Resumen extractivo de texto multidocumento usando un enfoque de optimización multiobjetivo basado en colonia de abejas artificiales

Jesus M. Sanchez-Gomez
Departamento de Matemáticas
Universidad de Extremadura
Cáceres, España
jmsanchezgomez@unex.es

Miguel A. Vega-Rodríguez
Depto. Tecnología de Computadores y Comunicaciones
Universidad de Extremadura
Cáceres, España
mavega@unex.es

Carlos J. Pérez
Departamento de Matemáticas
Universidad de Extremadura
Cáceres, España
carper@unex.es

***Resumen*—Los métodos automáticos de resumen de texto son cada vez más necesarios en la actualidad. Los enfoques de resumen extractivo multidocumento pretenden obtener el contenido principal de una colección de documentos a la vez que reducen la información redundante, lo que puede ser abordado desde un enfoque de optimización multiobjetivo. En este trabajo se ha diseñado e implementado un algoritmo MOABC (*Multi-Objective Artificial Bee Colony*) para esta tarea. Los experimentos se han realizado en base a conjuntos de datos de DUC (*Document Understanding Conferences*), y se han evaluado con las métricas ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*). Los resultados muestran mejoras importantes: del 31,09 % y del 18,63 % para el ROUGE-2 y del 8,43 % y del 6,09 % para el ROUGE-L, con respecto a los mejores resultados de enfoques mono-objetivo y multiobjetivo de la literatura científica. Además, se ha demostrado que los valores ROUGE obtenidos son más estables, con una reducción en la dispersión relativa de entre 620,63 % y 1333,95 %, es decir, entre 6 y 13 veces más robustos.**

***Palabras clave*—Resumen de texto multidocumento, optimización multiobjetivo, colonia de abejas artificiales, cobertura del contenido, reducción de la redundancia.**

I. Introducción

En la actualidad, la cantidad de información en Internet crece de forma exponencial en cualquier tema, y los usuarios desean obtener solo la información más relevante sobre dicho tema lo más rápido posible. Una forma de extraer esta información es mediante herramientas de minería de texto [1], con el fin de poder generar automáticamente un resumen a partir de toda la información del tema en cuestión [2], cubriendo la información más importante y evitando la redundancia de la misma.

Existen varios métodos de resumen automático: dependiendo de si la información se obtiene de uno o de más documentos, el resumen puede ser mono-documento o multidocumento [3]. Un resumen mono-documento reduce la información del documento a una breve representación, mientras que un resumen multidocumento selecciona información de toda la colección de documentos. Además, los métodos también pueden ser abstractivos o extractivos [4]. Los métodos abstractivos pueden construir un resumen con palabras o frases que no existen en el texto original, mientras que los métodos extractivos seleccionan subconjuntos del texto original. El objetivo principal del resumen extractivo multidocumento es representar la información más importante en una versión reducida de los documentos originales, manteniendo los contenidos principales y reduciendo la redundancia. De esta manera, los usuarios pueden obtener las ideas principales del texto de forma rápida.

Como se ha definido, la naturaleza del resumen extractivo multidocumento es multiobjetivo, pero la mayoría de los enfoques en la literatura científica se basan en modelos de optimización mono-objetivo (p. ej. [5]). En estos enfoques solo se optimiza una función objetivo, la cual incluye a su vez varios objetivos que deben ser ponderados. Esta asignación de pesos es subjetiva, lo que influye mucho en la solución final. Hasta la fecha solo se ha propuesto un enfoque de optimización multiobjetivo [6], el cual ha obtenido mejores resultados que los enfoques mono-objetivo. Por lo tanto, el resumen automático de texto multidocumento mediante optimización multiobjetivo es una línea de investigación muy reciente, en la que la hipótesis principal es que al utilizar la optimización multiobjetivo se pueden obtener mejores resultados que con la optimización mono-objetivo.

En este trabajo, el resumen extractivo multidocumento es abordado mediante el algoritmo MOABC (*Multi-Objective Artificial Bee Colony*), el cual maximiza simultáneamente tanto la cobertura del contenido como la reducción de la redundancia de la colección de documentos. Los experimentos se han realizado en base al conjunto de datos de DUC (*Document Understanding Conferences*) y los rendimientos del modelo se han evaluado con las métricas ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*). El enfoque propuesto ha obtenido resultados que mejoran las medias ROUGE de los mejores enfoques de la literatura científica, obteniendo además resultados más estables.

II. Trabajo relacionado

En esta sección se muestra una revisión de las principales técnicas de optimización que se han implementado para el resumen extractivo multidocumento.

En primer lugar, se revisan los trabajos mono-objetivo, donde todos los objetivos son ponderados subjetivamente para obtener una función única. [7] consideró el resumen orientado a consultas de documentos como un problema de optimización global con cuatro objetivos, en el que los experimentos se realizaron con enfoques linealizados y lexicográficos. [8] propuso un método genérico de resumen multidocumento basado en la agrupación de oraciones, resolviéndolo con una modificación del algoritmo de optimización de enjambre de partículas (PSO). [9] diseñó un modelo de resumen de texto no supervisado basado en la programación lineal entera, que resolvió con un algoritmo de ramificación y poda (B&B) y un algoritmo PSO. [10] y [11] propusieron el resumen multidocumento como un problema de la p-mediana modificada, implementando [10] un algoritmo de evolución diferencial (DE) autoadaptativo y [11] un algoritmo de evolución diferencial basado en mutación y cruce autoadaptativos (DESAMC). [12] y [13] abordaron el resumen de documentos como un problema de programación no lineal 0-1, donde la función objetivo se definió como la media heroniana de los criterios, y el problema fue resuelto con el algoritmo PSO en ambos casos. [5] consideró el resumen de documentos como un problema de optimización discreta, proponiendo un algoritmo DE adaptativo para resolverlo. [14] y [15] consideraron la programación booleana cuadrática, donde la función objetivo era una combinación ponderada de los objetivos. En ambos casos se implementó un algoritmo DE binario. [16] propuso el resumen de documentos como un problema de programación entera cuadrática, resolviéndolo con un algoritmo PSO. [17] describió un enfoque basado en optimización para el resumen de documentos, implementando un algoritmo DE mejorado. [18] presentó el resumen de documentos como un problema de optimización lineal y no lineal, utilizando un algoritmo PSO para resolverlo. [19] describió el resumen multidocumento como un problema de optimización binaria, proponiendo un algoritmo de selección elitista intergeneracional, recombinación heterogénea y mutación cataclísmica (CHC). En [20], el resumen de texto se trató como un problema de programación booleana, que se resolvió con un algoritmo DE. Finalmente, [21] consideró el resumen multidocumento basado en el enfoque de proximidad tópica, para el cual se propuso un algoritmo DE autoadaptativo.

La optimización multiobjetivo resuelve los problemas de rendimiento de los modelos mono-objetivo, optimizando cada función objetivo sin ponderaciones. En el único trabajo multiobjetivo encontrado ([6]), se propuso un modelo de resumen extractivo multidocumento basado en la optimización discreta, implementando un algoritmo genético de ordenación no-dominada II (NSGA-II) para la resolución del problema.

Los principales objetivos incluidos en estos trabajos son la cobertura del contenido y la reducción de la redundancia. Sin embargo, también existen otros criterios como la relevancia, la coherencia y la significancia, pero no son tan comunes como los anteriores. Además, todos los trabajos llevaron a cabo la experimentación utilizando los conjuntos de datos de DUC y las métricas de evaluación ROUGE.

III. Definición del problema

En esta sección se define el resumen de texto multidocumento como un problema de optimización. Los métodos más usados en este contexto son los métodos de representación de vectores de términos (palabras), donde cada oración se representa como un vector de términos, y la similitud entre oraciones se compara por pares mediante el uso de algún criterio. El criterio mayormente utilizado es la similitud coseno, como en [5] y [6].

III-A. Similitud coseno

Dado el conjunto $T = \{t_1, t_2, \ldots, t_m\}$, que contiene los m términos distintos de la colección de documentos D. Suponiendo un total de n oraciones, cada oración s_i de D se representa como un vector, $s_i = (w_{i1}, w_{i2}, \ldots, w_{im}), i = 1, 2, \ldots, n$, donde cada elemento es el peso de su término correspondiente. Este peso w_{ik} está asociado con el término t_k en la oración s_i, y puede calcularse usando el esquema *frecuencia del término - frecuencia inversa de la oración* (tf_isf), donde tf mide cuántas veces aparece el término en una oración, e isf mide cuántas oraciones de D contienen el término [22], de la siguiente forma:

$$w_{ik} = tf_{ik} \cdot log(n/n_k), \tag{1}$$

donde tf_{ik} cuenta cuántas veces aparece el término t_k en la oración s_i, y $log(n/n_k)$ es el factor isf, donde n_k denota el número de oraciones que contienen el término t_k.

El contenido principal de D se puede resumir cuantitativamente con la media de los pesos de los m términos en T con un vector medio $o = (o_1, o_2, \ldots, o_m)$, donde cada componente se calcula como sigue:

$$o_k = \frac{1}{n} \sum_{i=1}^{n} w_{ik}, \quad k = 1, 2, \ldots, m. \tag{2}$$

Finalmente, la similitud coseno se calcula a partir de los pesos previamente definidos, midiendo la semejanza entre las oraciones $s_i = (w_{i1}, w_{i2}, \ldots, w_{im})$ y $s_j = (w_{j1}, w_{j2}, \ldots, w_{jm})$ de la siguiente forma:

$$sim(s_i, s_j) = \frac{\sum_{k=1}^{m} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2 \cdot \sum_{k=1}^{m} w_{jk}^2}}, \quad i, j = 1, 2, \ldots, n. \tag{3}$$

III-B. Formulación del problema de optimización

Dado el conjunto $D = \{d_1, d_2, d_3, \ldots, d_N\}$, que contiene N documentos. D también se puede representar como un conjunto que contiene las n oraciones de la colección como $D = \{s_1, s_2, \ldots, s_n\}$. El objetivo es generar un resumen $\bar{D} \subset D$ teniendo en cuenta los siguientes tres aspectos:

- *Longitud*: el resumen generado $\bar{D}$ debe tener una longitud fija L (con cierta tolerancia).
- *Cobertura del contenido*: el tema principal de la colección de documentos D debe ser cubierto en el resumen $\bar{D}$ incluyendo las oraciones apropiadas.

- *Reducción de la redundancia*: el resumen $\bar{D}$ no debe ser redundante, es decir, las oraciones existentes en D que son similares entre sí no deben repetirse en el resumen generado.

Este problema de resumen de texto implica la optimización simultánea de la cobertura del contenido y de la reducción de la redundancia. Sin embargo, estos dos objetivos son contradictorios entre sí, por lo que un enfoque de optimización multiobjetivo es la forma natural de abordar este problema.

Sea $x_i \in \{0,1\}$ una variable de decisión binaria que considera la presencia ($x_i = 1$) o la ausencia ($x_i = 0$) de la oración s_i en el resumen generado $\bar{D}$. De esta forma, la representación de la solución (el vector de decisión) es la siguiente: $x = (x_1, x_2, \ldots, x_n)$.

El primer objetivo a optimizar, $\Phi_{cobertura}(x)$, se refiere al criterio de la cobertura del contenido. Dada cada oración $s_i \in \bar{D}$, la cobertura del contenido se expresa en base a la similitud entre s_i y el conjunto de oraciones de D (representado por o). Por lo tanto, la siguiente función debe ser maximizada:

$$\Phi_{cobertura}(x) = \sum_{i=1}^{n} sim(s_i, o) \cdot x_i. \tag{4}$$

El segundo objetivo a optimizar, $\Phi_{red\_redund}(x)$, se refiere a la redundancia de la información. En este caso, se define una variable de decisión binaria y_{ij} relacionada con la presencia simultánea ($y_{ij} = 1$) o la ausencia ($y_{ij} = 0$) de las oraciones s_i y s_j en el resumen generado $\bar{D}$. Para cada par de oraciones $s_i, s_j \in \bar{D}$ la similitud $sim(s_i, s_j)$ debe ser minimizada. Esto es equivalente a maximizar la reducción de la redundancia, esto es, a maximizar la siguiente función:

$$\Phi_{red\_redund}(x) = \frac{1}{\left(\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} sim(s_i, s_j) \cdot y_{ij}\right) \cdot \sum_{i=1}^{n} x_i}. \tag{5}$$

Finalmente, el problema de optimización multiobjetivo del resumen extractivo multidocumento se formula como:

$$\text{máx } \Phi(x) = \{\Phi_{cobertura}(x), \Phi_{red\_redund}(x)\}, \tag{6}$$

$$\text{sujeto a } L - \varepsilon \leq \sum_{i=1}^{n} l_i \cdot x_i \leq L + \varepsilon, \tag{7}$$

donde l_i es la longitud de la oración s_i y ε es la tolerancia para la restricción de longitud del resumen, definida como:

$$\varepsilon = \max_{i=1,2,\ldots,n} l_i - \min_{i=1,2,\ldots,n} l_i. \tag{8}$$

IV. ABC multiobjetivo

El algoritmo ABC (*Artificial Bee Colony*) es un algoritmo de optimización basado en población que se fundamenta en el comportamiento inteligente de un enjambre o colonia de abejas [23]. El ABC simula el comportamiento de las abejas de la miel para resolver problemas de optimización. Principalmente, existen tres tipos de abejas: las abejas obreras, que mantienen las soluciones actualmente conocidas del problema; las abejas observadoras, que permiten la explotación de las mejores soluciones encontradas hasta ahora; y las abejas exploradoras, que se encargan de la exploración de nuevas soluciones cuando algunas de las soluciones actuales se agotan (cuando no se pueden mejorar más).

Este algoritmo ha sido aplicado con éxito para resolver problemas del mundo real en múltiples contextos, ver p. ej. [24] y [25]. Además, algunos autores han propuesto nuevas variantes con el fin de mejorarlo (p. ej. [26] y [27]).

En esta sección se propone un algoritmo MOABC. Antes de describirlo, se explica el preprocesamiento de los documentos de entrada y los operadores de mutación y reparación.

IV-A. Preprocesamiento de entrada

Antes de la ejecución del algoritmo, los documentos de D deben ser preprocesados siguiendo estos pasos:

1. Segmentación. Las oraciones de D se extraen por separado para marcar el comienzo y el fin de cada una.
2. Tokenización. Las palabras (términos) de cada oración son separadas token a token. Los signos de puntuación, interrogación, exclamación... se eliminan en este paso.
3. Eliminación de las palabras vacías. Las palabras vacías de cada oración son eliminadas. Estas palabras son aquellas que carecen de significado principal, como artículos, preposiciones, conjunciones, etc. La lista de palabras vacías usada está proporcionada en el paquete ROUGE, y contiene 598 palabras [28].
4. *Stemming* del resto de palabras. Las raíces del resto de palabras son extraídas mediante el algoritmo de Porter [29], el cual ha sido ampliamente utilizado, convirtiéndose en el estándar para los métodos de análisis lexicográfico y para la recuperación de información en un amplio rango de lenguajes [30].

IV-B. Operador de mutación

La operación de mutación consiste en agregar o eliminar oraciones en el resumen correspondiente. Este operador se rige por la probabilidad de mutación $p_m \in (0,1)$. Para cada oración en una solución x se genera un número aleatorio $r_i \sim U(0,1)$. Si $r_i \leq p_m$, la oración s_i es candidata para la mutación, y si se cumple la siguiente condición:

$$sim(s_i, o) \geq \frac{1}{n} \sum_{j=1}^{n} sim(s_j, o), \tag{9}$$

s_i se incluye en el resumen. De lo contrario, se elimina. Esta condición verifica si la similitud entre s_i y o es mayor o menor que la media de la similitud de las oraciones de D.

IV-C. Operador de reparación

El operador de reparación comprueba que el resumen generado no viola la restricción de longitud definida en la Ecuación 7. Antes de finalizar cada ciclo, cualquier resumen generado puede violar dicha restricción, la cual es verificada en ambas direcciones. Si el resumen generado tiene una longitud inferior a la restricción menos la tolerancia, el resumen es descartado, ya que el número de resúmenes generados en este caso es muy bajo (se ha verificado experimentalmente que estos

resúmenes se producen 6 veces menos que los resúmenes con una longitud superior a la restricción más la tolerancia). Por otro lado, si el resumen generado tiene una longitud superior a la restricción más la tolerancia, este resumen sí es reparado como se explica a continuación.

Dado el resumen generado D^*, que incumple la restricción de longitud, la operación de reparación elimina de D^* la o las oraciones más redundantes (las que tienen un alto grado de similitud entre ellas). Para esto se considera un umbral de similitud $\delta = 0{,}9$ (una concordancia del 90 %, también usada en [6]). Las siguientes condiciones son comprobadas para cada par de oraciones s_i y s_j:

$$\{s_i, s_j \in D^*\} \wedge \{i \neq j\} \wedge sim(s_i, s_j) \geq \delta \quad i, j = 1, 2, \ldots, n. \tag{10}$$

Si estas condiciones se cumplen, entonces se usa el operador de reparación, eliminando la peor oración. Para ello, se calcula el siguiente valor de $calidad$ para s_i y s_j:

$$calidad_{s_i} = sim(s_i, o) + ((sim(o^{sum}, o) - sim(o^{sum-s_i}, o)) \cdot 10, \tag{11}$$

donde $sim(o^{sum}, o)$ es la similitud entre el centro del resumen generado (incluida la oración s_i) y el centro de la colección de documentos o, y $sim(o^{sum-s_i}, o)$ es la similitud entre el centro del resumen generado (excluyendo en este caso la oración s_i) y el centro de la colección de documentos o. Este segundo término tiene un orden de magnitud mayor (se multiplica por 10), ya que mide la calidad del resumen cuando se elimina la oración s_i. Finalmente, la oración con la $calidad$ más baja es eliminada del resumen, y este operador es aplicado hasta que la restricción de longitud requerida se cumple.

IV-D. Principales pasos del algoritmo

En esta subsección se propone una adaptación del algoritmo ABC para optimización multiobjetivo. Mientras que la optimización mono-objetivo devuelve la mejor solución encontrada, la optimización multiobjetivo devuelve un conjunto de soluciones que son no dominadas entre sí. Una solución es no dominada cuando ninguna de las funciones objetivo se puede mejorar sin que se degrade alguna de las otras [31]. El conjunto de soluciones no dominadas se conoce como conjunto de Pareto, y su representación gráfica como frente de Pareto.

El algoritmo MOABC se resume con el pseudocódigo del Algoritmo 1, que contiene los pasos principales del mismo.

En primer lugar, se inicializa el archivo de almacenamiento *Archivo_NDS*, que contiene las soluciones no dominadas (línea 1). Después, la colonia inicial se genera aleatoriamente (línea 2), es decir, para cada solución (resumen), las oraciones de D se seleccionan de forma aleatoria. Los pasos de las líneas 3 a 11 se repiten durante un número máximo de ciclos $ciclos_{max}$.

En el paso de las abejas obreras (línea 4) se aplica una mutación para mejorar la solución, que será seleccionada solo si domina a la original. En la línea 5 se utilizan dos operadores para determinar cuáles son las mejores soluciones asociadas a las abejas obreras: *rank* y *crowding*. El primero clasifica las soluciones en diferentes frentes de Pareto según

Algoritmo 1 Pseudocódigo del MOABC.

1: $Archivo\_NDS \leftarrow \emptyset$
2: $inicializar\_colonia()$
3: **for** $ciclo = 1$ **to** $ciclos_{max}$ **do**
4: $\quad enviar\_abejas\_obreras()$
5: $\quad rank\_y\_crowding(colonia_{tam})$
6: $\quad calcular\_probabilidades()$
7: $\quad enviar\_abejas\_observadoras()$
8: $\quad enviar\_abejas\_exploradoras()$
9: $\quad rank\_y\_crowding(2 * colonia_{tam})$
10: $\quad exportar\_colonia(Archivo\_NDS)$
11: **end for**

sus dominancias, mientras que el segundo da preferencia a las soluciones que aportan mayor diferenciación (a las más diversas) [32]. Una vez aplicados estos operadores, la selección se basa en las probabilidades calculadas en la línea 6, asignando mayores probabilidades a las mejores soluciones. En la línea 7 se envían las abejas observadoras, cuya tarea es seleccionar su abeja obrera asignada, teniendo en cuenta las probabilidades de selección, para tratar de mejorarla (el operador de mutación también es aplicado en este paso). En este punto del algoritmo, el tamaño de la colonia se duplica ($2 * colonia_{tam}$). Las últimas abejas, las exploradoras (línea 8), se encargan de verificar las soluciones agotadas, que son aquellas que no han mejorado más después de un determinado número de intentos. Estas abejas agotadas son reemplazadas por exploradoras, que son nuevas soluciones generadas aleatoriamente. Para finalizar un ciclo, el tamaño de la colonia se reduce al tamaño original ($colonia_{tam}$), aplicando los operadores *rank* y *crowding* nuevamente (línea 9), y se reparan las soluciones (cuando es necesario) para su almacenamiento en el archivo *Archivo_NDS* (línea 10).

V. Resultados experimentales

V-A. Conjuntos de datos

Los conjuntos de datos utilizados para medir el rendimiento han sido proporcionados por DUC (*Document Understanding Conferences*), siendo un banco de pruebas abierto de referencia para la evaluación de resúmenes automáticos. Las comparaciones se han realizado con los trabajos [5] y [6], donde se han utilizado 10 temas del conjunto de datos DUC2002 (del d061j al d070f) [33].

V-B. Métricas de evaluación

El rendimiento del modelo se ha evaluado mediante el uso de las métricas ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) [34], que es considerada la métrica de evaluación oficial para los resúmenes por DUC. En este caso, las dos variantes de ROUGE usadas son: ROUGE-2 y ROUGE-L. Además de la media (M), se han considerado otras dos estadísticas para la medición de la dispersión: el rango y un coeficiente de variación estadístico. El rango (R) se calcula como $R = ROUGE_{mejor} - ROUGE_{peor}$. Sin embargo, no es adimensional, por lo que también se muestra una modificación

Tabla I
Medias (M), rangos (R) y coeficientes de variación (CV) de los valores ROUGE-2 y ROUGE-L. Los mejores aparecen sombreados.

Tema	ROUGE-2									ROUGE-L								
	DE adaptativo [5]			NSGA-II [6]			Enfoque propuesto			DE adaptativo [5]			NSGA-II [6]			Enfoque propuesto		
	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV
d061j	0,266	0,290	109,02	0,306	0,263	85,95	0,365	0,093	25,43	0,542	0,208	38,38	0,554	0,205	37,00	0,590	0,056	9,46
d062j	0,188	0,275	146,28	0,200	0,422	211,00	0,342	0,023	6,60	0,473	0,239	50,53	0,481	0,306	63,62	0,536	0,017	3,26
d063j	0,245	0,208	84,90	0,275	0,279	101,45	0,272	0,005	1,84	0,493	0,156	31,64	0,528	0,171	32,39	0,509	0,050	9,74
d064j	0,194	0,280	144,33	0,233	0,356	152,79	0,308	0,009	2,83	0,462	0,235	50,87	0,488	0,287	58,81	0,495	0,011	2,19
d065j	0,144	0,209	145,14	0,182	0,208	114,29	0,198	0,026	13,32	0,431	0,141	32,71	0,457	0,174	38,07	0,464	0,057	12,29
d066j	0,201	0,257	127,86	0,181	0,245	135,36	0,290	0,019	6,54	0,455	0,196	43,08	0,441	0,149	33,79	0,519	0,007	1,36
d067f	0,239	0,235	98,33	0,260	0,298	114,62	0,356	0,005	1,39	0,509	0,232	45,58	0,529	0,244	46,12	0,580	0,012	2,15
d068f	0,491	0,384	78,21	0,496	0,281	56,65	0,444	0,084	18,83	0,666	0,226	33,93	0,626	0,226	36,10	0,639	0,071	11,09
d069f	0,184	0,166	90,22	0,232	0,239	103,02	0,240	0,008	3,15	0,454	0,135	29,74	0,476	0,191	40,13	0,554	0,010	1,81
d070f	0,224	0,260	116,07	0,262	0,215	82,03	0,305	0,002	0,76	0,496	0,173	34,88	0,513	0,158	30,80	0,515	0,005	0,90
Media	**0,238**	**0,256**	**114,03**	**0,263**	**0,281**	**115,72**	**0,312**	**0,027**	**8,07**	**0,498**	**0,194**	**39,13**	**0,509**	**0,211**	**41,68**	**0,540**	**0,030**	**5,43**

del coeficiente de variación tradicional de Pearson (CV), que implica la relación entre el rango y la media. Puede expresarse en términos de porcentaje como:

$$CV = \frac{R}{ROUGE_{media}} \cdot 100. \quad (12)$$

V-C. Configuración de los experimentos

Los parámetros del MOABC se han elegido para realizar comparaciones justas con los resultados de los enfoques propuestos en la literatura científica ([5] y [6]). Estos son: tamaño de colonia, $colonia_{tam} = 50$; número de ciclos, $ciclos_{max} = 1000$; probabilidad de mutación, $p_m = 0,1$; y el número de repeticiones/ejecuciones independientes, $reps_{max} = 20$.

V-D. Resultados

Los resultados del MOABC son comparados con [6] (el único enfoque multiobjetivo existente en la literatura científica) y con [5] (el mejor enfoque mono-objetivo). La Tabla I presenta la media, el rango y el CV de los valores de ROUGE-2 y ROUGE-L para cada tema y los valores medios de todos los temas para los tres enfoques comparados.

En los resultados obtenidos en la Tabla I se puede observar que el enfoque propuesto supera a los otros dos, mejorando a [5] en 9 de los 10 temas y a [6] en 8 de 10 para el ROUGE-2, y en 9 de los 10 temas a ambos para el ROUGE-L. Además, los rangos y los CV muestran que los resultados del modelo propuesto son muy robustos. Considerando la media de los diez temas, para el ROUGE-2 el enfoque propuesto produce resultados con un CV medio de 8,07 %, mientras que en los otros dos enfoques son de 114,03 % y 115,72 %. Para el ROUGE-L, mientras que el CV está alrededor del 40 % en los otros enfoques, en el modelo propuesto el valor es solo del 5,43 %. Finalmente, la Tabla II muestra las mejoras en términos promedio del enfoque propuesto con respecto a los otros enfoques.

Los resultados de la Tabla II dan lugar a las siguientes observaciones. En primer lugar, el enfoque propuesto mejora al presentado en [5], donde el algoritmo es un DE adaptativo mono-objetivo. La media del ROUGE-2 es mejorada en un 31,09 %, y la del ROUGE-L en un 8,43 %. En segundo lugar, el enfoque propuesto también supera al presentado en [6]. En este caso, el algoritmo es un NSGA-II multiobjetivo. La media del ROUGE-2 tiene una mejora del 18,63 %, y la del ROUGE-L del 6,09 %. Finalmente, los rangos y CV muestran que los resultados del enfoque propuesto son mucho más robustos. Para el ROUGE-2, el enfoque propuesto mejora la media de los CV en más de un 1300 % (13 veces más robusto), y para el ROUGE-L mejoran en más de un 600 % (6 veces más robusto). Esto significa que las soluciones del MOABC son más estables que las obtenidas en los otros dos enfoques.

Tabla II
Comparación de la media (M), del rango (R) y del coeficiente de variación (CV) entre el enfoque propuesto y los otros enfoques.

Enfoque	Mejora del enfoque propuesto (%)					
	ROUGE-2			ROUGE-L		
	M	R	CV	M	R	CV
DE adaptativo [5]	31,09	848,15	1313,01	8,43	546,67	620,63
NSGA-II [6]	18,63	940,74	1333,95	6,09	603,33	667,59

VI. Conclusiones y trabajo futuro

El problema de resumen multidocumento requiere la optimización de más de una función objetivo, por lo que es necesario aplicar enfoques de optimización multiobjetivo. Por primera vez, un enfoque basado en el algoritmo MOABC ha sido diseñado e implementado para este caso. Los resultados obtenidos no solo han mejorado en los valores ROUGE-2 (31,09 % y 18,63 % mejor) y en ROUGE-L (8,43 % y 6,09 % mejor), sino que también han mostrado una menor dispersión (alrededor de 1300 % y 600 % menos, es decir, alrededor de 13 y 6 veces más robusto) y, por lo tanto, la prueba de que el enfoque propuesto es estadísticamente más sólido que los enfoques comparables publicados.

Como línea de investigación futura, el enfoque se adaptará

para su aplicación en el software NeuroK [1], que es una plataforma de *e-learning* basada en la neurodidáctica [35]. El algoritmo generará resúmenes de los contenidos de los estudiantes (mensajes, comentarios, observaciones o valoraciones), en unidades de aprendizaje o en actividades concretas, lo que será útil para los profesores de los cursos para muchos propósitos, incluida la evaluación automática para las calificaciones.

Otra línea de investigación interesante es el análisis del tiempo de CPU del algoritmo, con vistas a una posible paralelización para mejorar el tiempo de ejecución. Las técnicas basadas en programación paralela con OpenMP podrían ser muy útiles, ya que este enfoque permite explotar el paralelismo existente en las arquitecturas multinúcleo actuales.

Agradecimientos

Esta investigación ha sido apoyada por el Ministerio de Economía y Competitividad (Centro para el Desarrollo Tecnológico Industrial, contrato IDI-20161039; Agencia Estatal de Investigación, proyectos TIN2016-76259-P y MTM2017-86875-C3-2-R), Junta de Extremadura (contrato AA-16-0017-1, y proyectos GR18108 y GR18090), Cátedra ASPgems y Unión Europea (Fondo Europeo de Desarrollo Regional).

Referencias

[1] W. Fan y A. Bifet, "Mining big data: current status, and forecast to the future," ACM SIGKDD Explorations Newsletter, vol. 14, no. 2, 2013, pp. 1–5.

[2] H. Hashimi, A. Hafez y H. Mathkour, "Selection criteria for text mining approaches," Computers in Human Behavior, vol. 51, 2015, pp. 729–733.

[3] D. M. Zajic, B. J. Dorr y J. Lin, "Single-document and multi-document summarization techniques for email threads using sentence compression," Information Processing & Management, vol. 44, no. 4, 2008, pp. 1600–1610.

[4] X. Wan, "An exploration of document impact on graph-based multi-document summarization," En Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2008, pp. 755–762.

[5] R. M. Alguliev, R. M. Aliguliyev y C. A. Mehdiyev, "Sentence selection for generic document summarization using an adaptive differential evolution algorithm," Swarm and Evolutionary Computation, vol. 1, no. 4, 2011, pp. 213–222.

[6] H. H. Saleh, N. J. Kadhim y B. A. Attea, "A Genetic Based Optimization Model for Extractive Multi-Document Text Summarization," Iraqi Journal of Science, vol. 56, no. 2, 2015 pp. 1489–1498.

[7] L. Huang, Y. He, F. Wei y W. Li, "Modeling document summarization as multi-objective optimization," En Intelligent Information Technology and Security Informatics (IITSI), 2010 Third International Symposium, IEEE, 2010, pp. 382–386.

[8] R. M. Aliguliyev, "Clustering Techniques and Discrete Particle Swarm Optimization Algorithm for Multi-Document Summarization," Computational Intelligence, vol. 26, no. 4, 2010, pp. 420–448.

[9] R. M. Alguliev, R. M. Aliguliyev, M. S. Hajirahimova y C. A. Mehdiyev, "MCMR: Maximum coverage and minimum redundant text summarization model," Expert Systems with Applications, vol. 38, no. 12, 2011, pp. 14514–14522.

[10] R. M. Alguliev, R. M. Aliguliyev y C. A. Mehdiyev, "pSum-Sade: a modified p-median problem and self-adaptive differential evolution algorithm for text summarization," Applied Computational Intelligence and Soft Computing, vol. 2011, 2011, pp. 1–13.

[11] R. M. Alguliev, R. M. Aliguliyev y N. R. Isazade, "DESAMC+ DocSum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization," Knowledge-Based Systems, vol. 36, 2012, pp. 21–38.

[12] R. M. Alguliev, R. M. Aliguliyev y C. A. Mehdiyev, "An optimization model and DPSO-EDA for document summarization," International Journal of Information Technology and Computer Science (IJITCS), vol. 3, no. 5, 2011, pp. 59–68.

[13] R. M. Alguliev, R. M. Aliguliyev y N. R. Isazade, "Formulation of document summarization as a 0–1 nonlinear programming problem," Computers & Industrial Engineering, vol. 64, no. 1, 2013, pp. 94–102.

[14] R. M. Alguliev, R. M. Aliguliyev y M. S. Hajirahimova, "Quadratic Boolean programming model and binary differential evolution algorithm for text summarization," Problems of Information Technology, vol. 3, no. 2, 2012, pp. 20–29.

[15] R. M. Alguliev, R. M. Aliguliyev y M. S. Hajirahimova, "GenDocSum+ MCLR: Generic document summarization based on maximum coverage and less redundancy," Expert Systems with Applications, vol. 39, no. 16, 2012, pp. 12460–12473.

[16] R. M. Alguliev, R. M. Aliguliyev y N. R. Isazade, "CDDS: Constraint-driven document summarization models," Expert Systems with Applications, vol. 40, no. 2, 2013, pp. 458–465.

[17] R. M. Alguliev, R. M. Aliguliyev y N. R. Isazade, "Multiple documents summarization based on evolutionary optimization algorithm," Expert Systems with Applications, vol. 40, no. 5, 2013, pp. 1675–1689.

[18] R. M. Alguliev, R. M. Aliguliyev y C. A. Mehdiyev, "An optimization approach to automatic generic document summarization," Computational Intelligence, vol. 29, no. 1, 2013, pp. 129–155.

[19] M. Mendoza, C. Cobos, E. Leon, M. Lozano, F. Rodriguez y E. Herrera-Viedma, "A new memetic algorithm for multi-document summarization based on CHC algorithm and greedy search," En Mexican International Conference on Artificial Intelligence, Springer, 2014, pp. 125–138.

[20] R. M. Alguliev, R. M. Aliguliyev y N. R. Isazade, "An unsupervised approach to generating generic summaries of documents," Applied Soft Computing, vol. 34, 2015, pp. 236–250.

[21] K. Umam, F. W. Putro, G. Q. O. Pratamasunu, A. Z. Arifin y D. Purwitasari, "Coverage, Diversity, and Coherence Optimization for Multi-Document Summarization," Jurnal Ilmu Komputer dan Informasi, vol. 8, no. 1, 2015, pp. 1–10.

[22] G. Salton y C. Buckley, "Term-weighting approaches in automatic text retrieval," Information Processing & Management, vol. 24, no. 5, 1988, pp. 513–523.

[23] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Tech. rep., Technical Report-TR06. Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[24] D. Karaboga y B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," Journal of Global Optimization, vol. 39, no. 3, 2007, pp. 459–471.

[25] D. Karaboga, B. Gorkemli, C. Ozturk y N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," Artificial Intelligence Review, vol. 42, no. 1, 2014, pp. 21–57.

[26] M. S. Kıran y O. Fındık, "A directed artificial bee colony algorithm," Applied Soft Computing, vol. 26, 2015, pp. 454–462.

[27] M. S. Kıran, H. Hakli, M. Gunduz y H. Uguz, "Artificial bee colony algorithm with variable search strategy for continuous optimization," Information Sciences 300, 2015, pp. 140–157.

[28] ROUGE Summary Evaluation Package, http://www.berouge.com/. [Accedido 20-Julio-2017].

[29] Porter stemming algorithm, http://www.tartarus.org/martin/PorterStemmer/. [Accedido 14-Junio-2018].

[30] P. Willett, "The Porter stemming algorithm: then and now," Program, vol. 40, no. 3, 2006, pp. 219–223.

[31] C. C. Coello, C. Dhaenens y L. Jourdan, "Advances in multi-objective nature inspired computing," SCI, vol. 272, Springer, 2010.

[32] K. Deb, A. Pratap, S. Agarwal y T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, 2002, pp. 182–197.

[33] Document Understanding Conference, http://duc.nist.gov. [Accedido 14-Junio-2018].

[34] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," En Text summarization branches out: Proceedings of the ACL-04 workshop, vol. 8, Barcelona, Spain, 2004, pp. 74–81.

[35] F. Calle-Alonso, A. Cuenca-Guevara, D. de la Mata Lara, J. M. Sanchez-Gomez, M. A. Vega-Rodríguez y C. J. Perez Sanchez, "NeuroK: A Collaborative e-Learning Platform based on Pedagogical Principles from Neuroscience," En Proceedings of the 9th International Conference on Computer Supported Education (CSEDU 2017), vol. 1, Science and Technology Publications, 2017, pp. 550–555.

[1] https://neurok.es/