

## Proyecto 2

# Recuperación de Documentos de Texto

### 1- Introducción

El logro del estudiante está enfocado a entender y aplicar los algoritmos de búsqueda y recuperación de la información basado en el contenido.

Esta primera parte del proyecto está enfocado a la construcción óptima de un Índice Invertido para tareas de búsqueda y recuperación en documentos de texto.

### 2- **Backend:** Implementación del Índice Invertido

Implementar el índice invertido para recuperación de texto usando el modelo de recuperación por ranking para consultas de texto libre. Considere los siguientes pasos generales:

- Preprocesamiento:
  - o Tokenization
  - o Filtrar Stopwords
  - o Reducción de palabras (*Stemming*)
- Construcción del Índice
  - o Estructurar el índice para guardar los pesos TF-IDF.
  - o Calcular una sola vez la longitud de cada documento (norma) y guardarlo para ser utilizado al momento de aplicar la similitud de coseno.
  - o Manejo del índice en memoria secundaria para soportar grandes colecciones de datos.
    - Blocked Sort-Based Indexing (slide 43-48)
    - Puede ayudarse también de las siguientes lecturas: [referencia 1](#), [referencia 2](#).
- Consulta
  - o La consulta es una frase en lenguaje natural.
  - o El scoring se obtiene aplicando la similitud de coseno sobre el índice invertido en memoria secundaria.
  - o La función de recuperación debe retornar una lista ordenada de documentos que se aproximen a la consulta.

### 3- **Frontend:** Motor de Búsqueda

Para probar el desempeño del índice invertido, se debe construir una aplicación frontend que permita interactuar con las principales operaciones del índice invertido:

- Carga e indexación de documentos en tiempo real.
- Búsqueda textual relacionado a ciertos temas de interés.
- Presentación de resultados de búsqueda de forma amigable e intuitiva.

Se proveerá una colección de aproximadamente 20mil tweets de Twitter (carpeta "clean"). En donde el diccionario de términos puede construirse usando el contenido del atributo "text", y el docID vendría a ser el Id del tweet. En la carpeta también se provee un código para extraer datos de Twitter (tracker.py). [Enlace del repositorio](#)

El grupo tiene la libertad de escoger cualquier otro tópico de interés para realizar la recolección de Tweets.

El grupo puede usar algún otro repositorio de textos ya existente. Ejemplo:

- COVID-19 pandemic:  
<https://www.kaggle.com/harshrey/tweets-covid-sentimentvalues>
- All the News  
<https://www.kaggle.com/snapcrack/all-the-news>

#### 4- Entregable

Los alumnos formaran grupos de máximo de tres integrantes.

El proyecto estará alojado enteramente en GitHub, GitLab o Bitbucket.

Trabajar de forma colaborativa, se considerará para su nota individual.

En el Canvas subir solo el **enlace público** del proyecto.

La fecha límite de entrega es el 04/06/2021 (no habrá prórroga).

#### 5- Informe del proyecto

- Archivo Readme o Wiki en GitHub
- Ortografía y consistencia en los párrafos.
- El informe debe describir todos los aspectos importantes de la implementación.
  - o Construcción del índice invertido
  - o Manejo de memoria secundaria
  - o Ejecución optima de consultas
  - o Incluir imágenes/diagramas de apoyo.
- Además, acompañar al informe un video de presentación del producto, en donde se visualice el programa en acción. La duración del video no debe exceder los 5 minutos. ¡¡Venda su producto!!
- Los resultados deben visualizarse de forma amigable e intuitiva para el usuario.

#### 6- Rúbrica

Va adjunto