Reporte Proyecto final

Yahaira Gomez Sucasaca
Ciencia de la Computación
Universidad Católica San Pablo
Email: yahaira.gomez@ucsp.edu.pe

I. Introducción

La recuperación de datos es un conjunto de técnicas utilizadas para acceder y extraer la información almacenada en diferentes tipos de bases de datos o medios digitales. El objetivo del siguiente trabajo es usar la estructura de datos Suffix tree para la recuperación de datos dado una consulta de tipo texto y calcular las ocurrencias sobre los documentos de otra cadena de texto 'P' junto con una puntuación de similitud por documento. La puntuación puede ser calculada de la siguiente manera:

$$f = TF * log(N/df) \quad (1)$$

- TF = Número de ocurrencias de la palabra en el documento
- N = Número total de documentos.
- df = Número de documentos que contienen la palabra.

II. Reporte de actividades

Se logró realizar la tarea encomendada con éxito haciendo modificaciones en la estructura de datos. A continuación, se muestra un reporte sobre las actividades realizadas:

- Primero, se hizo un preprocesamiento de datos para normalizarlos: Se quitaron signos de puntuación, convertimos todas las palabras en minúsculas se eliminaron los stopwords, también se agregó el símbolo de dólar (el cual indica el límite de cada abstract).
- Luego, se probó el programa con diferentes conjuntos de datos de tamaño 100, 1000, 10000, 100000 y 500000 abstracts.
- La operación resultó exitosa y se calculó el tiempo de inserción de palabras al árbol y la búsqueda de la palabra, los resultados se muestran en la Tabla I.

Sin stopwords		
Palabra usada: Describe		
	Time	Time
Data number	Insertion (ns)	Searching (ns)
100	134260700	52097900
1000	173382500	46864800
10000	2157691000	567126600
100000	23251984600	6486541200
500000	108416000000	20393742100

Tabla I
Tiempos de inserción y búsqueda de palabras eliminando stopwords

- Por último, se comparó el rendimiento de las consultas de un grupo de datos que tomaba cuyo pre-procesamiento eliminaba las stopwords (Tabla I) y otro grupo que las tomaba en cuenta (Tabla II), dando un resultado notoriamente visible.

Con stopwords		
Palabra usada: Describe		
	Time	Time
Data number	Insertion (ns)	Searching (ns)
100	266591100	135975800
1000	442204500	215781000
10000	3006877200	1331890700
100000	46251004900	13877822500

Tabla II
Tiempos de inserción y búsqueda de palabras considerando stopwords

III. Dificultades presentadas

El programa presentó un buen rendimiento en cuanto a tiempo y espacio de memoria. Sin embargo, se presentaron algunos inconvenientes en el camino, los cuales se describen a continuación:

- Se demoró mucho tiempo al momento de pre-procesar datos mayores a 500000 abstracts y mucho más al momento de insertar cada palabra en el árbol.
- Una limitante del programa es que se utiliza un solo árbol para todos los abstracts que se insertan, provocando que este crezca notoriamente en cuanto a amplitud.
  OJO: No hay problema con la altura, ya que la altura de este árbol viene a ser igual al tamaño de la palabra más larga que pueda haber en algún abstract.

IV. Trabajos futuros

Actualmente se llegó a calcular la puntuación de similitud de ocurrencias de la palabra a consultar. Sin embargo, también se podría imprimir la región/es del texto, similar a lo que hace una interfaz de terminal regex/grep o el motor de búsqueda Google.

Referencias

[1] S. Muthukrishnan *Efficient Algorithms for Document Retrieval Problems*, 2007.
[2] M,artin Farach *Optimal Suffix Tree Construction with Large Alphabet*, 1997.
[3] Wikipedia *Trie*, 2021.
[4] GeeksforGeeks *Suffix Tree - Introduction*, 2017.
[5] J. Karkkainen and P. Sanders *Simple Linear Work Suffix Array Construction*, 2003.