Ciencias Matemáticas, Vol. 35, No. 2, 2021, Pag. 91-99
Recibido 02-2021, Aceptado 09-2021

Nueva propuesta para el ajuste del rango interno en el agrupamiento de documentos mediante Factorizaciones No Negativas de Matrices
New proposal to adjust the internal rank in the documents clustering by Non Negative Matrix Factorizations

Iosvanny Jesús Alfonso Veloso[1]*, Dra. Marta Lourdes Baguer Díaz-Romañach[2], Dra. Lydia Castro Odio[3].

Resumen Las técnicas de agrupamiento de documentos han recibido mucha atención como herramienta fundamental para la organización eficiente, navegación, recuperación y resumen de grandes volúmenes de textos. Con un método de agrupamiento robusto se pueden organizar los documentos en una jerarquía de grupos que permita la búsqueda y navegación eficiente a través de un corpus, lo cual es un valioso complemento a las deficiencias de las tecnologías tradicionales de recuperación de información. En este trabajo se presenta un *software* desarrollado en MATLAB que incorpora un procedimiento adaptativo para determinar el rango en la Factorización no negativa de la matriz TF-IDF de un corpus. El *software* agrupa los documentos según las temáticas y muestra las palabras más importantes de cada grupo. Para ello se suponen conocidos los conjuntos de palabras por temáticas.

Abstract Documents clustering techniques have received a lot of attention as a fundamental tool for the efficient organization, navigation, retrieval, and summary of large volumes of text. With a robust clustering method, documents can be organized into a hierarchy of groups, allowing efficient search and navigation through a corpus, which is a valuable complement to the shortcomings of traditional information retrieval technologies. In this paper, we present software developed in MATLAB that incorporates an adaptive procedure to determine the range in the non-negative Factorization of the TF-IDF matrix of a corpus. The software groups the documents according to the themes and shows the most important words in each group. For this purpose, the word sets by subject are assumed to be known.

Palabras Clave
Corpus, Factorización, Agrupamiento.

Keywords
Corpora, Factorization, Clustering.

[1] *Facultad de Matemática y Computación, Universidad de La Habana, La Habana, Cuba, iosvanny.alfonso@estudiantes.matcom.uh.cu*
[2] *Facultad de Matemática y Computación, Universidad de La Habana, La Habana, Cuba, mbaguer@matcom.uh.cu*
[3] *Facultad de Artes y Letras, Universidad de La Habana, La Habana, Cuba, lydia@fayl.uh.cu*
***Autor para correspondencia.**

Introducción

Es de interés, dado un conjunto de documentos (denominado *corpus*), poder clasificarlos por temáticas, sin tener que hacer un análisis directo y exhaustivo con cada uno de ellos y apelar a la interpretación para poder llegar a resultados. En la *minería de datos* podemos encontrar una rama denominada *minería de textos*, en la cual se estudian métodos para la extracción de relaciones entre los contenidos de textos en general y, por tanto, poder hacer una clasificación con esa información obtenida.

La minería de textos es un área multidisciplinaria basada en la recuperación de información, minería de datos, aprendizaje automático, estadísticas y Procesamineto del Lenguaje Natural[1] (PLN). Su objetivo fundamental es examinar una colección de documentos no estructurados escritos en lenguaje

[1] En inglés se denomina *Natural Language Processing (NLP)*.

natural y descubrir información no contenida en ningún documento individual de la colección, la detección de tendencias, patrones o similaridades en los textos. Dada una colección de documentos, a menudo surge la necesidad de clasificarlos en grupos basados en la similitud de sus contenidos. Cuando se trata de grandes volúmenes de textos, el proceso de agrupamiento manual sería en extremo agotador y engorroso. El empleo de programas para su automatización reduce considerablemente el tiempo necesario para la realización de la clasificación y el procesamiento de los textos.

Entre las aplicaciones de la minería de textos podemos encontrar el resumen automático de textos, la detección de fraudes, el estudio de tendencias electorales, el análisis de sentimientos y la clasificación de textos; a esta última le prestaremos mayor atención.

La Factorización No Negativa de Matrices ha sido valorada positivamente en una serie de aplicaciones en el área del procesamiento de textos, entre las cuales se destaca el agrupamiento por temas. Es menester entonces expresar el problema en una estructura de datos matemática, por lo que se introduce la Matriz Término-Documento, que contendrá la información, extraída de los textos, necesaria para el trabajo. Luego, con los resultados obtenidos de la factorización matricial se conforman las clases por temas en que se va a clasificar el corpus.

1. Fundamentos Teóricos

Dado un conjunto de textos, deseamos separarlos en clústeres según sus temáticas, para ello llevemos el problema al espacio $\mathbb{R}^{m\times n}$, representando la información esencial del corpus en una una estructura bidimensional que relaciona los documentos y las palabras mediante la *importancia* que tenga una palabra en un documento; esta estructura es llamada matriz Término-Documento, de modo que cada columna de la misma caracteriza un documento. Dicha *importancia* o *peso* puede variar según los criterios de conformación de la matriz, una buena opción para resumir la información del corpus es la Matriz de Frecuencia de Término - Frecuencia Inversa de Documento, o por sus siglas: TF-IDF.

1.1 Matriz TF-IDF

La matriz TF-IDF [2] (Frecuencia de Término - Frecuencia Inversa de Documento) es una matriz cuyos elementos constituyen una medida numérica que expresa cuán relevante es una palabra para un documento en una colección. El valor *tf-idf* (lo denotaremos *tfidf*) aumenta proporcionalmente al número de veces que una palabra aparece en el documento, pero es compensado por la frecuencia de la palabra en la colección de documentos, lo que permite ponderar el hecho de que algunas palabras son generalmente más comunes que otras.

Las ponderaciones $tfidf$ se calculan como el producto de dos medidas, la frecuencia de aparición del término (tf) y la frecuencia inversa del documento (idf). La fórmula para esta métrica es la siguiente:

$$tfidf(t,d,D) = tf(t,d) * idf(t,D)$$

donde t es el término, d denota cada documento, D el espacio total de documentos y $tfidf$ es el peso asignado a ese término en el documento correspondiente. La combinación de los valores de tf e idf brinda una métrica que permite conocer cuán únicas son las palabras de un documento. La ponderación asigna un alto peso a un término si se produce con frecuencia en ese documento, pero rara vez en la colección completa. Sin embargo, si el término ocurre pocas veces en el documento o aparece prácticamente en todos ellos, disminuye el peso asignado por la ponderación $tfidf$ [21].

1.1.1 Tipos de tf

$tf(t,d)$ se define como *frecuencia del término* o *term frecuency*. Existen distintas formas de medir esta frecuencia, entre las que destacan:

1. Recuento (Raw):

   $$tf(t,d) = n_{t,d}$$

   donde $n_{t,d}$ es la cantidad de veces que aparece el término t en el documento d.

2. Forma booleana (Binary):

   $$tf(t,d) = \begin{cases} 1 & \text{si } t \text{ aparece en } d \\ 0 & \text{si no} \end{cases}$$

3. Frecuencia de Término Normalizada:

   $$tf(t,d) = \frac{n_{t,d}}{\sqrt{\sum_{t'\in d} n_{t',d}^2}}$$

4. Frecuencia Logarítmica Escalada (Log):

   $$tf(t,d) = \begin{cases} 1+\log n_{t,d} & \text{si} \quad n_{t,d} > 0 \\ 0 & \text{si} \quad n_{t,d} = 0 \end{cases}$$

1.1.2 Tipos de idf

$idf(t,D)$ se define como *frecuencia inversa de documento* o *inverse document frecuency*. Durante el cálculo de la frecuencia del término se considera que todos los términos tienen igual importancia, no obstante, se conocen casos en los que ciertos términos pueden aparecer muchas veces pero tienen poca importancia. Es necesario mencionar las *stopwords*, que son vocablos no conceptuales o de contenido gramatical que

[2] Las siglas son por su nombre en inglés: Term Frequency - Inverse Document Frequency

son irrelevantes para el estudio que se hace y aumentan la dimensión del problema (en nuestro caso son considerados en esta clase los artículos, preposiciones, conjunciones, entre otros). Esta segunda parte de la fórmula completa el análisis de evaluación de los términos y actúa como corrector de *tf*. Los *idf* más utilizados son los que se presentan a continuación:

1. Unitaria (Unary):

$$idf(t,D) = 1$$

2. Frecuencia Inversa del Documento (Normal):

$$idf(t,D) = \log \frac{N}{1+df(t)}$$

donde

- $N = |D|$: Número de documentos del corpus.
- $df(t)$: Frecuencia de documentos $df(t) = |\{d \in D : t \in d\}|$

3. Frecuencia Inversa del Documento con suavizado:

$$idf(t,D) = \log\left(1 + \frac{N}{df(t)}\right)$$

con $\frac{N}{df(t)} = 0 \Longleftrightarrow df(t) = 0$

4. Frecuencia Inversa Máxima del Documento (Max):

$$idf(t) = \log \frac{\max_{t'} df(t')}{1+df(t)}$$

Estos y otros pesos *tf* e *idf* pueden ser encontrados en [21].

1.2 Factorización No Negativa de Matrices (NMF) y agrupamiento

Dada una matriz $M \in \mathbb{R}^{n\times m}$ con coeficientes $m_{ij} \geq 0$ (la notación m_{ij} indica que es el elemento que se ubica en la fila $i-$ésima y la columna $j-$ésima de la matriz M) y un entero positivo $r << \min(m,n)$, el objetivo es encontrar dos matrices no negativas $W \in \mathbb{R}^{n\times r}$ y $H \in \mathbb{R}^{r\times m}$ tales que:

$$M \approx WH$$

Si cada columna de M representa un objeto (en nuestro caso tal objeto es un documento del corpus), la *NMF* lo aproxima mediante una combinación lineal de r columnas base en W. Esta factorización ha sido utilizada en varias áreas de investigación, tales como la búsqueda de vectores base en imágenes, descubrimiento de patrones moleculares y agrupamiento de documentos, como veremos más adelante.

El modelo clásico para encontrar W y H se basa en minimizar la diferencia entre M y WH debido a que en pocos casos se puede obtener una factorización exacta:

$$\min_{W,\,H} \frac{1}{2}\sum_{i=i}^{n}\sum_{j=1}^{m}(m_{ij} - (wh)_{ij})^2$$

sujeto a $w_{ia} \geq 0,\ h_{bj} \geq 0,\ \forall i,\ a,\ b,\ j\ (a,b = 1,\ldots,r)$.

Estamos en presencia de un problema de Optimización No Lineal con restricciones de no negatividad. Notemos que:

$$\sum_{i=i}^{n}\sum_{j=1}^{m}(m_{ij} - (wh)_{ij})^2 = \|M - WH\|_F^2$$

donde $\|.\|_F$ es la norma de Frobenius [23]. La función objetivo puede considerarse de modo más general como una función de divergencia que mide la diferencia entre M y el producto WH; entre estas funciones podemos encontrar la Divergencia φ de Csiszár, la $\alpha-$Divergencia, la Divergencia de Bregman, $\beta - Divergencia$ [19], la Divergencia Itakura-Saito (IS), la Divergencia Kullback-Leiber (K-L), el Error Mínimo Cuadrático (LSE-Least Square Error), entre otras.

Un elemento de gran importancia es el rango de la factorización, rango interno o dimensión interna, el cual es la dimensión r que corresponde con el número de componentes latentes [20] y determina la reducción de la dimensión del problema. Son llamados "componentes" porque tratan de recomponer la matriz original a través de nuevas bases y se denominan "latentes" porque no emergen hasta que el algoritmo de NMF las construye. Usualmente r se escoge tal que

$$(n+m)r \leq nm.$$

Como $r \leq \min(n,m)$ se puede entender la aplicación de la NMF a la matriz M como una compresión de datos (con pérdidas, desde luego). Valores altos de r pueden tener como resultado matrices dispersas (*sparse*) en la factorización, es decir, que tienen una cantidad considerable de elementos nulos; en muchos casos esto es beneficioso, pues establece fácilmente cuáles son los componentes más importantes para la labor de reconstrucción de la matriz original. En la bibliografía consultada, para mayores valores de r generalmente se obtenían mejores resultados, como en [23], aunque a un mayor costo computacional.

Se enfrentan varias dificultades al tratar de obtener una NMF de una matriz, entre ellas destaca el hecho de que es un problema NP-duro (NP-hard) [2] debido a que todos los elementos de W y H son variables a determinar [18], a diferencia de la factorización sin restricciones que puede ser resuelta eficientemente mediante la descomposición en valores singulares (SVD), por tanto, se aplican algoritmos o variantes de algoritmos de optimización no lineal con restricciones o métodos de descenso donde la función objetivo es convexa en W y en H individualmente, son iterativos y minimizan alternadamente W y H [3]. En el caso general el problema NMF es no convexo y los algoritmos pueden converger a

mínimos locales. Se han reportado varios algoritmos para solucionar la NMF, entre los que destacan los de Actualización Multiplicativa ([19], [16], [3], [1], [11]), Mínimos Cuadrados Alternados (ALS) ([23], [16], [13], [15], [11]), Gradiente Proyectado ([11], [16], [3], [23]), algoritmo de Newton [3], método Quasi-Newton ([3], [16]), Algoritmo de las Proyecciones Sucesivas (SPA) [18], entre otros. Estos métodos han sido probados satisfactoriamente en diversas aplicaciones y la mayoría tiene una complejidad computacional de $O(nmr)$, donde la matriz a factorizar es de $n \times m$ y el rango interno de la factorización es r. Otro problema es la selección correcta de r cuando se realiza la factorización [18], el cual se tratará más adelante.

Como la mayoría de los algoritmos para la solución del problema de la NMF son iterativos se deben inicializar ambas matrices o una de ellas en algunos casos; dicha inicialización es crucial generalmente para la obtención de buenos resultados; una buena inicialización en la NMF mejora la velocidad de convergencia y la exactitud de las soluciones en muchos algoritmos, aunque también puede producir una rápida convergencia a mínimos locales. En algunos se requieren inicializaciones de ambas matrices y en otros solo se inicializa una de ellas y la otra se obtiene de esta como resultado de un paso de algoritmo de forma alternada [10]. Entre las inicializaciones podemos encontrar la *aleatoria*, que consiste en que ambas matrices sean inicializadas con números aleatorios[3] del intervalo $\{x \in \mathbb{R} : \ 0 \leq x \leq 1\}$; esta variante no brinda, en general, una buena estimación inicial para los algoritmos NMF. En artículos especializados se refiere la *inicialización de centroides* [6] construida a partir de la descomposición de centroides [4], la cual constituye una mejor alternativa que la anterior. Otra es la *inicialización de centroides SVD*, la cual inicializa A con la descomposición de centroides a partir del factor que contiene los valores singulares de la descomposición $X = SVD$ [9]. C. Boutsidis y E. Gallopoulos proponen en [12] otra variante de inicialización basada en la descomposición SVD denominada *Nonnegative Double Singular Value Decomposition (NNDSVD)*.

El modelo NMF tiene carácter generativo [7]. A continuación denotaremos m_j a la columna $j-$ésima de la matriz M y m^i a la fila $i-$ésima de la matriz M. Cada columna de la matriz TF-IDF, denotada por M, contiene la codificación de un documento del corpus y cada m_{ij} del vector columna m_j es la importancia del término i con respecto a la semántica de m_j, donde i toma valores en los elementos del vocabulario de D (corpus). Entonces, el problema de la Factorización No Negativa de la matriz M se entiende como: encontrar una aproximación de M de rango r (en este caso dicho rango se elige por el usuario, ya que representará el número de tópicos en los cuales se agruparán los textos) empleando alguna métrica por medio de la factorización de M en el producto de dos matrices de menor dimensión (W y H), donde las filas de W son los indicadores de importancia de cada palabra en los grupos y las filas de H los indicadores de pertenencia de los documentos a los grupos. Cada documento puede ser representado como una combinación lineal de cada tópico:

$$m_j \approx \sum_{k=1}^{r} h_{kj} \cdot w_k$$

Cada fila de M, que es la codificación de cada palabra en el corpus, se puede representar como una combinación lineal de los tópicos también:

$$m^i \approx \sum_{k=1}^{r} w_{ik} \cdot h^k$$

Esta descomposición se puede interpretar de la siguiente forma:

1. w_{ik} indica el grado de pertenencia de la palabra i al grupo k.
2. h_{kj} indica el grado de pertenencia del documento j al grupo k.
3. Si el documento q pertenece al grupo s entonces h_{sq} toma un valor alto, mientras que el resto de elementos de la columna son mucho menores. De igual forma, si la palabra p pertenece al grupo t entonces w_{pt} toma un valor alto, mientras que el resto de elementos de la fila son mucho menores.

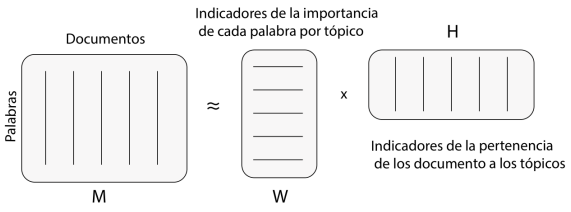

Figura 1. Análisis de la NMF de la TF-IDF

Por tanto, dado un conjunto de documentos, la NMF identifica los tópicos y clasifica los documentos en cada uno de estos: la posición del mayor peso en cada fila de W indica en qué tópico se encuentra la palabra correspondiente y el mayor de cada columna de H dice en cuál se encuentra el documento (ver Figura 1) [23].

En la bibliografía se consideran otros modelos para la factorización que constituyen modificaciones del clásico presentado anteriormente, como son el tri-NMF (NMTF) ([16], [17], [22],[24]) donde se agregan restricciones de ortogonalidad sobre los factores, el Non-Smooth NMF (nsNMF), el NMF Multicapa [16], entre otros. En [24] se comparan algunos de los modelos mencionados.

2. *TextClustersMaker*: *Software* para el agrupamiento

TextClustersMaker fue programado en MATLAB R2018a, utilizando Text Analytics Toolbox, que resulta de gran utili-

[3]Los números generados son realmente pseudoaleatorios debido a la incapacidad de generación de números aleatorios de los ordenadores.

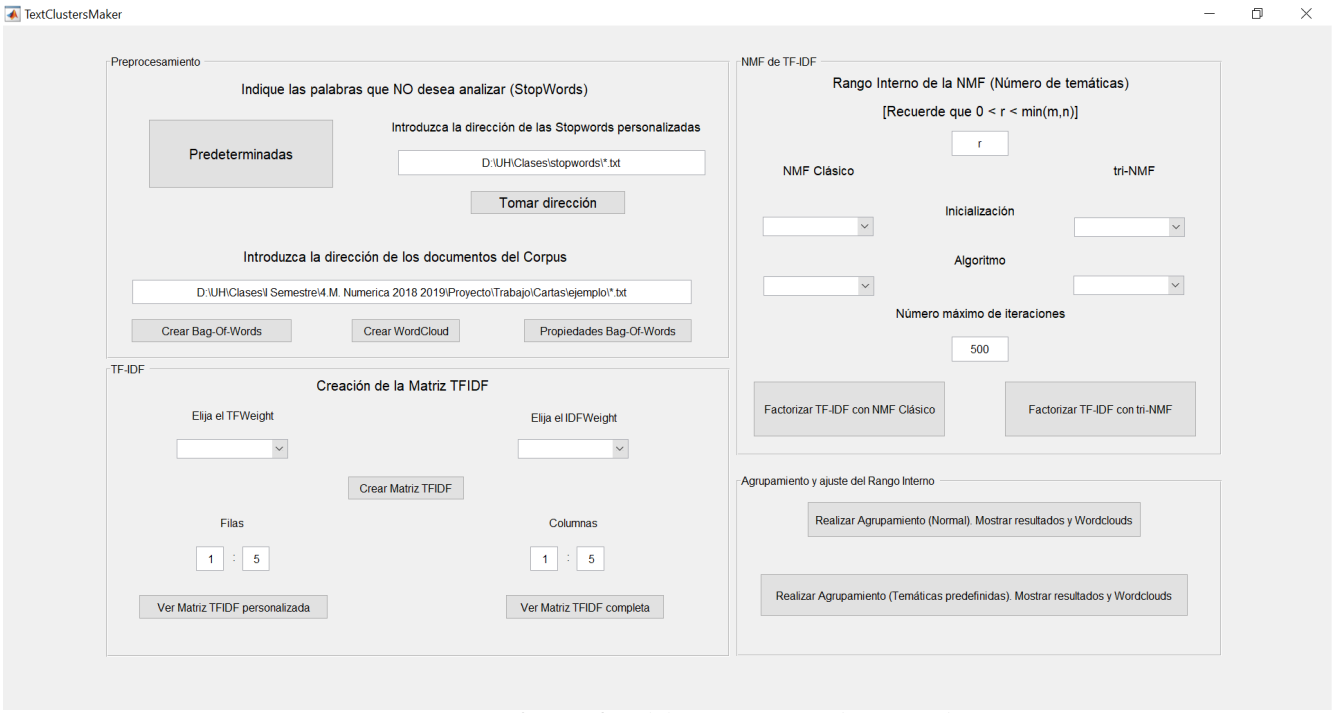

Figura 2. Interfaz Gráfica del *software TextClustersMaker*

dad para el procesamiento de textos. La interfaz de usuario (ver Figura 2) fue concebida con la herramienta GUIDE [14], que se emplea en la creación de estructuras de este tipo. Para la obtención de la matriz TF-IDF se consideró el modelo *Bag-of-Words*, que trata al corpus como una bolsa de palabras que cuenta automáticamente las apariciones de estas en los diferentes documentos que lo conforman; se eliminan palabras que no son trascendentales (*stopwords*) y pueden ser eliminadas las que aparecen poco, las de gran longitud o muy pequeñas, con lo cual disminuyen las dimensiones de la matriz. Es importante precisar que la eliminación anteriormente descrita puede ser personalizada por el usuario.

Utilizando la *Bag-of-Words* se puede extraer la matriz TF-IDF con una función de Text Analytics Toolbox llamada *tfidf*; esta recibe como entrada, además, los pesos de TF e IDF (vistos en la Subsección 1.1).

Se proponen al usuario opciones para la inicialización del algoritmo que eligirá para la factorización. Entre estas inicializaciones se encuentran la aleatoria y la NNDSVD.

La factorización puede ser NMF clásica (dos factores) o tri-NMF. Para la factorización NMF clásica se propone el algoritmo de actualización multiplicativa propio de MATLAB, el *Alternated Constrained Least Squares* (ACL) y el *Alternated Hoyer-Constrained Least Squares* (AHCLS) . Para la factorización tri-NMF se proponen varios algoritmos de actualización multiplicativa [24]. El número máximo de iteraciones a realizar en cada algoritmo está predefinida (son 500), aunque se puede modificar por el usuario. La tolerancia es de $1e-4$ y se considera como criterio de parada que dos iteraciones consecutivas tengan una diferencia en sus respectivos residuos (norma de Frobenius de la matriz residuo) menor que la tolerancia o que alcance el número máximo de iteraciones.

Luego se hace el análisis mencionado en la Subsección 1.2 con el objetivo de obtener el agrupamiento deseado, a través del trabajo con matrices en MATLAB [5]. La función *wordcloud* se encarga de mostrar al usuario una imagen con las palabras más importantes de la colección.

Las principales funcionalidades de este *software* son:

a) Realizar el preprocesamiento de los documentos del corpus (con el mismo formato: TXT, PDF, etc.)

b) La creación de la Matriz Término - Documento que mide los *pesos* de las palabras en el corpus que, en este caso, es la Matriz TF-IDF; se provee al usuario de diversas opciones para poder formar dicha matriz ya que no es única.

c) La Factorización No Negativa de la matriz TF-IDF con un rango interno inicial r_0.

d) El agrupamiento por temáticas de los documentos del corpus y la creación de WordClouds (ver Figura 3), que muestran de una forma gráfica las palabras más importantes en cada grupo (sin colores por temáticas).

e) El agrupamiento por temáticas de los documentos del corpus y la creación de WordClouds (ver Figura 4), que muestran de una forma gráfica las palabras más importantes en cada grupo (con colores por temáticas).

f) La propuesta de rango interno r para la factorización, basado en el método que utiliza en campo semántico (Sección 3).

3. Método de ajuste del rango interno de la NMF

Si el usuario conoce el número de temáticas presentes en el corpus y su objetivo es agrupar los textos en dependencia de dichas temáticas entonces el problema de determinar el rango interno de la factorización queda resuelto debido a que estos dos valores deben coincidir. Más cercano a la realidad es el caso en que el usuario no conoce cuántos temas se ven implicados en el corpus. En este trabajo se propone un procedimiento adaptativo para determinar el rango interno, basado en un rango interno inicial r_0.

Se cuenta con una familia de conjuntos $T_i,\ i = 1,\dots n$, cada uno contiene palabras que pertencen al mismo campo semántico, es decir, están relacionadas con el tema i; a cada uno de los conjuntos anteriores se le asocia una terna RGB[4] que lo caracterizará, la cual está generada de manera pseudo-aleatoria, por lo que tomamos el primer elemento de la terna (R) como representación de cada conjunto. Se desea comprobar si el agrupamiento es correcto. Se verifica si existen temas que estén presentes en clústeres distintos y si en algún clúster, al menos, hay temas difusos; para ello se utilizan los representantes R de cada conjunto.

En cada clúster j de los r_0 creados se hace lo siguiente: Se crea una lista L_j donde se registran los indicadores de pertenencia de cada palabra a cada grupo ($c_{jk},\ k = 1,\dots,|W_j|$, donde $|W_j|$ es la cantidad de palabras almacenadas en el clúster j), es decir, el número R que caracteriza al tema. De esta lista se extrae la moda c_{j_M} y su frecuencia absoluta $f(c_{j_M})$. Puede suceder que la frecuencia relativa de dicha moda, $f_R(c_{j_M})$ no sea alta (por ejemplo: $f_R(C_{j_M}) < 0{,}80$), lo que indicaría que el clúster j debe contener temas difusos, o sea, temas cuyas frecuencias relativas se encuentran en cierto rango y cuyo extremo mayor es menor que 0,80. Una vez calculadas todas las frecuencias relativas $f_R(c_{jk})$ se eligen aquellos c_{jk} que puedan estar en dicha situación (se pueden tomar aquellos que $0{,}40 < f_R(c_{jk}) < 0{,}80$) y se agregan a la lista D; en caso de que esto no suceda se agrega c_{j_M} a la lista A. Se crean luego las listas $H = A \cup D$ y $K = \{x \in H\}$, esta última contiene los elementos de H. Teniendo en cuenta las definiciones de K y H, se cumple la desigualdad

$$|K| \leq |H|.$$

La propuesta para el rango interno está acotada por el cardinal del conjunto que contiene a los temas que tienen importancia en todos los grupos, es decir:

$$1 < r_{new} \leq |K|.$$

Si se obtiene que $|D| = 0$ y $|K| = |H|$ entonces el agrupamiento es correcto. Si son elegidos los parámetros para las frecuencias relativas de la mejor forma y los conjuntos T_i son correctamente confeccionados, entonces

$$r_{new} = |K|.$$

Es importante destacar que este método propuesto depende directamente de la correcta selección de los campos semánticos a tener en cuenta y del preprocesamiento de los textos, principalmente de la correcta eliminación de los vocablos que no aportan al análisis (*stopwords*).

4. Experimentación

4.1 Ejemplo sin la utilización de la propuesta de método de ajuste del rango interno de la NMF.

Para ilustrar el correcto funcionamiento del programa, se estudió un corpus de 6 documentos pequeños, distribuidos de la forma siguiente: los dos primeros tratan sobre Albert Einstein, los dos siguientes sobre España y el último sobre el Papa. Los ubicamos en 3 grupos (los tres mencionados), utilizando *TextClustersMaker*. Primero el programa nos muestra las propiedades del corpus, la cantidad de documentos y de palabras (sin tomar en cuenta las *stopwords*, pues fueron eliminadas en el preprocesamiento), así como una lista con los 10 vocablos más frecuentes; se muestra una WordCloud que da una visión más gráfica de la frecuencia de las palabras en la colección (3). Luego de elegir los pesos tf e ifd (*raw* y *normal*) [5], respectivamente, se crea la matriz TF-IDF. Se asume 3 como rango interno debido a que es la cantidad de grupos que queremos formar, realizamos la factorización no negativa de la matriz TF-IDF y el agrupamiento. Se muestran 3 tablas con los documentos que pertenecen a cada grupo y 3 WordClouds con las palabras más importantes en cada uno, donde el tamaño indica la importancia del vocablo en ese tema respecto a las demás palabras y el color su pertenencia a los temas predefinidos, para poder así evaluar el agrupamiento (Figura 4).

```
bag =     Counts: [6x481 double]
      Vocabulary: [1x481 string]
        NumWords: 481
    NumDocuments: 6
```

```
top = "papa"           10
      "españa"          8
      "familia"         6
      "padre"           6
      "islas"           6
      "iglesia"         6
      "también"         5
      "mar"             5
      "territorio"      5
      "obispo"          5
```

[4] Una terna RGB es un vector de tres componentes donde cada una especifica las intensidades de rojo, verde y azul de cierto color. Las intensidades en MATLAB se toman en el intervalo $[0;1]$.

[5] Se pueden elegir otros pesos.

Figura 3. WordCloud con la información del ejemplo ilustrador

```
Agrupamiento

Grupo 1: 1, 2.
Grupo 2: 3, 4.
Grupo 3: 5, 6.
```

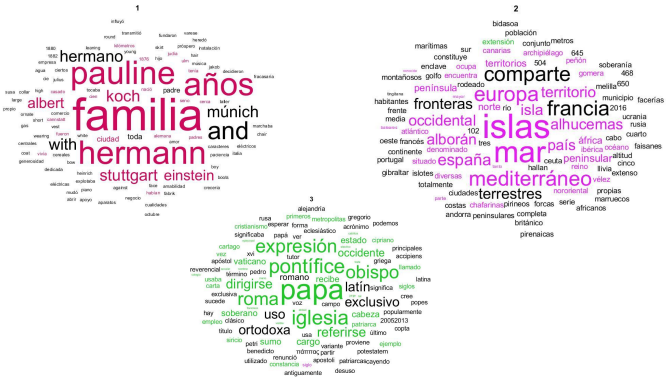

Figura 4. WordCloud con el agrupamiento y el complemento de colores.

Podemos notar que en (4) hay palabras en color negro, estas son aquellas que no fueron clasificadas dentro de ninguno de los conjuntos T_i, lo cual puede provocar errores en la evaluación automática del agrupamiento. También podemos apreciar que en las nubes 2 y 3 hay presencia de varios colores, lo que indica que existen palabras de grupos distintos, aunque sobresalen las del grupo representadas por el color verde.

4.2 Ejemplo con la utilización de la propuesta de método de ajuste del rango interno de la NMF.

Se utilizaron los mismos documentos para analizar las consecuencias de la incorrecta designación del rango interno para la factorización. Se tomaron como los conjuntos T_i $(i = 1,\ 2,\ 3)$ las palabras de los textos que trataban sobre cada uno de los tres temas, de modo que este fuera el *campo semántico* correspondiente. Se hicieron corridas del programa con los rangos $r = 2,\ r = 4,\ r = 3$, en cada caso el programa propuso que el rango interno debía ser $r = 3$. A continuación se presentan los resultados de cada agrupamiento y en las figuras 5, 6 y 7 las WordClouds correspondientes:

- $r = 2$

Figura 5. WordClouds donde se muestra el agrupamiento erróneo con $r = 2$. Una de las WordClouds tiene temas difusos (varios colores predominantes).

```
Agrupamiento
Grupo 1: 3, 4.
Grupo 2: 1, 2, 5, 6.

Propuesta de rango interno: 3
```

- $r = 3$

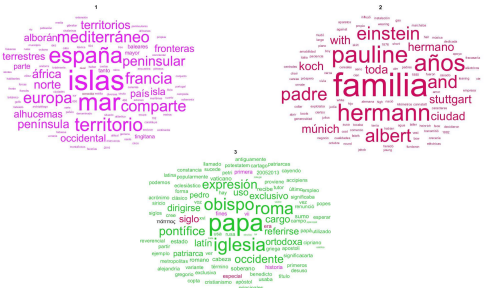

Figura 6. WordClouds donde se muestra el agrupamiento correcto con $r = 3$. Una de las WordClouds tiene algunos colores pero uno de ellos es predominante.

```
Agrupamiento
Grupo 1: 3, 4.
Grupo 2: 1, 2.
Grupo 3: 5, 6.

Propuesta de rango interno: 3
```

- $r = 4$

```
Agrupamiento
Grupo 1: 1, 2.
Grupo 2: 3.
```

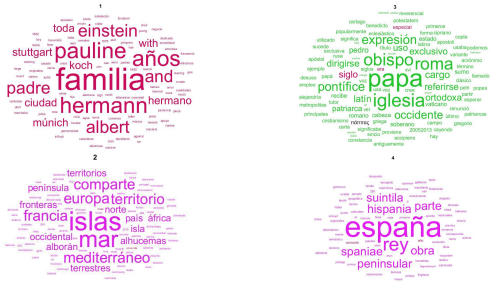

Figura 7. WordClouds donde se muestra el agrupamiento erróneo con $r = 4$. Dos de las WordClouds tienen el mismo tema (predomina el mismo color).

```
Grupo 3: 5, 6.
Grupo 4: 4

Propuesta de rango interno: 3
```

5. Conclusiones

Las factorizaciones no negativas de matrices constituye una herramienta de gran utilidad para el agrupamiento de textos. El programa *TextClustersMaker*, que se apoya en dichas factorizaciones, cuenta con una interfaz gráfica sencilla y que permite el preprocesamiento de documentos para su análisis y la determinación de sus temas subyacentes. Este programa integra varias inicializaciones y algoritmos para hallar la factorización de la matriz término-documentos TF-IDF que contiene la información de la colección a analizar. Se propuso un método adaptativo para determinar el rango interno de la factorización tomando como base el campo semántico. Este *software* puede ser utilizado para estudios sobre la utilización del idioma en distintos contextos sociales, su desarrollo en el tiempo; también para la organización de bibliografía digital. El programa propiciará resultados más acertados en dependencia de la selección de los conjuntos de palabras que definen las temáticas.

En trabajos posteriores se estudiarán otros modelos de factorizaciones matriciales y se compararán los resultados del método propuesto con cada uno de estos.

Referencias

[1] D. D. Lee, S. H. Seung : *Algorithms for non-negative matrix factorization.* Advances in Neural Information Processing Systems 401. 2001.

[2] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein: *Introduction to Algorithms*. The Massachusetts Institute of Technology. 2001.

[3] Z-Y. Zhang: *Nonnegative Matrix Factorization: Models, Algorithms and Applications*. 2001.

[4] I. S. Dhillon: *Concept decompositions for large sparse text data using clustering*. Machine Learning, 42(1/2). 2001.

[5] J. Atencia, R. Nestar: *Aprenda Matlab 6.0 como si estuviera en primero*. Escuela Superior de Ingenieros Industriales, Universidad de Navarra, San Sebastián. 2001.

[6] S. Wild: *Seeding non-negative matrix factorizations with spherical k-means clustering*. Master's thesis, University of Colorado. 2003.

[7] D. Donoho, V. Stodden: *When Does Non-Negative Matrix Factorization Give a Correct Decomposition into Parts?*. 2003.

[8] R. Mitkov: *The Oxford Handbook of Computational Linguistics.* Oxford University Press. 2003.

[9] A. N. Langville: *Experiments with the nonnegative matrix factorization and the reuters10 dataset*. Slides from SAS Meeting. 2005.

[10] A. N. Langville, C. D. Meyer, R. Albright: *Initializations for the Nonnegative Matrix Factorization*. 2006.

[11] C-J. Lin: *Projected Gradient Methods for Non-negative Matrix Factorization*. 2006.

[12] C. Boutsidis, E. Gallopoulos: *SVD based initialization: A head start for nonnegative matrix factorization*. Computer Engineering and Informatics Departament, Patras University. 2007.

[13] H. Kim, H. Park: *Sparse non-negative matrix factorizations via alternatingnon-negativity-constrained least squares for microarraydata analysis*. 2007.

[14] D. O. Barragán: *Manual de Interfaz Gráfica de Usuario en MATLAB (Parte I)*. 2008.

[15] A. N. Langville, C. D. Meyer, R. Albright, J. Cox, D. Duling: *Algorithms, Initializations, and Convergence for the Nonnegative Matrix Factorization*. 2008.

[16] A. Cichocki, R. Zdunek, A. H. Phan, S. I. Amari: *Nonnegative matrix and tensor factorizations*. John Wiley Sons, Ltd. 2009.

[17] J. Yoo, S. Choi: *Orthogonal nonnegative matrix tri-factorization for co-clustering: Multiplicative updates on Stiefel manifolds.* Information Processing and Management 46. Elsevier Ltd. 2010.

[18] N. Gillis: *The Why and How of Nonnegative Matrix Factorization*. 2014.

[19] J. M. Rodríguez y R. Hausdorff: *Selección de β en Factorización de Matrices No Negativas usando la $\beta-$ divergencia.* Tesis de Grado en Licenciatura en Matemáticas Aplicadas. Instituto Tecnológico Autónomo de México. 2014.

[20] M. Á. Pérez: *Técnicas de Factorización No-negativa de Matrices en Sistemas de Recomendación*. Tesis de grado en Ingeniería de las Tecnologías de Telecomunicación, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla. 2017.

[21] M. Calvo: *Text Analytics para Procesado Semántico*. Trabajo Fin de Máster en Técnicas Estadísticas. Universidad de Vigo. 2017.

[22] N. Del Buono, G. Pio: *Non-Negative Matrix Tri-Factorization for co-clustering: an analysis of the block matrix.* Information Sciences. 2017.

[23] R. Díaz: *Análisis Factorial y Factorizaciones no Negativas de Matrices en Lingüística de Corpus*. Tesis de Diploma. Facultad ed Matemática y Computación, Universidad de La Habana. 2018.

[24] I. Alfonso: *Una aplicación de las Factorizaciones no Negativas de Matrices a la Minería de Textos*. Tesis de Diploma. Facultad ed Matemática y Computación, Universidad de La Habana. 2020.

[25] J. Gamboa: *Text Mining: Análisis de sentimientos para la toma de decisiones*. Presentación en VISIÓN, Congreso Internacional de Ingeniería, Ciencias Aeronáuticas y Arquiford. XXI Edición.