Algoritmos y Estructuras de Datos
Carné: 17584; 17238; 17555
Carrera: Ingeniería en Ciencias de la Computación y TI
Fecha: 30/04/2018
Nombre: Grupo 5

UVG UNIVERSIDAD DEL VALLE DE GUATEMALA

Algoritmos de Recomendación

Parte de la prime etapa de la entrega de proyecto, es la entrega de un documento con investigación de algoritmos de búsqueda o de recomendación. En este documento se recogen algunos que nos parecieron interesantes y que llamaron nuestra atención. Y los cuales creemos que pueden ayudar a resolver el problema planteado.

Hay dos grandes ramas de algoritmos de recomendación, una es la de Filtrado Colaborativo y los Basados en Contenido.

Filtrado Colaborativo

Esta familia de recomendaciones se caracteriza porque se utiliza información de unos usuarios para producir recomendaciones a otros. Así pues, en estos algoritmos los usuarios se benefician de la experiencia de otros usuarios. Este principio general se ha concretado en muy diversas formas, entre las cuales se distingue comúnmente entre los métodos basados en memoria y los basados en modelo. La diferencia radica en que en estos últimos el recomendado genera o aprende una representación propia de los datos (modelo), mientras que en los basados en memoria el algoritmo utiliza los datos en crudo en tiempo de recomendación. En el fondo todos los algoritmos realizan y almacenan parte de los cálculos en una fase online previa a la recomendación, por lo que la distinción entre estas dos vertientes del filtrado colaborativo no es una línea estricta. (Pepa, 2014)

K-vecinos más próximos

La idea básica sobre la que se fundamenta este paradigma es que un nuevo caso se va a clasificar en la clase más frecuente a la que pertenecen sus K vecinos más cercanos. El paradigma se fundamenta por tanto en una idea muy simple e intuitiva, lo que unido a su fácil implementación hace que sea un paradigma clasificatorio muy extendido. (Moujahid, Inza, & Larrañaga, 2014)

		X_1	...	X_j	...	X_n	C
$(\mathbf{x}_1, c_1)$	1	x_{11}	...	x_{1j}	...	x_{1n}	c_1
	⋮	⋮		⋮		⋮	⋮
$(\mathbf{x}_i, c_i)$	i	x_{i1}	...	x_{ij}	...	x_{in}	c_i
	⋮	⋮		⋮		⋮	⋮
$(\mathbf{x}_N, c_N)$	N	x_{N1}	...	x_{Nj}	...	x_{Nn}	c_N
$\mathbf{x}$	$N+1$	$x_{N+1,1}$	...	$x_{N+1,j}$	...	$X_{N+1,n}$	?

Figura 1: Notación para el paradigma K-NN

D indica una entrada de N casos, cada uno de los cuales está caracterizado por n variables predictoras, X1, . . . , Xn y una variable a predecir, la clase C.

El pseudocódigo de cómo funciona es el siguiente:

COMIENZO
  Entrada: $D = \{(\mathbf{x}_1, c_1), \ldots, (\mathbf{x}_N, c_N)\}$
    $\mathbf{x} = (x_1, \ldots, x_n)$ nuevo caso a clasificar
  PARA todo objeto ya clasificado (x_i, c_i)
    calcular $d_i = d(\mathbf{x}_i, \mathbf{x})$
  Ordenar $d_i (i = 1, \ldots, N)$ en orden ascendente
  Quedarnos con los K casos $D_{\mathbf{x}}^K$ ya clasificados más cercanos a $\mathbf{x}$
  Asignar a $\mathbf{x}$ la clase más frecuente en $D_{\mathbf{x}}^K$
FIN

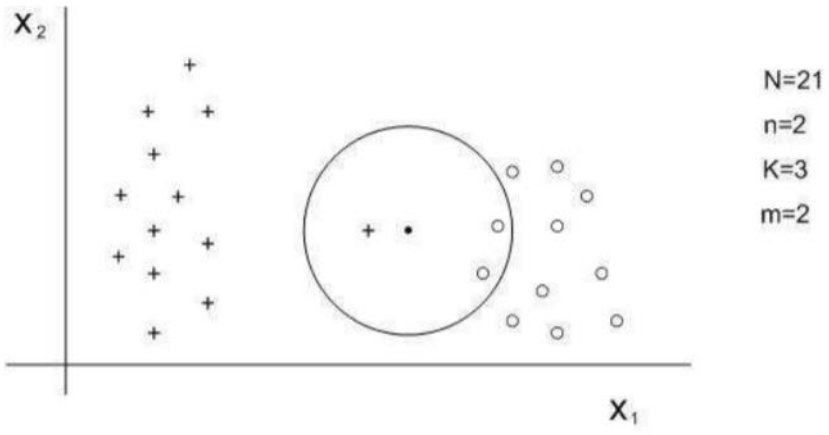

Figura 3: Ejemplo de aplicación del algoritmo K-NN básico

En la Figura 3 tenemos 24 casos ya clasificados en dos posibles valores (m = 2). Las variables predictoras son X1 y X2, y se ha seleccionado K = 3. De los 3 casos ya clasificados que se encuentran más cercanos al nuevo caso a clasificar, x (representado por •), dos de ellos pertenecen a la clase ◦, por tanto, el clasificador 3-NN predice la clase ◦ para el nuevo caso. (Moujahid, Inza, & Larrañaga, 2014)

Variante K-NN con rechazo

La idea subyacente al K-NN con rechazo es que para poder clasificar un caso debo de tener ciertas garantías. Es por ello por lo que puede ocurrir que un caso quede sin clasificar, si no existen ciertas garantías de que la clase a asignar sea la correcta. (Moujahid, Inza, & Larrañaga, 2014)

Dos ejemplos utilizados para llevar a cabo clasificaciones con garantías son los siguientes:

- El número de votos obtenidos por la clase debería superar un umbral prefijado. Si suponemos que trabajamos con K = 10, y m = 2, dicho umbral puede establecerse en 6.
- establecimiento de algún tipo de mayoría absoluta para la clase a asignar. Así, si suponemos que K = 20, m = 4, podemos convenir en que la asignación del nuevo caso a una clase sólo se llevará a cabo en el caso de que la diferencia entre las frecuencias mayor y segunda mayor supere 3.

Variante K-NN con distancia Media

En el K-NN con distancia media la idea es asignar un nuevo caso a la clase cuya distancia media sea menor. Así que en el ejemplo de la Figura 5, a pesar de que 5 de los 7 casos más cercanos al mismo pertenecen a la clase ◦, el nuevo caso se clasifica como +, ya que la distancia media a los dos casos + es menor que la distancia media a los cinco casos. (Moujahid, Inza, & Larrañaga, 2014)

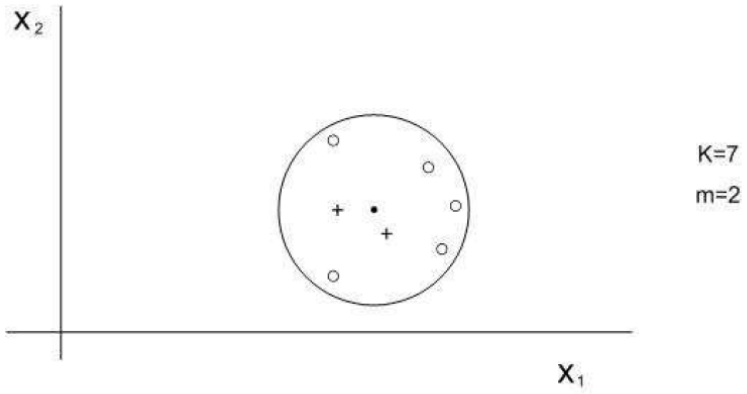

Figura 5: Ejemplo de ilustración del K-NN con distancia media

Descomposición Matricial

(Moujahid, Inza, & Larrañaga, 2014)

La Descomposición en Valores Singulares (en inglés, *Singular Value Decomposition, SVD*) es una técnica de factorización de matrices que permite descomponer una matriz $A \in \Re^{m \times n}$ en otras tres matrices U, S, y V de la siguiente manera:

$$SVD(A) = U \times S \times V^T, \tag{2.14}$$

donde $U \in \Re^{m \times m}$ y $V \in \Re^{n \times n}$ son matrices ortogonales formadas por los vectores singulares de $A \cdot A^T$ y $A^T \cdot A$, respectivamente. $S \in \Re^{m \times n}$ es una matriz formada por los valores singulares de A en su diagonal principal ordenados de mayor a menor.

Reescribiendo el sistema se obtiene

$$U \times S \times V^T = \underbrace{\begin{bmatrix} u_1 & \dots & u_m \end{bmatrix}}_{U \in \Re^{m \times m}} \underbrace{\begin{bmatrix} \lambda_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_r & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix}}_{S \in \Re^{m \times n}} \underbrace{\begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix}}_{V \in \Re^{n \times n}} \tag{2.15}$$

Basados en Contenido

Rocchio

Se basa, como ya se ha anticipado, en el cálculo de centroides para cada usuario, de forma que se obtenga un vector "representante" para cada uno. Estas clases se corresponderán con las características (features) de los ítems, por ejemplo, en Twitter, las palabras clave del contenido de los tweets. De esta forma, se obtiene para cada usuario un centroide que representa su relación con cada característica (término) (Pepa, 2014)

La fórmula para el cálculo de los centroides es la siguiente:

$u[f] = \frac{1}{|\mathbf{u}|} \sum_{i:r(u,i)\neq\emptyset} tfidf(f,i) * r(u,i)$, donde $\mathbf{u} = \{r(u,i) \neq \emptyset | i \in \mathcal{I}\}$

Una vez se dispone de los centroides, el cálculo de la similitud de los usuarios con cada uno de los ítems se realiza mediante cualquiera de los métodos anteriormente descritos. En este caso he seguido la fórmula de similitud mediante coseno:

$$f(u,i) = sim(u,i) = \frac{\sum_f u[f] * tfidf(f,i)}{\sqrt{\sum_f u[f]^2}\ \sqrt{\sum_f tfidf(f,i)^2}}$$

Ítem kNN

La estructura de este algoritmo es idéntica a la de FC del mismo nombre, pero se diferencian en la forma de calcular la similitud entre los ítems. Mientras el de FC utiliza los ratings de otros usuarios, éste utiliza la descripción de los ítems. Por ejemplo, mientras el primero recomendaría películas siguiendo las puntuaciones de los usuarios, el segundo se basaría en, por ejemplo, el género, la sinopsis, el director, y/o el reparto de cada una de ellas. A partir de aquí, las fórmulas se representan prácticamente de la misma forma, pero teniendo en cuenta la diferencia anteriormente descrita. (Pepa, 2014)

$$sim(i,j) = \frac{\sum_f tfidf(f,i) * tfidf(f,j)}{\sqrt{\sum_f tfidf(f,i)^2}\ \sqrt{\sum_f tfidf(f,j)^2}}$$

Bibliografía

Moujahid, A., Inza, I., & Larrañaga, P. (2014). *Clasificadores K-NN.* Recuperado el 30 de Abril de 2018, de http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t9knn.pdf

Pepa, S. M. (1 de Mayo de 2014). *Suite de Algoritmos de recomendacion en aplicaciones reales.* Recuperado el 4 de 30 de 2018, de https://repositorio.uam.es/bitstream/handle/10486/660903/marina_pepa_sofia_tfg.pdf?sequence=1