Escuela Superior
de Ingeniería y Tecnología
Universidad de La Laguna

Gestión del Conocimiento en las Organizaciones:

Sistema de recomendación.

Modelo basado en el contenido.

Jeff Pérez Frade
(alu0101038520@ull.edu.es)

Índice:

1. Análisis realizado.

En esta ocasión se ha realizado un sistema de recomendación basado en contenido, es decir, que crea una especie de perfil para cada artículo. Posteriormente se crea un perfil de usuario el cual es una lista de los artículos consumidos para encontrar elementos relevantes a comparar entre perfil de usuario y perfil de elementos.

Estas recomendaciones son específicas para cada usuario por lo que el modelo no necesita ningún tipo de dato sobre otros usuarios además de capturar los intereses específicos y recomendar elementos en los que muy pocos usuarios están interesados.

El objetivo de esta aplicación es aprender las preferencias de usuario además de localizar y recomendar ítems que sean similares a las preferencias del usuario.

2. Term Frequency (TF).

TF(x, y) ⇨ Frecuencia de aparición del término **x** en el documento **y**.

Código JavaScript.

```javascript
// Devuelve cuantas veces aparece un termino en un documento
function termFrequency(doc, term){
  let counter = 0;
  for(var i = 0; i < doc.length; i++)
    if(doc[i] == term) counter++;
  return counter;
}
```

3. Inverse Document Frequency (IDF).

Frecuencia inversa calculada como **IDF(x)** = log(N/dfx) donde N es el número de documentos que pueden ser recomendados y **dfx** el número de documentos en **N** donde aparece el término **x**.

Código JavaScript.

```
// Realiza el IDF de un termino analizandolo con todos los documentos
function inverseDocumentFrequecy(matrizVal, term){
  let numDocumentos = matrizVal.length;
  let counter = 0;
  for(var i = 0; i < numDocumentos; i++)
    if(termFound(matrizVal[i], term)) counter++;
  return Math.log10(numDocumentos/counter);
}
```

4. TF - IDF.

Esta es una medida estándar que codifica los documentos en un espacio Euclídeo multi-dimensional. Consiste en la multiplicación de ***TF x IDF***.

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

5. Similitud Coseno.

Esta se utiliza para calcular la similitud entre cada par de documentos, con fórmula:

Código JavaScript.

```
// Funcion para calcular la similitud entre articulos
function cos(u, v){
  let res = 0;
  for(var i = 0; i < u.length; i++){
    for(var j = 0; j < v.length; j++){
      if(u[i].termino == v[j].termino){
        res += u[i].TFNorm * v[j].TFNorm;
      }
    }
  }
  return res;
}
```

Es necesario calcular el TF de todos los atributos de un documento y conformar un vector de atributos para cada uno.

La longitud de estos vectores se calcula como la raíz cuadrada de la suma de los valores al cuadrado.

Código JavaScript.

```
function lengthOfVector(fila){
  let res = 0;
  for(var i = 0; i < fila.length; i++)
    res += Math.pow(fila[i].TF,2);
  return Math.sqrt(res);
}
```

Una vez hecho esto se normalizan los valores, para esto se divide el TF calculado de cada atributo entre la longitud del vector de atributos.

Código JavaScript.

```
// Normaliza los valores de la matriz de terminos para realizar la similitud
function normalizar(obj){
  let objNormalizado = [];
  for(var i = 0; i < obj.length; i++){
    let fila = [];
    for(var j = 0; j < obj[i].length; j++){
      // Divido el TF entre la la longitud del vector
      let TFNorm = obj[i][j].TF / lengthOfVector(obj[i]);
      // Creo un objeto con los datos recogidos
      fila.push({"termino": obj[i][j].termino, "TFNorm": TFNorm});
    }
    objNormalizado.push(fila);
  }
  return objNormalizado;
}
```

Finalmente se suman los productos de los valores en cada par de documentos como se muestra en el siguiente ejemplo:

Código JavaScript.

```
// Funcion para calcular la similitud entre articulos
function cos(u, v){
  let res = 0;
  for(var i = 0; i < u.length; i++){
    for(var j = 0; j < v.length; j++){
      if(u[i].termino == v[j].termino){
        res += u[i].TFNorm * v[j].TFNorm;
      }
    }
  }
  return res;
}
```

6. Eliminación de StopWords.

Este código cuenta con un fichero llamado *stopword.js* el cual cuenta con una especie de base de datos con varias 'stopwords'. Se ha incluido una función que, dado un término por argumento, devuelve un booleano indicando que la palabra se encuentra o no.

Código JavaScript.

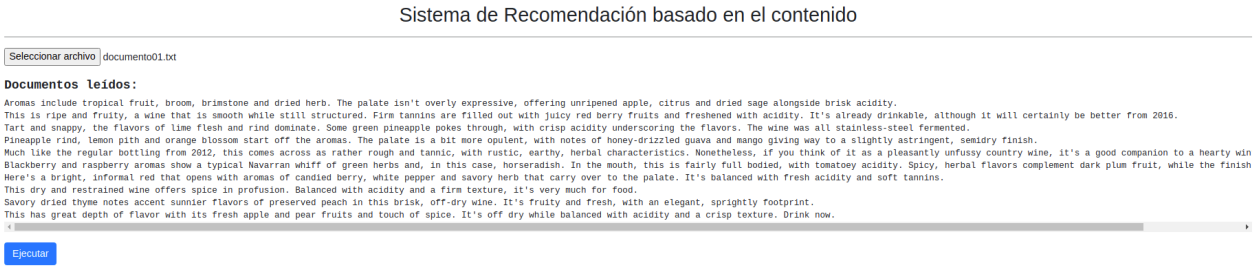
```
function foundStopWord(term){
    let found = false;
    for(var i = 0; i < stopwords.length; i++)
        if(stopwords[i] == term) found = true;
    return found
}
export { foundStopWord };
const stopwords = ['i',
 'me',
 'my',
 'myself',
 'we',
 'our',
 'ours',
 'ourselves',
 'you',
 "you're",
 "you've",
 "you'll",
 "you'd",
 'your',
 'yours',
 'yourself',
 'yourselves',
 'he',
```

La lista continúa pero es omitida por razones de brevedad.

7. Capturas de la aplicación.

La aplicación se conforma en una pantalla principal sencilla, la cual se necesita elegir un archivo texto donde se encuentren los documentos y darle al botón de '*Ejecutar*'.

Prueba del **documento-01.txt** de GitHub



Luego de ejecutar aparecerá una tabla por cada documento con los cálculos correspondientes para cada término.

Documento Nº 1

Término	Indice	TF	IDF	TF-IDF
dried	7	2	0.70	1.40
Aromas	0	1	1.00	1.00
include	1	1	1.00	1.00
tropical	2	1	1.00	1.00
broom	4	1	1.00	1.00
brimstone	5	1	1.00	1.00
overly	12	1	1.00	1.00
expressive	13	1	1.00	1.00
offering	14	1	1.00	1.00
unripened	15	1	1.00	1.00
citrus	17	1	1.00	1.00
sage	20	1	1.00	1.00
alongside	21	1	1.00	1.00
fruit	3	1	0.70	0.70
herb	8	1	0.70	0.70
apple	16	1	0.70	0.70
brisk	22	1	0.70	0.70
The	9	1	0.52	0.52
palate	10	1	0.52	0.52
acidity	23	1	0.15	0.15

Y por último la matriz coseno la cual calcula la similaridad coseno entre cada par de documentos.

Matriz de Similaridad Coseno

Documentos	Doc 1	Doc 2	Doc 3	Doc 4	Doc 5	Doc 6	Doc 7	Doc 8	Doc 9	Doc 10
Doc 1	1.000	0.043	0.091	0.089	0.000	0.079	0.140	0.058	0.147	0.101
Doc 2	0.043	1.000	0.091	0.000	0.042	0.039	0.233	0.173	0.147	0.202
Doc 3	0.091	0.091	1.000	0.093	0.044	0.165	0.049	0.121	0.154	0.106
Doc 4	0.089	0.000	0.093	1.000	0.000	0.081	0.095	0.000	0.050	0.000
Doc 5	0.000	0.042	0.044	0.000	1.000	0.038	0.000	0.055	0.047	0.000
Doc 6	0.079	0.039	0.165	0.081	0.038	1.000	0.127	0.052	0.089	0.092
Doc 7	0.140	0.233	0.049	0.095	0.000	0.127	1.000	0.062	0.105	0.217
Doc 8	0.058	0.173	0.121	0.000	0.055	0.052	0.062	1.000	0.065	0.336
Doc 9	0.147	0.147	0.154	0.050	0.047	0.089	0.105	0.065	1.000	0.114
Doc 10	0.101	0.202	0.106	0.000	0.000	0.092	0.217	0.336	0.114	1.000

8. Conclusiones.

Teniendo en cuenta que hoy en día la inmensa mayoría de las plataformas online cuentan con sofisticados sistemas de recomendación los cuales son especialmente importantes para el marketing, estos se han convertido en sistemas muy complejos y elaborados por lo que ha sido necesario estudiarlos muy detalladamente e intentar practicar recreandolos.