‘NOUN’ (Sustantivo), ‘VERB’ (Verbo）y ‘ADJ’ (Adjetivo) ya que estas etiquetas son las que nos aportan una mayor información, después se quitan las stopwords, se realiza lematización y finalmente se hace stemming.

### Pasar a minúsculas

El texto se pasa a minúsculas para asegurarnos que todos los tokens sean tratados de la misma manera.

🡺🡺

*2.1 Tokens antes y después de ser pasados a minúsculas*

### Eliminación de caracteres especiales, signos de puntuación y abreviaturas (solo en inglés).

En este proceso de la normalización se eliminan signos especiales, saltos de linea y signos que puntuación que no aportan nada de información a nuestro texto, además para el texto en ingles se realiza la eliminación de abreviaturas, pero eso se verá más adelante.





*2.2 Mismo fragmento de texto antes y después de este proceso*

### Pos-Tagging para mantener tokens con etiquetas de mayor relevancia.

Este proceso se realiza para quedarnos solo con las palabras que tengan etiquetas de una mayor relevancia como lo son los sustantivos, verbos y adjetivos. Este proceso se coloca aquí ya que en el siguiente se realiza la remoción de stopwords, en algunos casos estas stopwords nos pueden proveer contexto para asignarle a cada token una etiqueta correcta.





*2.3 Mismo fragmento de texto antes y después de este proceso*

### Eliminación de stopwords

Como lo dice el titulo en este paso se van a remover las stopwords dentro del texto, las stopwords son palabras que no aportan información relevante.





*2.4 Mismo fragmento de texto antes y después de este proceso*

### Lematización

En este proceso se lematizarán los tokens, esto quiere decir que pondrá cada palabra en su forma base.





*2.5 Mismo fragmento de texto antes y después de este proceso*

### Stemming

Este proceso es similar a la lematización, pero devuelve las palabras a una forma aun mas básica. Para este caso se utilizará el módulo NLTK ya que Spacy no cuenta con esta funcionalidad.





*2.6 Mismo fragmento de texto antes y después de este proceso*

## Exploración de texto a posterior normalización

Se realizara lo mismo que en la sección dos pero ahora con el texto normalizado.

### Numero total de tokens en el texto

Una vez hecha la normalización se regresa en el programa una lista de tokens ya normalizados

Imagen de la pantalla de un computador

Descripción generada automáticamente con confianza baja

*3.1 Lista de tokens normalizados*

**

*3.2 Numero de tokens obtenido mediante len(nombre\_Array)*

### Numero de tokens únicos en el texto

Imagen de la pantalla de un computador

Descripción generada automáticamente con confianza baja

*3.3 Lista de tokens únicos después de normalizar*

**

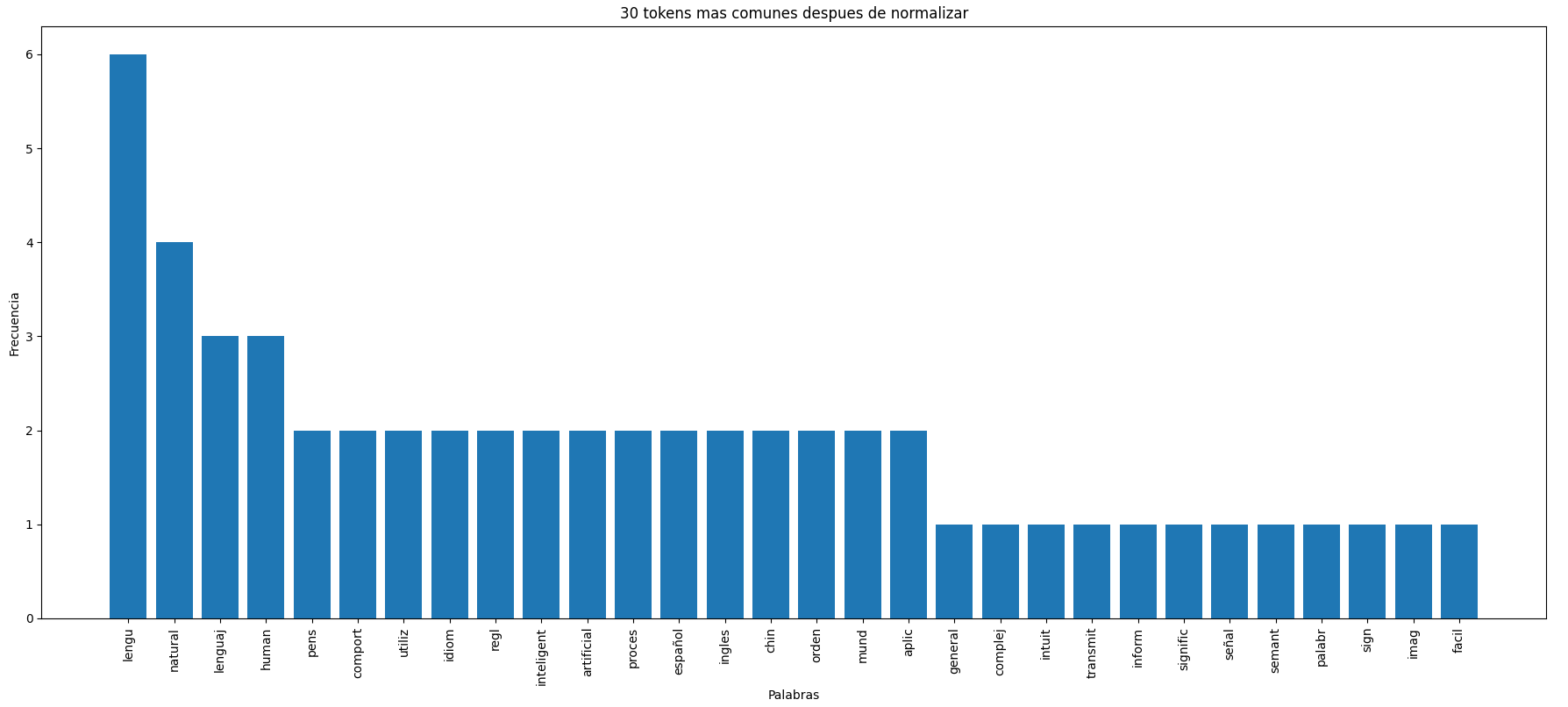
*3.4 Numero de tokens obtenido mediante len(nombre\_Array)*

### Histograma de tokens únicos dentro del texto

Gráfico, Histograma

Descripción generada automáticamente

*3.5 Histograma de los tokens únicos después de normalizar*



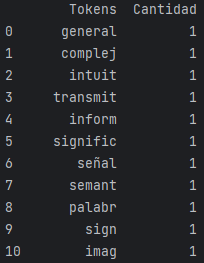
*3.6 30 tokens mas comunes después de normalizar*

### Lista 10 tokens más comunes



*3.7 10 tokens más comunes después de normalizar*

### Lista 10 tokens menos comunes



*3.8 10 tokens menos comunes*

## Análisis y conclusiones

Después de haber realizado la tarea de normalización y haber hecho el análisis exploratorio nos podemos dar cuenta de la gran diferencia que se obtiene al haber hecho la normalización, si comparamos la imagen 1.2 y 3.2 se puede recalcar que la cantidad de tokens que se obtienen es menor, esto nos puede ayudar a procesar de manera más rápida los documentos cuando se tenga una gran cantidad de estos, pero esta no es la única ventaja de la normalización ya que al comparar las imágenes 1.7 y 3.7 se puede observar como en la imagen 1.7 la coma(‘ , ’) ocupa el primer lugar, en cambio en la 3.7 el token “lengu” ocupa el primer lugar, con este simple hecho podemos argumentar que el tema principal del texto es el lenguaje, si intentáramos decir cual es el tema principal cuando el texto no está normalizado esto se convierte en una tarea casi imposible para una máquina. Los pasos que se siguieron dentro del proceso de normalización fueron de gran importancia ya que desde la conversión a minúsculas de cada token para que estos sean tratados de igual manera, la eliminación de signos que no aportan nada al texto, el mantener solo los tokens que tengan etiquetas de mayor relevancia, luego la lematización para que todas las palabras que en un principio pueden parecernos diferentes se puedan unificar como una sola y finalmente el stemming para realizar una unificación aun mayor para estas palabras. Sin todos estos procesos los tokens solo estarían llenos de ruido y se nos haría complicado o no nos sería útil realizar un análisis donde la mayor parte de las palabras no aportan información alguna al texto.

Finalmente en la siguiente sección se realizara lo mismo pero para el anexo B el cual esta en ingles.

# Procesamiento para el texto en ingles

Se realizará el mismo proceso que el texto en español.

## Analisis exploratorio

### Numero total de tokens

Patrón de fondo

Descripción generada automáticamente

*4.1 Lista tokens Anexo B*

**

*4.2 Numero de tokens obtenido mediante len(nombre\_Array)*

### Tokens únicos en el texto

Imagen que contiene Texto

Descripción generada automáticamente

*4.3 Lista tokens únicos anexo B*

**

*4.4 Numero de tokens obtenido mediante len(nombre\_Array)*

### Histograma de tokens únicos

Gráfico, Histograma

Descripción generada automáticamente

*4.5 Histograma de tokens únicos en el Anexo B*

*Gráfico, Gráfico de barras

Descripción generada automáticamente*

*4.6 Histograma de los 30 tokens mas comunes en el Anexo B*

### 10 tokens más comunes

Pantalla de computadora con números

Descripción generada automáticamente con confianza media

*4.7 Lista de los 10 tokens mas comunes en el Anexo B*

### 10 tokens menos comunes

Imagen que contiene Texto

Descripción generada automáticamente

*4.8 Lista de los 10 tokens menos comunes en el Anexo B*

## Normalizacion de texto

Se realizan los mismos pasos que en el Anexo A, solo comparemos los resultados finales con el original. Solo que en este caso si se removerán las abreviaciones que tienen las palabras.