Estudio de Corpus y Analizador de Sintaxis

Autor: Julio García

1. Introducción Teórica

El análisis sintáctico (Syntax Analysis) es una fase fundamental en el Procesamiento del Lenguaje Natural (PLN). Consiste en el análisis de una cadena de símbolos, en lenguaje natural o artificial, conforme a las reglas de una gramática formal. El objetivo es comprender la estructura gramatical de las oraciones, identificando sujetos, verbos, complementos y sus relaciones.

Un Corpus es un conjunto amplio y estructurado de ejemplos reales de uso de la lengua. Estos cuerpos de texto son esenciales para entrenar y probar modelos de PLN. Project Gutenberg es una de las bibliotecas digitales más antiguas, ofreciendo una gran cantidad de libros electrónicos gratuitos libre de derechos de autor, lo que lo convierte en una excelente fuente de corpus tanto en inglés como en español.

2. Herramientas Utilizadas

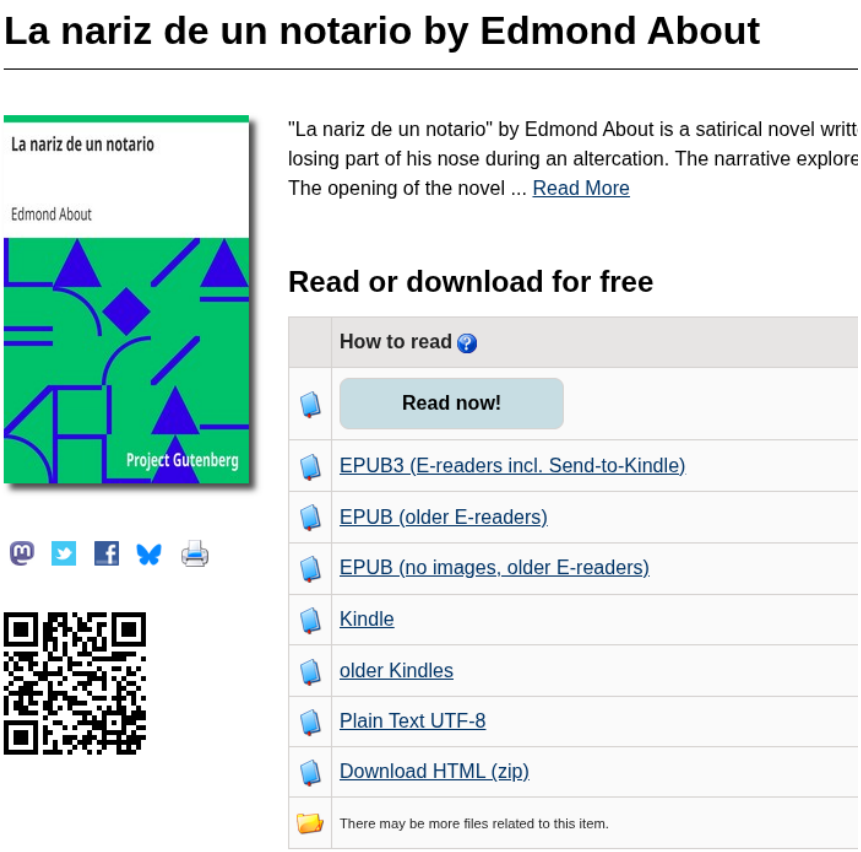
Para el desarrollo de esta práctica se han utilizado las siguientes herramientas y librerías:

- Proyecto Gutemberg: Web donde obtenemos los libros en el formato necesario para utilizar en el proyecto ya que los libros son de dominio público.
- Libreria OS: Librería estándar de Python para gestionar rutas y archivos.
- Libreria RE: Librería estándar de Python para limpiar y procesar texto, eliminando saltos de línea o espacios innecesarios.
- Pyttsx3: Librería mediante la cual podemos transformar un elemento de texto a un archivo de audio de forma local.

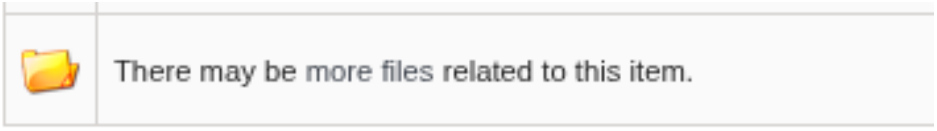
3. Desarrollo Práctico

3.1 Selección y Descarga del Corpus

El primer paso consistió en navegar por el repositorio de Project Gutenberg para seleccionar una obra literaria adecuada. Se eligió 'La Nariz del Notario' como corpus en español.



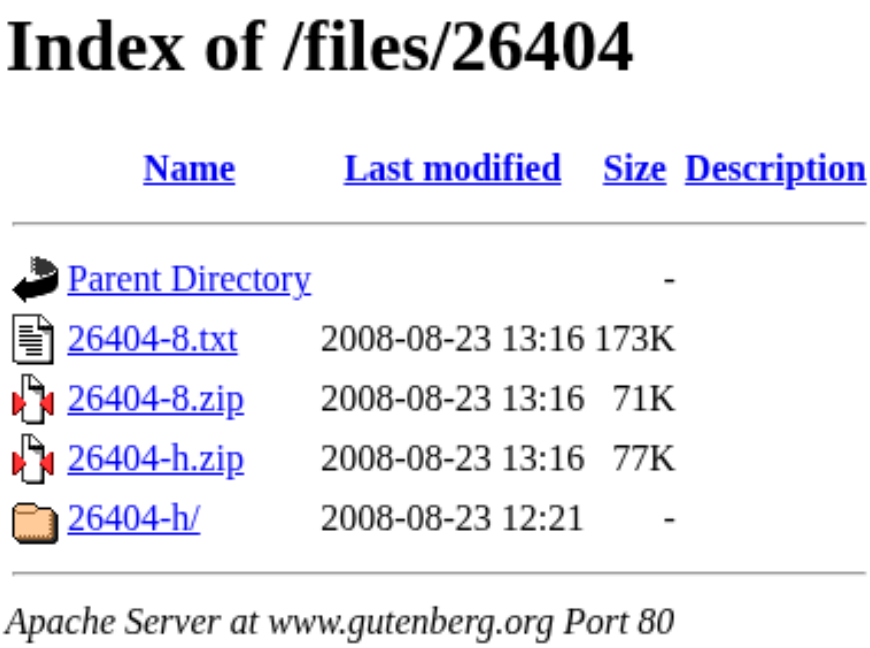
Una vez seleccionada la obra, se procedió a descargar el archivo en formato de texto plano (.txt) para facilitar su procesamiento.



4.2 Procesamiento y Código

Se desarrolló un script en Python ('TextoAudio.py') para automatizar la lectura y conversión del texto. Inicialmente se consideró utilizar la API de Google Translate, pero debido a limitaciones de cuota y tamaño de archivo, se optó por la librería `pyttsx3` que permite una conversión offline eficiente.

El código realiza las siguientes acciones:
1. Busca archivos .txt en el directorio.
2. Lee el contenido y aplica una limpieza básica (eliminación de espacios y saltos de línea redundantes).
3. Detecta el idioma del texto.
4. Utiliza el motor TTS para generar un archivo de audio (.mp3).



Index of /files/26404

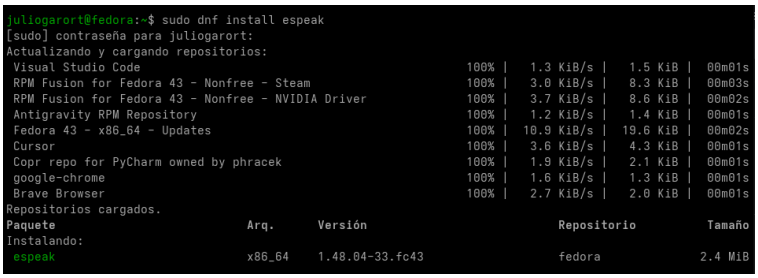
Name	Last modified	Size	Description
Parent Directory		-	
26404-8.txt	2008-08-23 13:16	173K	
26404-8.zip	2008-08-23 13:16	71K	
26404-h.zip	2008-08-23 13:16	77K	
26404-h/	2008-08-23 12:21	-	

Apache Server at www.gutenberg.org Port 80

4.3 Ejecución y Resultados

Tras la implementación, se ejecutó el script exitosamente en un entorno Windows. El resultado final fue la generación del archivo de audio correspondiente al texto procesado.

Al comienzo la primera ejecución se ejecutó en Linux ya que empecé la tarea en mi portátil, pero dio varios errores y después de preguntarle a Chat GPT resulta que la misma librería que uso, en Linux funciona algo diferente por eso los errores por lo que decidí continuar la tarea en Windows y todo funcionaba correctamente.



```
juliogarort@fedora:~$ sudo dnf install espeak
[sudo] contraseña para juliogarort:
Actualizando y cargando repositorios:
 Visual Studio Code                                      100% |   1.3 KiB/s |   1.5 KiB |  00m01s
 RPM Fusion for Fedora 43 - Nonfree - Steam              100% |   3.0 KiB/s |   8.3 KiB |  00m03s
 RPM Fusion for Fedora 43 - Nonfree - NVIDIA Driver      100% |   3.7 KiB/s |   8.6 KiB |  00m02s
 Antigravity RPM Repository                              100% |   1.2 KiB/s |   1.4 KiB |  00m01s
 Fedora 43 - x86_64 - Updates                            100% |  10.9 KiB/s |  19.6 KiB |  00m02s
 Cursor                                                  100% |   3.6 KiB/s |   4.3 KiB |  00m01s
 Copr repo for PyCharm owned by phracek                  100% |   1.9 KiB/s |   2.1 KiB |  00m01s
 google-chrome                                           100% |   1.6 KiB/s |   1.3 KiB |  00m01s
 Brave Browser                                           100% |   2.7 KiB/s |   2.0 KiB |  00m01s
Repositorios cargados.
Paquete                         Arq.      Versión                      Repositorio          Tamaño
Instalando:
 espeak                         x86_64    1.48.04-33.fc43              fedora             2.4 MiB
```

5. Conclusiones

La realización de esta práctica permitió consolidar los conceptos sobre el análisis de corpus y el uso de herramientas de NLP. La disponibilidad de grandes volúmenes de texto en Project Gutenberg facilita enormemente el acceso a datos para entrenamiento y pruebas. También, se verificó la importancia de contar con alternativas offline como `pyttsx3` cuando

las APIs en la nube presentan limitaciones de uso. El flujo de trabajo implementado (Descarga -> Limpieza -> Procesamiento -> Salida) es representativo de las tareas comunes en la ingeniería de datos lingüísticos. El código se podría mejorar aún más y depurar para evitar caracteres especiales o distintos fallos e incluso hacer retoques sobre el narrador (como suene la voz, acento, idioma...)