

Modelos probabilísticos: Basados en N-Gramas

Alumno: Marco Medina

Fechas: 21/06

Institución: Instituto Tecnologico Beltran



En esta práctica vamos a usar de las técnicas más utilizadas en el procesamiento del lenguaje natural (PLN), los N-Gramas nos permite estudiar la distribución de secuencias de palabras en un texto, útil para modelado de lenguaje, predicción de palabras y análisis de contenido.

Para desarrollar esta técnica en nuestro fue necesario de 3 cosas:

* Un corpus a analizar
* las BIBLIOTECAS con las que limpiaremos y analizaremos el corpus
* Tener en mente que tipo de N-grama vamos a graficar su frecuencia

El análisis fue implementado en Python utilizando bibliotecas especializadas como NLTK, sklearn, pandas y seaborn. El código se estructuró modularmente a través de funciones.

Para poder usar estas bibliotecas hay que descargarlas en el cuerpo del código y la terminal:

* **En el cuerpo:**

nltk.download("stopwords")

nltk.download("punkt")

nltk.download("wordnet")

nltk.download("averaged\_perceptron\_tagger")

* **En la terminal:**

pip install nltk

pip install pandas

pip install seaborn

pip install matplotlib

**Se aplicaron los siguientes pasos sobre cada línea del corpus:**

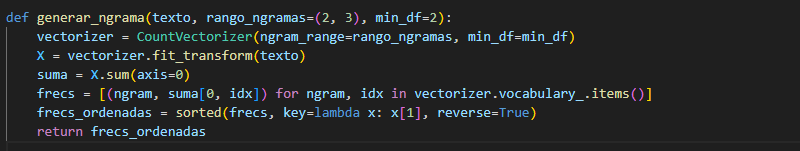
* **Tokenización**: segmentación de las líneas en palabras.
* **Eliminación de stopwords**: se eliminaron palabras vacías en español usando NLTK.
* **Lematización**: se transformaron las palabras a su forma base (por ejemplo, “estudiando” a “estudiar”).



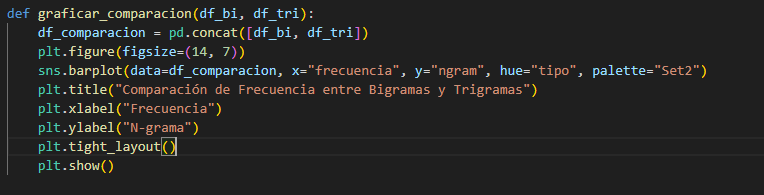
**Luego de preparar el corpus, generamos los N-Gramas. Se utilizó la clase CountVectorizer de sklearn para generar:**

* **Bigramas**: combinaciones de dos palabras contiguas.
* **Trigramas**: combinaciones de tres palabras contiguas.

Para evitar ruido, se definió un parámetro min\_df=2, lo cual indica que solo se consideran las secuencias que aparecen al menos en dos documentos (lineas).



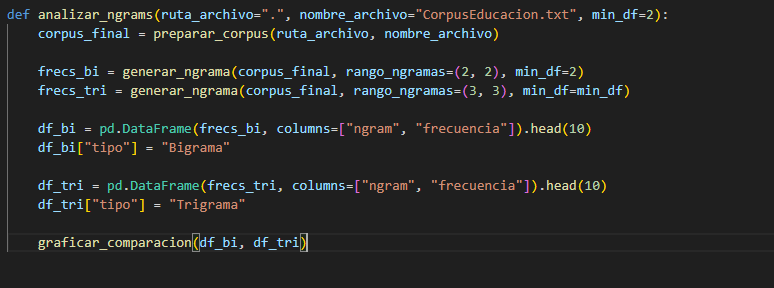
**Y finalmente nos queda realizar la concatenación de los gráficos, en el cual se compara la frecuencia de bi-gramas y tri-gramas:**

****

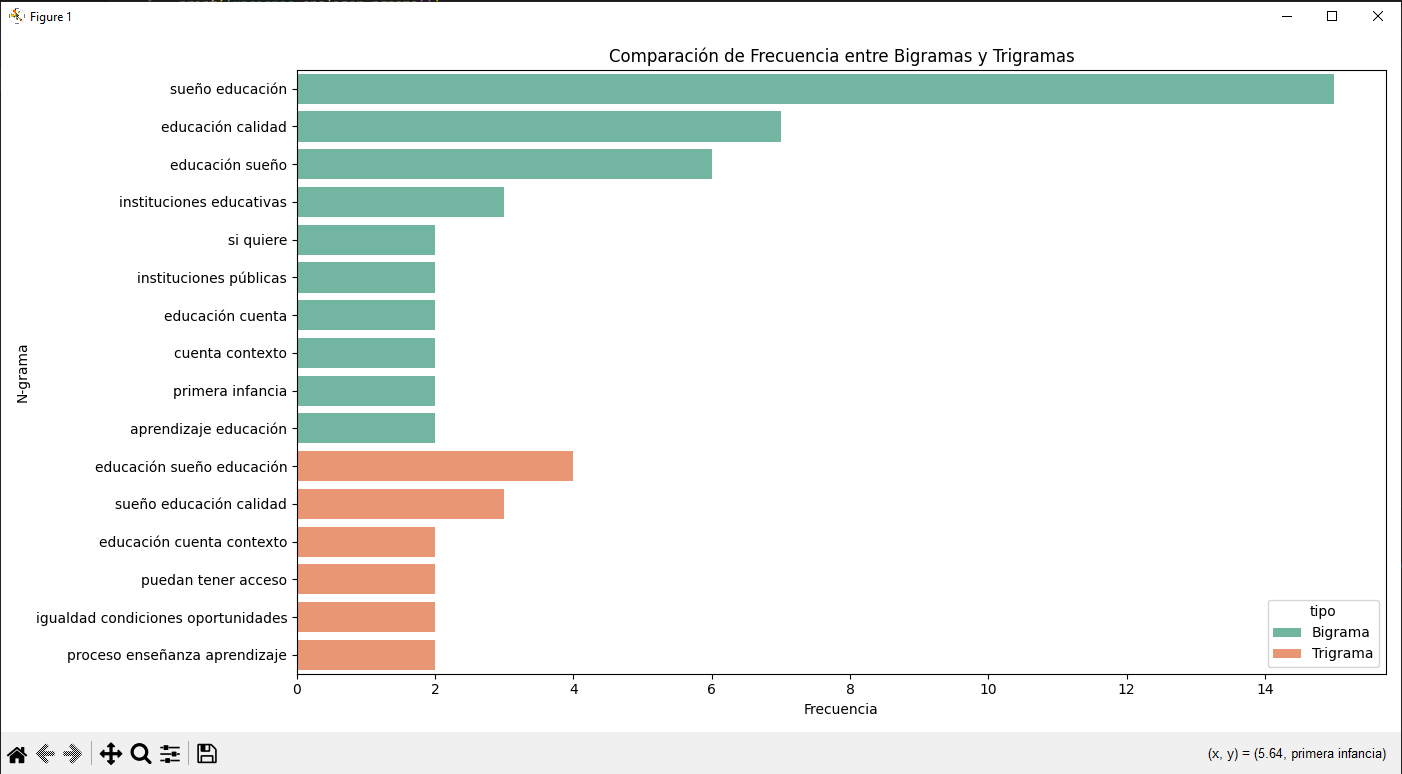
### 

**Resultado:**

### Se llama a la función **analizar\_ngrama:**



* Desde nuestro **main** printeamos la función:



### **Conclusiones**

* Los modelos N-grama permiten descubrir estructuras de lenguaje frecuentes en corpus textuales.
* La limpieza del texto (stopwords, lematización) es esencial para eliminar ruido lingüístico y obtener combinaciones más significativas.
* Este enfoque puede ser útil en sistemas de recomendación, motores de búsqueda, o análisis de sentimientos.