ISSN 1870-4069

Sistema para la generación personalizada de resúmenes a partir de múltiples documentos

Orlando Hernández Hernández, Esaú Villatoro Tello,
Christian Lemaître León

Universidad Autónoma Metropolitana (UAM) Unidad Cuajimalpa,
Departamento de Tecnologías de la Información,
México

orlandoox5@gmail.com,
{evillatoro,clemaitre}@correo.cua.uam.mx

Resumen. En el mundo actual las necesidades de información son cada vez mayores y al mismo tiempo muy diversas, esto último debido a los distintos perfiles de usuarios en la web. Durante la última década el aumento en la generación de contenidos ha crecido tanto que se requiere una gran cantidad de recursos para almacenar y procesar toda esta información. Esta explosión de información obliga a desarrollar herramientas inteligentes que faciliten la búsqueda, organización y recuperación de la información para los distintos usuarios. Dentro de este trabajo se describe el desarrollo de una herramienta para la generación automática de resúmenes de múltiples documentos, los cuales son guiados por la consulta de un usuario. La herramienta desarrollada emplea técnicas de Inteligencia Artificial para identificar los distintos sub-tópicos contenidos en una colección de documentos, con los cuales es posible construir un resumen que satisface las necesidades de información de usuarios específicos.

Palabras clave: resumen automático de múltiples documentos, resumen guiado por consulta, aprendizaje automático, agrupamiento de documentos, similitud de textos.

User's Profile-Aware Multi-Document Summarization System

Abstract. Nowadays the information needs are constantly increasing and at the same time they are very diverse, this last due to the presence of users with different profiles. During the last decade the increase in content generation has grown so much that it requires a great amount of resources for storing and accurately processing all this information. This explosion of information requires the development of intelligent

tools that facilitates searching, organizing and retrieving information for different users. In this paper we describe the development of a tool for the automatic generation of summaries of multiple documents, which are guided by the query given from the user. Developed tool uses Artificial Intelligence techniques to identify the different subtopics contained in a collection of documents, then, the most relevant pieces od information are used for building a summary that satisfies the information needs of specific users.

Keywords: multiple document summarization, query focused summarization, machine learning, clustering, textual similarity.

1. Introducción

Actualmente la generación de contenido textual ha crecido de una manera exponencial, de tal forma que se hace una tarea casi imposible leer toda la información referente a uno o varios temas. La información que actualmente podemos encontrar en la red gracias a los buscadores, nos facilitan en gran medida la tarea de recuperación de información ya que el usuario solo debe de ingresar una consulta. Sin embargo, estos buscadores devuelven una lista muy grande de documentos relacionados a la consulta [1]. Debido a que los resultados de la búsqueda son demasiados, es tarea del usuario realizar la actividad de discriminar entre aquellos documentos que le son relevantes y los que no, pues es altamente probable que muchos de los documentos recuperados presentan información redundante, información desactualizada, o incluso algunos que no contengan información relevante para el usuario.

Herramientas como lo son los sistemas de generación de resúmenes pueden impactar de manera positiva en la problemática planteada. Un resumen es la síntesis concisa y coherente de la información más importante contenida en uno o más documentos, por lo que un sistema generador de resúmenes tiene como objetivo presentar al usuario las ideas principales de los documentos de referencia en un texto pequeño [2, 3]. Tradicionalmente, los sistemas de generación de resúmenes se caracterizan por proponer métodos eficientes para la detección de la información relevante de uno o varios documentos, así como identificar las redundancias de forma que es posible obtener un texto simplificado en el que se plantean las ideas clave del conjunto de documentos. Sin embargo, los sistemas de generación de resúmenes han tenido enfoques muy genéricos, es decir, suelen construir resúmenes similares para distintos usuarios. Ante la diversidad de usuarios y de necesidades de información, es que surgen los sistemas de generación de resúmenes guiados por consulta. El objetivo principal de este tipo de sistemas es construir un resumen que satisfaga las necesidades de información de usuarios específicos [2, 4, 5].

En este sentido, el presente trabajo describe el desarrollo de una herramienta para la generación de resúmenes de múltiples documentos guiados por consulta.

Nuestro software emplea técnicas de Inteligencia Artificial y de Procesamiento de Lenguaje Natural para la identificación de información redundante así como de la diversidad de los tópicos contenidos en una colección de documentos. Una característica relevante del sistema desarrollado es la incorporación de un módulo de recuperación de información, el cual es capaz de descargar información de la web, que posteriormente es procesada para la construcción del resumen. Esta funcionalidad permite al usuario obtener en tiempo real todos aquellos documentos que son relevantes a una necesidad de información, es decir, documentos que están relacionados temáticamente. Una vez descargados los documentos, el usuario podrá realizar uno o varios resúmenes que responden a necesidades específicas información. Es importante mencionar la herramienta desarrollada tiene un modo de funcionamiento 'fuera-de-línea', con el cual el usuario puede proporcionar a la herramienta directamente los documentos que quiere analizar. Como se mostrará más adelante, el software desarrollado está listo para descargarse y ejecutarse en plataformas Windows y Linux.

El resto del documento está organizado de la siguiente manera. La sección 2 describe brevemente algunos de los trabajos relacionados, la sección 3 describe el método de generación de resúmenes de nuestro software; la sección 4 muestra algunas pantallas del sistema así como un ejemplo de cómo el sistema toma en cuenta las preferencias del usuario para su funcionamiento. Finalmente la sección 5 enuncia las conclusiones alcanzadas así como algunas líneas de trabajo futuro para la optimización del sistema desarrollado.

2. Trabajo relacionado

La generación de resúmenes es una tarea que el PLN aborda por medio de distintos enfoques y métodos con la finalidad de mejorar la calidad de los resúmenes generados para satisfacer las necesidades de información del usuario. A continuación se describirán algunos trabajos precursores al nuestro.

En el trabajo descrito en [7], los autores emplean a los términos más frecuentes como los más relevantes en el texto, y por lo tanto son los que ayudan a determinar la temática principal de un texto. En [7], el autor solo extrae para la generación del resumen aquellas oraciones que contienen presencia de estos términos. Por otro lado, trabajos que han empleado técnicas de aprendizaje automático para la construcción de resúmenes son como el descrito en [5]. En este enfoque se utilizan atributos que describen a las oraciones en términos de su ubicación en el documento, número de palabras, etc. Con estos atributos los autores entrenan un modelo de aprendizaje computacional, con el cual logran generar resúmenes de un sólo documento. Estos dos trabajos, representan referentes clásicos de la generación de resúmenes de un sólo documento. Y muestran que, hasta cierto punto, atributos relacionados a la posición de las palabras, su frecuencia, su complejidad, similitud con el título, longitud de las oraciones, etc., son atributos que ayudan a identificar porciones de texto importantes para la construcción del resumen. Sin embargo, su funcionalidad no incorpora al usuario

en el proceso de la construcción, es decir, los resúmenes que se generan para un mismo documento serán similares sin importar el perfil del usuario.

Más recientemente, con el afán de incorporar las necesidades del usuario en la generación de los resúmenes, surgen propuestas que incorporan el perfil del usuario. Por ejemplo en [9], los autores proponen un método de regresión para la clasificación de las oraciones que podrían ser parte (o no) del resumen tomando como referencia una consulta dada por el usuario. Se utilizan siete características en total para la generación de resúmenes de múltiples documentos, tres son dependientes de la consulta: coincidencia nombre y entidad, la similitud de palabras y la coincidencia semántica. Las otras 4 son independientes de la consulta: la posición de la oración, entidad nombrada, el resultado de TF-IDF en las palabras y la penalización de palabras vacías[1]. Su sistema propuesto se necesita entrenar con una serie de resúmenes generados por humanos. Este método utiliza varios técnicas basadas en n-gramas[2] las cuales se encargan de calcular el puntaje de las oraciones para identificar las que son relevantes y las que no. Finalmente para eliminar la redundancia de información se utilizan MMR (Maximal Marginal Relevance), la cual es una medida para cuantificar desemejanza entre la oración que se está considerando y las que ya están seleccionadas.

De forma similar, en [11] se utiliza un modelo de aprendizaje supervisado, utilizando el corpus del DUC 2005 para entrenar al sistema. En general, el sistema propuesto consta de tres pasos para lograr la generación de resúmenes. El primero es la clasificación de oraciones en orden de relevancia de acuerdo a la consulta dada, en este paso se utilizaron dos algoritmos de clasificación: *Support Vector Regresion* (SVR) [10] y *LambdaMART* [3]. En este paso se eliminan palabras vacías así como un conjunto de palabras especificas, las cuales son: *"discuss, describe, specify, explain, identify, include, involve, note"*. El segundo es un método de compresión de oraciones, y finalmente el tercer paso es el Post-procesamiento para la generación del resumen, para lo cual utilizan esquemas de pesado *TF-IDF*[3] para la identificación de los elementos relevantes. La desventaja principal de estos trabajos es que requieren de colecciones de datos etiquetados para poder aplicar técnicas de aprendizaje supervisado.

Por otro lado, en [4] se propone un método que no requiere de un conjunto de datos etiquetados. Los autores proponen un método que se compone de cuatro etapas. La primera es un proceso que busca identificar la similitud de los elementos de los documentos, para lo cual crea una matriz de similitudes, misma que involucra a la consulta. El siguiente paso es la ponderación, para esto se suma todas las filas correspondientes a cada documento, de esta forma se logra identificar (rankear) a los documentos más representativos. El tercer paso es ordenar los documentos en forma descendente y eliminar los que tengan

[1] Del ingles *stop words* son términos que no tienen carga semántica como los conectores, artículos, etc.

[2] Es una subsecuencia de n elementos de una secuencia dada

[3] Del inglés *Term Frequency–Inverse Document Frequency*, este esquema le da mayor relevancia a los términos que son menos frecuentes en la colección, pero más frecuentes en el documento

menor importancia. Finalmente se generará el resumen con los documentos más similares a la consulta.

Como se puede observar, la variedad de heurísticas utilizadas para la construcción de métodos de generación de resúmenes es muy variada. Sin embargo, algo que tienen en común estos trabajos es el uso de técnicas de PLN para identificar aquellas porciones de información que se asemejan a la necesidad de información del usuario. Inspirados por estos trabajos, nuestro sistema desarrollado emplea técnicas no supervisadas de aprendizaje, aspectos que le proporcionan una flexibilidad importante a nuestro sistema al no depender de un conjunto cerrado de etiquetas. Finalmente, es importante mencionar que para el desarrollo de nuestra herramienta de generación de resúmenes, tomamos como base las ideas propuestas en [8].

3. Método propuesto

La figura 1 muestra de manera esquemática los componentes principales del sistema de generación de resúmenes. En términos generales el sistema esta conformado por dos módulos: el sistema de Recuperación de Información, y el módulo de Generación del Resumen.

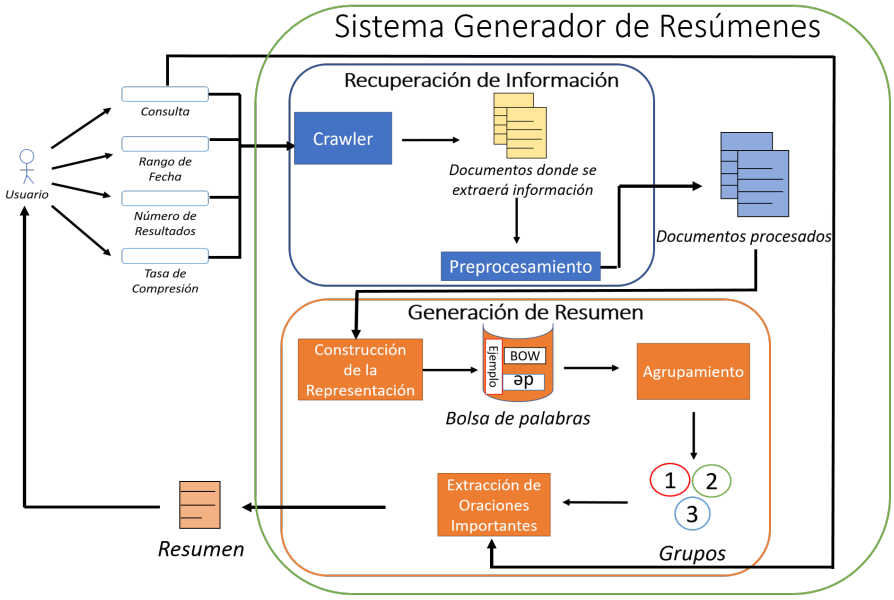

Fig. 1. Arquitectura general del sistema propuesto.

Como es posible observar en la figura 1, el sistema recibe como parámetros de entrada la 'consulta', 'rango de fechas', y la 'tasa de compresión' que deberá

contener el resumen. La consulta proporcionada sirve para que el módulo de recuperación de información descargue de la web todos aquellos documentos que se presumen relevantes a la consulta. El parámetro de rango de fechas sirve para delimitar la búsqueda, de igual forma el parámetro de 'número de resultados'. Una vez concluida la descarga, el módulo de generación de resumen entra en acción. En su interior, éste utiliza técnicas de agrupamiento y de PLN para la construcción del resumen final. Note que para la construcción del resumen se utiliza la consulta dada por el usuario, sin embargo es importante mencionar que esta consulta puede variar con respecto a la consulta que provoco la descarga. A continuación se describirá con mayor detalle el funcionamiento de cada uno de los componentes del sistema desarrollado.

3.1. Recuperación de información

Nuestro módulo de Recuperación de Información (RI) tiene como objetivo la obtención de la información relevante a una consulta. Para la versión actual de nuestro sistema, la fuente de información donde se hace la búsqueda de información es el dominio del periódico 'El Universal'[4].

De esta forma, como cualquier sistema de RI, el principal parámetro de entrada es la consulta del usuario, sin embargo, con el afán de delimitar el proceso de búsqueda, dos parámetros adicionales son incorporados: rango de fechas, y número de resultados. El primero ayuda a establecer los rangos de fechas entre los cuales debieron haber sido publicados los documentos para ser recuperados. El segundo parámetro establece a través de un número específico la cantidad de documentos que se deben de recuperar. Si el usuario decide emplear el segundo parámetro, la descarga siempre se hará del documento más reciente al más antiguo hasta cumplir con el valor establecido.

Específicamente, los submódulos que se desarrollaron para el motor de RI son: un crawler y un módulo de pre-procesamiento de información. A continuación describimos los objetivos de cada uno.

Crawler. Para el desarrollo del crawler fue necesario familiarizarse con la estructura del sitio de 'El Universal'. Una vez realizado esto, lo que nuestro crawler hace es aprovechar el motor de búsqueda propio de la página del universal para localizar aquellas noticias que satisfacen la necesidad de información del usuario, es decir, se localizan todos aquellos documentos que contienen los términos de la consulta proporcionada por el usuario.

Una vez localizadas los documentos relevantes, el crawler navega por cada uno de estos documentos descargando solo aquellos que satisfacen la restricción de las fechas proporcionadas por el usuario, o si se da un número máximo de descargas, se descargan del más actual al más antiguo hasta cubrir el requisito. Es importante mencionar que el crawler descarga en el disco local solo el texto de cada documento, de momento, no se recuperan fotos y/o vídeos. Un ejemplo de una nota descargada por nuestro sistema es

[4] http://www.eluniversal.com.mx/

mostrado en la figura 2. Observe que del lado izquierdo se muestra la misma nota en su formato original en el sitio de El Universal.

Fig. 2. Visualización de la noticia original(lado izquierdo) y la noticia descargada por el *Crawler*(lado derecho).

Pre-procesamiento de la información. Una vez que se han obtenido los documentos relevantes, este módulo hace un limpiado de la información quitando etiquetas XML, HTML o código Java Script. Cuando el documento ha sido limpiado el texto es guardado en documentos .txt, y son nombrados con el numero de descarga y el nombre de la noticia, ejemplo: *"12 Elecciones 2018.txt"(sin comillas)*, indicando que es la noticia número 12 descargada. Internamente, el módulo de pre-procesamiento segmenta las noticias en oraciones[5] y genera dos copias de cada noticia (figura 3) segmentadas por oraciones. La copia 1 es utilizada para obtener las oraciones en su formato original, mientras que las oraciones almacenadas en la copia 2 pasan por un proceso de normalización, es decir truncado, llevadas a minúsculas, eliminación de símbolos de puntuación, etc. Los documentos en su versión de la copia 2 son la entrada del módulo de generación del resumen.

3.2. Generación de resumen

Este módulo es el encargado de procesar la información con la finalidad de construir un resumen que satisfaga las necesidades de información de un usuario en especifico. Como se mostró en la figura 1, este módulo comprende

[5] Para hacer la segmentación se utilizó la función `sent_tokenize` proporcionada por la biblioteca de Python NLTK (`https://www.nltk.org`)

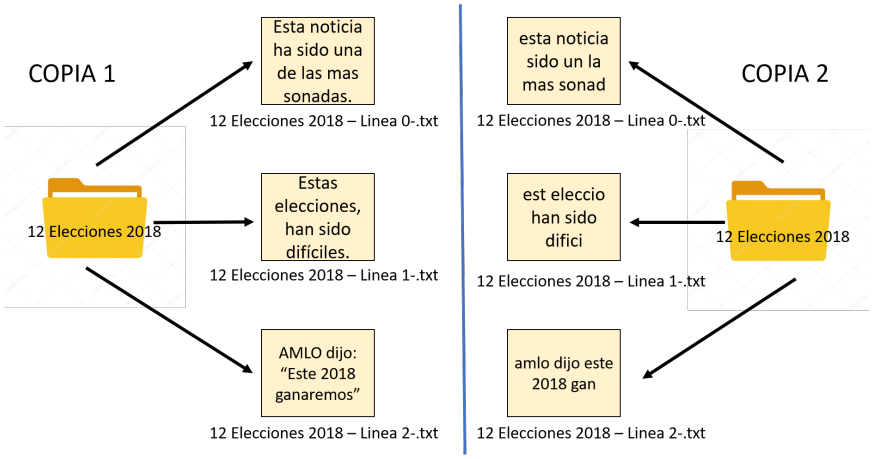

Fig. 3. Almacenamiento interno de la información en dos modalidades, información original (izq.) , e información pre-procesada para su posterior análisis (der.).

varios procesos: construcción de la representación, agrupamiento, y extracción de oraciones importantes. A continuación describiremos el objetivo de cada uno de estos procesos.

Construcción de la representación. El primer paso obligado es el *indexado* de las oraciones (S), actividad que denota hacer el mapeo de una oración s_i en una forma compacta de su contenido. La representación más comúnmente utilizada para representar textos es un vector con términos ponderados como entradas, concepto tomado del modelo de espacio vectorial usado en recuperación de información. Esto es, cada texto s_i es representado como el vector $\overrightarrow{s}_i = \langle w_{ki}, \ldots, w_{|\tau|i} \rangle$, donde τ es el *diccionario*, *i.e.*, el conjunto de términos que ocurren al menos una vez en algún elemento de S, mientras que w_{ki} representa la importancia del término t_k dentro del contenido del documento s_i. Este método de representación, también conocido como bolsa de palabras (BoW), propone varios esquemas para definir w_{ki}, en particular, para nuestro sistema desarrollado se utilizó el esquema de pesado TF-IDF [2].

Clustering. El *Clustering* o agrupamiento es un proceso que nos permite encontrar los distintos sub-temas contenidos en la colección de documentos. Este proceso nos ayudará a identificar la información relevante y redundante que existe en dicha colección de documentos. Para este proceso se utilizó el algoritmo de agrupamiento estrella [1] (Algoritmo 1). Entre las bondades de este método se tiene que induce de manera natural el numero de grupos [6] en una colección. La salida del algoritmo son grupos de documentos en forma de estrella en donde el centro de la estrella es el documento mas representativo del grupo y los satélites son los documentos que están relacionados a ese centro de estrella.

Datos: Grafo G, umbral de similitud σ
Resultado: Grupos en forma de estrella
Calcular $G_\sigma = (V, E_\sigma)$ donde $E_\sigma = \{e \in E : w(e) \geq \sigma\}$;
Poner cada vértice en G_σ inicialmente marcado como *no-visitado*;
mientras *no estén todos los vértices marcados como visitados* **hacer**
1. Tomar el vértice de mayor grado que tenga la etiqueta "no-visitado" como centro de la estrella;
2. Construir un grupo con éste como centro de la estrella y sus satélites con sus vértices asociados;
3. Marcar cada nodo de la estrella recién construida como "visitados";

fin

Algoritmo 1: Algoritmo de agrupamiento estrella.

La entrada del algoritmo de agrupamiento estrella es un grafo G el cual es un grafo completo con aristas de peso variable. Para nuestro caso, este grafo es generado a través de considerar a todas las oraciones s_i de la colección de documentos de entrada como los vértices de G, mientras que las aristas llevan como peso el valor de similitud entre los vértices respectivos[6]. De esta forma, el grafo umbralizado G_σ, es un grafo no dirigido obtenido de G al ir eliminando todas las aristas cuyos pesos son menores a σ.
Observe que otro parámetro de entrada del método descrito en el algoritmo 1 es el valor de σ, el cual representa un valor de similitud mínimo que deben de tener los elementos de cada grupo formado. El valor de σ se define por medio de la similitud media entre los documentos de entrada más la desviación estándar. De esta forma, aseguramos que los grupos formados (sub-temas) sean realmente relacionados.

Extracción de oraciones importantes. El objetivo de este proceso es la construcción final del resumen. Para esto, se toma como entrada los diferentes sub-temas encontrados por la etapa de agrupamiento y la consulta del usuario que deberá guiar el resumen. Para esto se identifican las oraciones de mayor similitud de cada sub-tema con la consulta del usuario. De esta forma, estas oraciones se incorporan en el resumen hasta cubrir la tasa de compresión solicitada por el usuario.

4. Sistema desarrollado

El sistema desarrollado consiste en una aplicación de escritorio la cual permite al usuario generar resúmenes a partir de noticias descargadas del sitio 'El Universal', a lo cual le denominamos "Resumen en-línea". Por otro lado, también se incorporó un modo en el cual el usuario puede proporcionar directamente la colección de documentos que quiere analizar, a lo cual denominamos "Resumen fuera-de-línea". Para la programación de esta herramienta se utilizó los lenguajes

[6] Para la versión actual del sistema, solo la medida del coseno está considerada [2].

Python y Java, lo cual permite que el sistema funcione en plataformas tanto Windows como Linux[7].

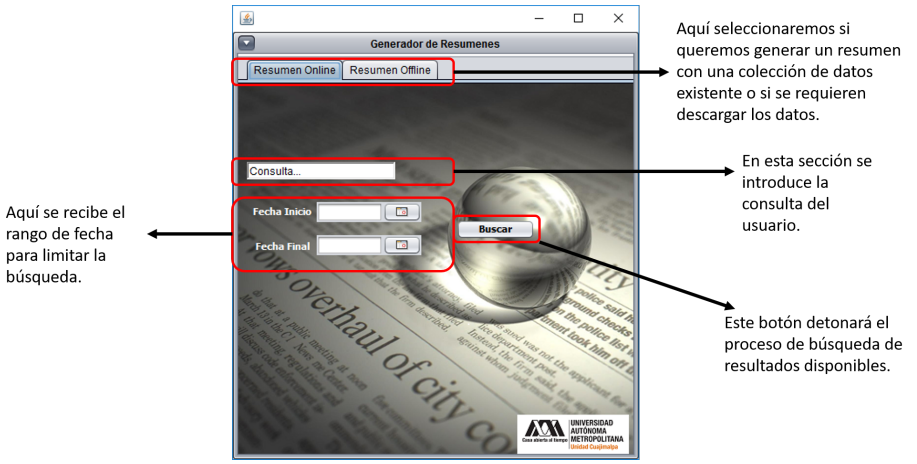

Fig. 4. Visualización de la interfaz en la sección "Resumen en-línea'. Esta ventana permite conectarse a Internet y descargar todos aquellos documentos relacionados con la consulta proporcionada entre los rangos de fechas especificados.

En la figura 4 se muestra la pantalla del modo "Resumen en-línea". Como se puede ver en la figura los parámetros que se requieren del usuario en esta primer pantalla es una consulta y un rango de fechas entre los cuales desea buscar documentos. Una vez se hace la búsqueda, el sistema mostrará una ventana como la que se visualiza en la figura 5. En esta pantalla el usuario puede ver cuántos documentos se encontraron, y además de que se le da la opción de definir un número específico de documentos a descargar. Al mismo tiempo se le pide que defina una tasa de compresión para la generación del resumen final. Al hacer click en el botón 'Descargar y generar resumen', el proceso de construcción del resumen comenzará.

En la figura 6 se muestra el modo de operación 'Resumen Off-line'. Bajo este modo de funcionamiento no es necesario tener una conexión a Internet para que el sistema trabajo, sin embargo el usuario deberá indicar la ubicación de los documentos que se quieren analizar. Continuando con el ejemplo anterior, si se quisiera seguir analizando una colección previamente descargada, basta con especificar la ruta en donde se guardaron los documentos.

Note que en el modo 'fuera-de-línea' se pide nuevamente una *consulta* (*query*) al usuario así como la tasa de compresión del resumen a construir. Bajo este esquema es posible generar resúmenes distintos dependiendo de la consulta proporcionada por el usuario.

[7] La versión ejecutable de la herramienta desarrollada se puede descargar desde: `http://ccd.cua.uam.mx/~evillatoro/Resources/SistemasGeneradordeResmenes.rar`

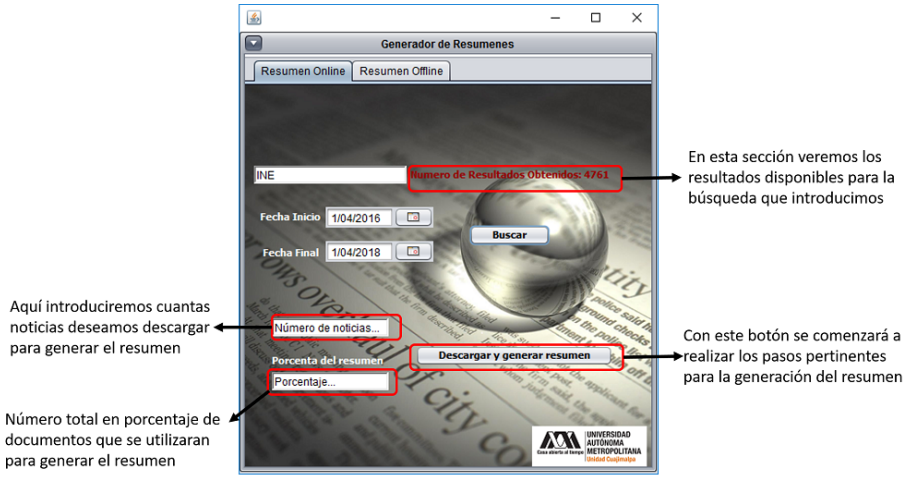

Fig. 5. Interfaz en la sección "Resumen en-línea"que se muestra una vez que se han identificado los documentos relevantes a la consulta. En rojo se muestran el total de documentos encontrados; una vez que el usuario define la tasa de compresión y presiona el botón de descarga, se comienza el proceso de generación del resumen.

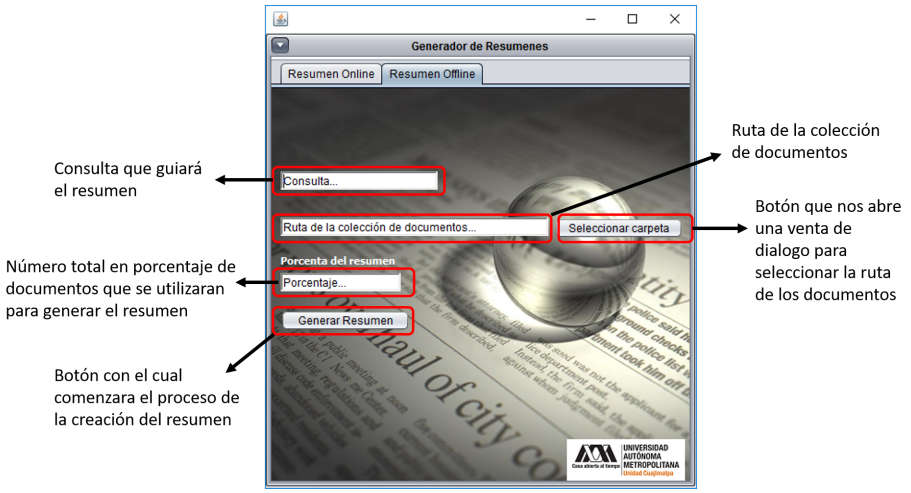

Fig. 6. Ventana que se muestra en el modo de funcionamiento 'fuera-de-línea'. El usuario debe especificar la ruta de dónde están los documentos sobre los cuales quiere trabajar, la consulta, y la tasa de compresión del resumen. Una vez definidos estos parámetros, se procede a la construcción del resumen.

Finalmente, una vez construido el resumen se visualizará una ventana como la mostrada en la figura 7 donde será posible ver el resumen construido. Este resumen es también almacenado en la ubicación donde se encuentra la aplicación

Tabla 1. Resúmenes generados ante dos consultas distintas empleando la misma colección de documentos.

consulta$_1$: *Dinero invertido en armas y la relación del narcotráfico con Estados Unidos.* *Resumen_1:* De acuerdo con Pamela Starr, experta en la relación entre Estados Unidos y México de la Universidad del Sur de California, la medida tiene sentido y guarda relación con la supuesta oferta que Trump hizo al presidente mexicano, Enrique Peña Nieto, de colaborar en la lucha contra el narcotráfico y los cárteles de la droga, la cual se habría planteado durante la conversación telefónica que mantuvieron hace un par de semanas. Según expedientes judiciales, la organización efectuaba sus actividades desde la zona suburbana de Dallas, y Treviño Morales había invertido 16 millones de dólares de dinero proveniente del narcotráfico en la compra y entrenamiento de caballos para que participaran en carreras en el suroeste de Estados Unidos. Desde 2004, Yarrington enfrenta acusaciones por presunto narcotráfico y lavado de dinero. El narcotráfico, la inseguridad.
consulta$_2$: *Los estados con mayor indice de violencia y corrupción policíaca con el narcotráfico.* *Resumen_2:* Aseguró que las fuerzas de seguridad están siendo maltratadas, hay corrupción en los mandos, convenios con los cárteles y corrupción en los ministerios públicos. Evitar que fueran reclutados por los criminales, con la prevención social de la violencia y la delincuencia, y por el otro, asegurar mayores oportunidades de educación de calidad para las y los jóvenes. El narcotráfico, la inseguridad. .En 20 años de gobierno del PRD y Morena, nuestras familias han visto como se deteriora la ciudad por la corrupción.

de escritorio del sistema.

En la Tabla 1 se muestra un ejemplo de dos resúmenes construidos a partir de necesidades de información diferentes empleando la misma colección de documentos. Para el ejemplo mostrado se descargaron 3,883 noticias empleando la consulta *Narcotráfico*, mientras que para la generación de los resúmenes se emplearon las siguientes consultas: *Dinero invertido en armas y la relación del narcotráfico con Estados Unidos*, y *Los estados con mayor indice de violencia y corrupción policíaca con el narcotráfico*. Para la construcción del resumen se solicitó al sistema una tasa de compresión del 1 %.

Como es posible observar, el sistema genera resúmenes muy diferentes a necesidades de información planteadas, lo cual permite, hasta cierto punto, responder a las necesidades de usuarios diferentes. Internamente, una vez que el sistema identifica a las oraciones candidatas para estar en el resumen, las mismas son ordenadas de acuerdo a su nivel de similitud con la consulta, y son colocadas en este orden hasta cubrir la restricción de tamaño solicitado.

5. Conclusiones

Este trabajo describe la implementación de un sistema de generación de resúmenes de múltiples documentos guiado por consulta. El sistema desarrollado

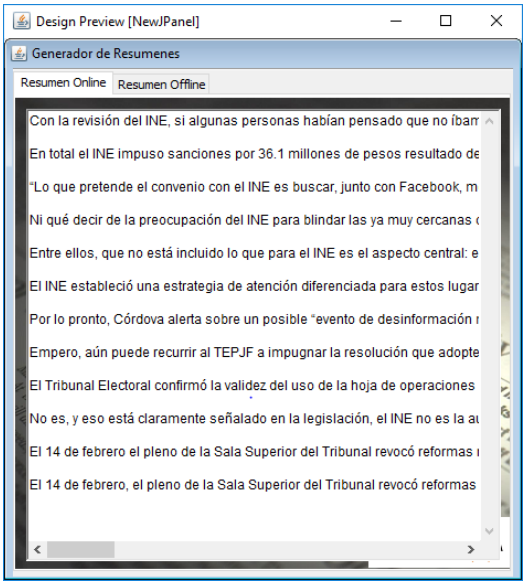

Fig. 7. Ventana que se muestra con el resumen final. Una vez finalizado el proceso de generación del resumen, se muestra una ventana de texto con las piezas de información que se identificaron como más relevantes. Si el usuario desea generar otro resumen con otra consulta diferente, bastará con volver a las ventanas previas.

es capaz de responder a distintas necesidades de información por medio de considerar una consulta proporcionada en lenguaje natural por uno o varios usuarios. Una de las ventajas del sistema desarrollado es que se incorporó un módulo de recuperación de información, el cual se conecta en tiempo real a el sitio de noticias de El Universal para descargar documentos que satisfacen una necesidad de información inicial. Posteriormente, usuarios con diferentes perfiles o necesidades de información pueden generar variados resúmenes que responden a consultas específicas.

Para el desarrollo del presente sistema se utilizaron técnicas de Inteligencia Artificial así como de Procesamiento de Lenguaje Natural. Por un lado, en lo que respecta a PLN, técnicas tradicionales de representación de textos así como de cálculo de similitud entre documentos son utilizadas para procesar los documentos. Por otro lado, técnicas de aprendizaje no supervisado son empleadas para la identificación de tópicos importantes entre los documentos descargados.

Durante la implementación del sistema fue posible observar que mejores formas de representación de la información pueden ser incorporadas al sistema, así como técnicas más eficientes de agrupamiento. Como parte del trabajo futuro, queremos incorporar representaciones que capturen de manera más eficiente la semántica de los documentos, de forma que sea posible identificar oraciones relevantes aunque éstas no compartan términos de manera explícita. Agregado a eso, queremos emplear estrategias de agrupamiento jerárquico, técnica que

permitirá construir el resumen considerando otro esquema de organización de la información relevante. Ambas estrategias permitirán la construcción de resúmenes más valiosos para el usuario final.

Agradecimientos. Agradecemos a la Coordinación de la Licenciatura Tecnologías y Sistemas de Información de la Universidad Autónoma Metropolitana, Unidad Cuajimalpa por el apoyo otorgado para la realización de este trabajo.

Referencias

1. Aslam, J., Pelekhov, K., Rus, D.: A practical clustering algorithm for static and dynamic information organization. In: Proceedings of the 1999 Symposium on Discrete Algorithms. pp. 208–217 (1999)
2. Baeza-Yates, R., Ribeiro-Neto, B., et al.: Modern information retrieval, vol. 463. ACM press New York (1999)
3. Burges, C.J., Ragno, R., Le, Q.V.: Learning to rank with nonsmooth cost functions. In: Advances in neural information processing systems. pp. 193–200 (2007)
4. Canhasi, E., Kononenko, I.: Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization. Expert Systems with Applications 41(2), 535–543 (2014)
5. Chuang, W.T., Yang, J.: Text summarization by sentence segment extraction using machine learning algorithms. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. pp. 454–457. Springer (2000)
6. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM computing surveys (CSUR) 31(3), 264–323 (1999)
7. Luhn, H.P.: The automatic creation of literature abstracts. IBM Journal of research and development 2(2), 159–165 (1958)
8. Luna-Tlatelpa, L., Villatoro-Tello, E., Ramírez-de-la Rosa, G., Rivero-Moreno, C.J.: Resúmenes de múltiples documentos guiados por consulta empleando representaciones distribucionales. Research in Computing Science 134, 127–139 (2017)
9. Ouyang, Y., Li, W., Li, S., Lu, Q.: Applying regression models to query-focused multi-document summarization. Information Processing & Management 47(2), 227–237 (2011)
10. Petsche, T., Mozer, M.C., Jordan, M.I.: Advances in Neural Information Processing Systems 9: Proceedings of the 1996 Conference. MIT Press (1997)
11. Wang, L., Raghavan, H., Castelli, V., Florian, R., Cardie, C.: A sentence compression based framework to query-focused multi-document summarization. arXiv preprint arXiv:1606.07548 (2016)