Ariel Arévalo Alvarado y Jorge Díaz Sagot

Índice de Temas de Salud - Especificación

Objetivo

Elaborar una aplicación que reciba un archivo XML con temas de salud ("*health topics*") obtenidos de Medline Plus y lleve a cabo análisis léxico y sintáctico del mismo, mostrando los datos resultantes al usuario de manera legible y agradable.

Requisitos

- Presentar una GUI amigable y de fácil uso.
- Presentar al menos tres funcionalidades originales de exploración de datos
- Utilizar la mayor cantidad de datos posibles del archivo
- Presentar una pantalla de inicio con un resumen de la información general del conjunto de datos procesado. Esto incluye
    - Cantidad de registros
    - Fecha y hora de los datos
- Presentar al menos un gráfico de datos. Como sugerencia se presenta la cantidad total de referencias de las 10 *information-category* más populares de los sitios
- Proveer enlaces a las páginas que estén en los datos de entrada
- Validar los datos de entrada de manera robusta
- Utilizar *ply* para el análisis léxico y sintáctico
- Identificar durante el análisis léxico los siguientes tipos de tokens
    - Distintas etiquetas XML
    - Hileras de texto
    - Fechas y Horas
    - Direcciones web

Tecnologías y Herramientas

Lenguaje de Programación

Python 3.12.4

Bibliotecas y Dependencias

- **ply**: Para análisis léxico y sintáctico
- **streamlit**: Para la GUI de la aplicación
- **pandas**: Para el manejo de datos
- **matplotlib**: Para graficar los datos

Funcionalidades

1. **Tabla con los datos de cada Health Topic**: Tabla en la cual se muestran los datos relevantes de cada *Health Topic*. También permite la búsqueda y ordenamiento de los datos sobre cualquier columna.
2. **Tabla con los datos de cada Sitio (Secundario):** Tabla en la cual se muestran los datos relevantes de cada uno de los *Site*, los cuales son referencias secundarias proporcionadas para cada tema. También permite la búsqueda y ordenamiento de los datos sobre cualquier columna.
3. **Gráfico de barras con las categorías de información más populares**: Gráfico de barras con conteo de temas y categorías de información.

Estructuras de datos utilizadas

- **dict**: Para representar un *health topic* una vez parseado del archivo XML
- **list**: Para contener los conjuntos de registros en forma de *dict* resultado de parsear los datos completos del archivo XML
- **DataFrame**: Para ordenar los datos en forma de tabla para facilitar las operaciones estadísticas sobre los mismos, así como su visualización

Diseño del Sistema

Estructura del Sistema

Estructura de webapp, con un paquete para la funcionalidad principal (de datos) de la aplicación, y un *app.py* que representa el ciclo de vida de la aplicación en términos de su interfaz al usuario (GUI).

Diseño de la Interfaz de Usuario

Pantalla con información y resultados: Página principal única, donde se van a mostrar la información de la aplicación y datos de resultados del archivo xml cargado.
Panel lateral: Panel que va a funcionar como menú.

- Provee un buscador de archivos locales, donde se sube el archivo XML que se va a procesar.
- Tiene cuatro botones: uno para volver a la página principal, y tres que son la funcionalidad del app de datos(solo se habilitan una vez se cargó el XML).

Navegación: el flujo es ir al menú del panel lateral, subir el archivo XML con temas de salud (*Health Topics*) tomados de *Medline Plus*. Seguidamente se elige la funcionalidad en el mismo menú para procesar los datos. Si se quiere escoger otra funcionalidad, se escoge sin problema en el mismo menú.