

Como funciona el buscador?

Al iniciar el servidor se cargan y leen todos los documentos , esto se hace en la función Load de la clase Moogle , haciendo uso de la clase estática Directory y su método GetFiles el cuál devuelve un array de string con los nombres de los documentos , para acceder al directorio y que este sea válido en cualquier sistema operativo o pc la ruta a la carpeta Content se calcula usando la clase estatica Path y su método GetFullPath . Luego de cargados los documentos , son leídos por separado usando el método ReadAllText de la clase estática File , la cuál devuelve un string con el texto que contiene el documento , para eliminar los caracteres no deseados y separar en palabras independientes utilizo el método procesar_palabras definido por mí , en dicho método utilizando expresiones regulares llevo a cabo tal tarea . Usando una estructura diccionario modelo el TD-IDF , con el método sacar_diccionario definido por mí leo las palabras tokenizadas de cada documento y las almaceno con su respectivo TF , luego de haber construido este diccionario lo almaceno en otro , donde las llaves son los nombres de los documentos y los valores estos diccionarios que modelan cada documento . Luego de esto construyo una lista donde almacenaré todas las palabras de todos los documentos , al que llamaré vocabulario , en el proceso construyo un diccionario donde almaceno por cada palabra , la cantidad de documentos en que aparece , usando todos estos datos creo una matriz donde las filas representan los documentos y las columnas representan las palabras , en dicha matriz la posición ij indica el TF-IDF de la palabra j en el documento i , esto se hace en el método Vectorizar implementado por mí donde recorro cada palabra del vocabulario y asigno los valores de TF almacenado en el documento por cada palabra , al cuál le multiplico el IDF calculado en el método Calcular_Idf implementado por mí . Teniendo esto precalculado estamos listos para la búsqueda . Al usuario insertar un query en el apartado gráfico , este llega como un string al Método Query de la clase Moogle , se procesa este utilizando los métodos mencionados para los documentos , luego de vectorizar el query , se multiplica por la matriz del sistema obteniendo un vector donde la posición i ésima representa el score del documento i respecto al query del usuario , esto se hace utilizando el método CalcularScore luego de ordenar este vector , creo el array de SearchItem que se envía al apartado gráfico modificado por mí para que se vea más colorido y atractivo para el usuario.