Universidad Central “Marta Abreu” de Las Villas
Facultad de Matemática, Física y Computación
Licenciatura en Ciencia de la Computación

Trabajo de Diploma

Nuevo modelo de agrupamiento para documentos XML utilizando estructura y contenido

Autor
Ivett Elena Fuentes Herrera

Tutores
MSc. Damny Magdaleno Guevara
Dra. María Matilde García Lorenzo

Santa Clara, 2013

Hago constar que el presente trabajo fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de los estudios de la especialidad de Ciencia de la Computación, autorizando a que el mismo sea utilizado por la institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos ni publicado sin la autorización de la Universidad.

_____

Firma del autor

Los abajo firmantes, certificamos que el presente trabajo ha sido realizado según acuerdos de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

_____	_____
Firma del tutor	Firma del jefe del Seminario de Inteligencia Artificial

A mis padres

AGRADECIMIENTOS

A Dios.

A mis padres por apoyarme y confiar en mí siempre.

A mis tutores por su entrega y apoyo incondicional todo el tiempo.

A mis amigos por alegrarme y estar siempre pendientes del avance en este proyecto.

A todos mis profesores, en especial al grupo de Inteligencia Artificial de quienes he aprendido mucho.

A todos los que los que veían terminado este trabajo, aun sin haberlo comenzado.

A todos, gracias.

RESUMEN

Cada día más datos electrónicos en formato semiestructurado se encuentran disponibles en el World Wide Web, intranets corporativas, y otros medios de comunicación. Gestionar el conocimiento a partir de la información encontrada es fundamental en el trabajo científico. La gestión de información científica se vuelve cada vez más compleja y desafiante, sobre todo porque las colecciones de documentos generalmente son heterogéneas, grandes, diversas y dinámicas. Superar estos desafíos es esencial para dar a los científicos mejores condiciones de administrar el tiempo necesario para procesar la información científica.

En este trabajo se implementó el sistema LucXML, con un nuevo método de agrupamiento automático de documentos XML a partir del contenido y la estructura existente en los mismos, sustentado en un sistema para la gestión de la información existente en los artículos científicos, que contribuye al descubrimiento de conocimiento relevante.

Se definió la función de similitud *OverallSimSUX* que facilita capturar el grado de semejanza entre los documentos tomando como génesis la relación existente entre la colección como un todo y las sub-colecciones resultantes de las unidades estructurales. La evaluación a través los experimentos y los casos de estudios definidos arrojaron mejores resultados con la metodología propuesta, que con otras variantes existentes en la literatura.

ABSTRACT

The amount of electronic data with semistructured format available on the World Wide Web, intranets, and other media increases every day. Knowledge Management from the information found is essential in scientific papers. The management of scientific information becomes increasingly complex and challenging, especially since document collections are usually heterogeneous, large, diverse and dynamic. Overcoming these challenges is essential to give scientists a better position to manage the time needed to process scientific information.

In this thesis a system named LucXML was implemented, with a new method of automatic clustering for XML documents based on the content and structure existing in them and supported by a system for information management in scientific papers, which contributes to relevant knowledge discovery.

The similarity function OverallSimSUX was defined, which facilitates to capture the degree of similarity between documents using as genesis the entire collection and the relationship between the structural units, when handled as independent collections. The evaluation through defined experiments and data sets achieves better results with the proposed methodology than with other variants of the literature.

Tabla de Contenidos

INTRODUCCIÓN

XML (*Extensible Markup Language*) es un metalenguaje desarrollado por el W3C [1] proveniente de GML (*Generalized Markup Language*) que surgió por la necesidad que tenía la empresa de almacenar grandes cantidades de información. Un documento XML es una estructura jerárquica autodescriptiva de información, que consiste en un conjunto de átomos, elementos compuestos y atributos (Dalamagas et al., 2006).

A esto se añade que los documentos XML contienen su información en forma semiestructurada (Abiteboul, 1997) ya que incorporan estructura y datos en una misma entidad. Son extensibles, con estructura de fácil análisis y procesamiento, por lo que XML se ha convertido en el formato de intercambio de datos estándar entre las aplicaciones Web (Dalamagas et al., 2006). Las etiquetas existentes en los documentos XML permiten la descripción semántica del contenido de los elementos. De este modo, la estructura de los documentos puede ser explotada para realizar recuperación de documentos relevantes (Guerrini et al., 2006).

La proliferación de información disponible en el *World Wide Web*, intranets corporativas y bases de datos, cables de noticias electrónicas, y otros medios de comunicación es arrolladora. La creación y diseminación de información es soportada por un número creciente de herramientas, sin embargo, mientras que la cantidad de información disponible está continuamente creciendo, nuestra habilidad de procesarla y asimilarla permanece constante (Dixon, 1997, Lanquillon, 2001). Cada día más datos electrónicos son presentados en la Web en formato semiestructurado (Dalamagas et al., 2006). Por tanto, es necesario que las computadoras resuelvan esta incapacidad humana.

Gestionar el conocimiento a partir de la información encontrada es fundamental en el trabajo científico (Passoni, 2005). Sin embargo, la gestión de información científica se vuelve cada vez más compleja y desafiante, sobre todo porque las colecciones de documentos generalmente son heterogéneas, grandes, diversas y dinámicas. Superar estos desafíos es

[1] http://www.w3c.org

esencial para dar a los científicos mejores condiciones de administrar el tiempo necesario para procesar la información científica, lo cual constituye la motivación principal de este trabajo.

En la actualidad diversos gobiernos y organizaciones científicas con el propósito de asegurar el uso productivo de la información; dirigen gran parte de sus proyectos al desarrollo de sistemas, que faciliten el proceso de toma de decisiones óptima y contribuyan de esta forma a la Gestión del Conocimiento (Bueno, 2001, Dalkir, 2005, Canals et al., 2003).

Existen varias formas de gestionar el conocimiento: la categorización, la clasificación y el agrupamiento (Dixon, 1997, Tan, 1999).

Particularmente, el agrupamiento nos permite organizar la información, delimitar la información relevante y descubrir nuevo conocimiento a partir de la información disponible en una colección especificada u obtenida como resultado de un proceso de recuperación de información (C.D. et al., 2008).

Para una eficiente organización y recuperación de los documentos relevantes, una posible solución es agrupar los documentos XML basándose en su estructura y/o en su contenido (Tien T., 2007).

Un algoritmo de agrupamiento intenta encontrar grupos naturales de datos, basándose principalmente en la similitud y las relaciones de los objetos, de forma tal que se obtenga una distribución interna del conjunto de datos mediante su particionamiento en grupos. Cuando el agrupamiento se basa en la similitud de los objetos, se desea que los objetos que pertenecen al mismo grupo sean tan similares como se pueda y los objetos que pertenecen a grupos diferentes sean tan disímiles como sea posible. El análisis de grupos es una herramienta para descubrir una estructura previamente oculta en los datos, asumiendo que existe un agrupamiento natural o cierto en ellos. Sin embargo, la asignación de los objetos a las clases y la descripción de esas clases son desconocidas (Kruse et al., 2007).

El desarrollo de sistemas que faciliten a los usuarios gestionar grandes colecciones de documentos, mediante la organización y extracción del conocimiento es una necesidad real. Estos sistemas a partir de una colección personal presentada como entrada, deben proponer como salida, grupos homogéneos de documentos afines y la calidad con que fueron obtenidos los grupos, proporcionando el control para la evaluación de los resultados del agrupamiento obtenido (Arco, 2009).

En (Arco, 2009) se propone un esquema general que detalla una variante para procesar colecciones textuales, de manera que se extraiga la información organizada y se presentan los elementos relevantes de cada grupo de documentos relacionados. Este esquema está compuesto por cuatro módulos: recuperación de la información o especificación del corpus textual a procesar, representación del corpus textual obtenido o fijado por el usuario, agrupamiento de los documentos y valoración (validación y etiquetamiento) de los grupos textuales obtenidos.

En el Centro de Estudios de Informática (CEI) de la Universidad Central "Marta Abreu" de las Villas (UCLV) se han propuesto los sistemas para la gestión de la información y el conocimiento (SATEX y GARLucene) que implementan el esquema propuesto por (Arco, 2009) para la confección de sistemas gestores de información en dominios textuales. Aunque estos sistemas brindan amplias ventajas para la gestión de la información y del conocimiento, su diseño no incorpora un algoritmo de agrupamiento capaz de explorar la estructura de documentos en formato semiestructurado, específicamente en documentos XML.

Lo antes expuesto ratifica una problemática que la ciencia aún no aborda de manera completa y justifica el siguiente **planteamiento de investigación**:

Los trabajos dirigidos al agrupamiento de documentos XML se clasifican principalmente en tres categorías: los que se centran solo en el contenido, los que utilizan solo la estructura y los que tienen en cuenta ambas componentes. La mayoría de los enfoques existentes no utilizan estas dos dimensiones en conjunto dada su gran complejidad. Sin embargo, para obtener mejores resultados en el agrupamiento, es esencial utilizar ambas.

El **objetivo general** de esta investigación consiste en implementar un nuevo método de agrupamiento automático de documentos XML sustentado en un sistema para la gestión de la información existente en los artículos científicos, a partir del contenido y la estructura existente en los mismos, que contribuya al descubrimiento de conocimiento relevante.

Este objetivo se desglosa en los siguientes **objetivos específicos**:

1. Realizar un análisis crítico sobre el estado actual de las técnicas de agrupamiento, enfatizando en aquellas utilizadas en colecciones de textos semiestructurado y determinar la variante a implementar.

2. Definir una función de similitud para el agrupamiento.
3. Diseñar e implementar un modelo para el agrupamiento de documentos en formato XML, utilizando el contenido y la estructura presente en los mismos.
4. Evaluar el modelo propuesto a partir de corpus de XML, representativos del universo investigado, utilizando los resultados obtenidos por el software que soporta el modelo.

Las **preguntas de investigación** planteadas son:

1. ¿Cómo combinar la relación estructura-contenido de los documentos XML, a nivel de las Unidades Estructurales (UE) existentes en los documentos?
2. ¿Cómo fusionar la relación entre los documentos objetos de estudio, teniendo en cuenta los agrupamientos realizados por cada unidad estructural y la visión global de todas las unidades estructurales?
3. ¿En qué medida el nuevo modelo aporta mejores resultados al agrupamiento de documentos XML que otras variantes propuestas?

Como respuestas a las preguntas de investigación y después de haber realizado el marco teórico se formuló la siguiente **hipótesis de investigación**:

H1: Los algoritmos de agrupamiento de documentos XML, que combinan la relación estructura-contenido a nivel de las Unidades Estructurales y fusionan en una única medida de similitud estas dependencias, logran mejores resultados en el agrupamiento, contribuyendo a una eficiente gestión de la información y el conocimiento.

Para lograr los objetivos trazados y demostrar la hipótesis planteada se acometieron las siguientes **tareas de investigación**:

- Análisis de los métodos de agrupamiento de documentos XML y colecciones textuales, medidas de validación de los resultados.
- Estudio de la herramienta *Lucene* para la gestión documental.
- Definición de una nueva función de similitud que permita capturar la relaciones de dependencia entre los documentos recuperados, mediante la combinación de los resultados de agrupamientos para cada UE con la información global del documento.

- Diseño de un nuevo método de agrupamiento de documentos XML, que utilice la relación estructura-contenido.
- Implementación del método propuesto para el agrupamiento.
- Comparación del método general propuesto con otros métodos utilizados para el agrupamiento de documentos XML.

El **valor teórico** de la investigación está directamente vinculado con su novedad científica.

El **valor práctico** del trabajo está enfocado a:

- Disponer de un algoritmo de agrupamiento, que permita procesar grandes volúmenes de datos y obtener conocimiento relevante a partir de la información recuperada, con el propósito de facilitar a los investigadores y docentes el inicio de una revisión del estado del arte, organizar materiales por equipos de estudiantes para la docencia, organizar por temáticas los artículos que han sido recopilados por el comité científico de un evento, así como tener una idea de las asociaciones que existen entre los documentos recuperados.

1

ACERCA DE LOS MÉTODOS DE AGRUPAMIENTO DE DOCUMENTOS XML

1. ACERCA DE LOS MÉTODOS DE AGRUPAMIENTO DE DOCUMENTOS XML

El volumen de datos e información disponible está en continuo ascenso; una de sus causas es el crecimiento exponencial de las colecciones de datos en formato semiestructurado; específicamente las almacenadas en formato XML (Dalamagas et al., 2006). Por esta razón, es fundamental desarrollar nuevas técnicas que permitan el análisis exploratorio de estos datos y que capturen eficientemente las relaciones internas que describen la propia estructura jerárquica y autodescriptiva de estos documentos. Particularmente, a partir de la información disponible, el agrupamiento permite organizar, delimitar relevancia y descubrir nuevo conocimiento (Dixon, 1997, Tan, 1999). A continuación, se presenta un análisis del formato XML como herramienta para el almacenamiento semiestructurado de información y las principales técnicas para el agrupamiento de documentos en este formato. Antes de finalizar el capítulo, se exponen algunas facilidades de una poderosa herramienta para la gestión de la información, obtenida como resultado de un proceso de recuperación.

1.1 Qué es XML

Algunos autores plantean que los documentos son unidades indivisibles e independientes (Martín, 2007). Estas unidades pueden representar obras literarias, artículos científicos, imágenes, etc. Reflexionando brevemente sobre el concepto de documento, se pueden encontrar múltiples tipos en los que resulta más natural tratarlos como un conjunto de partes; entre estos se encuentran los artículos científicos, que normalmente constan de título, resumen, palabras claves, una serie de secciones (que pueden dividirse en varias subsecciones y así sucesivamente), conclusiones, entre otras. Consecuentemente un conjunto dado de documentos $D= \{D_1,\ldots, D_m\}$, se corresponden con un conjunto de unidades estructurales $U= \{U_1,\ldots, U_n\}$. De esta forma, desaparece el concepto de documento como unidad indivisible (Martín, 2007).

La forma que se almacena la información estructurada es un aspecto muy importante a tener en cuenta y para ello se utilizan lenguajes específicos. No es ningún secreto que XML es el lenguaje más utilizado hoy en día para este tipo de tarea (Martín, 2007).

XML fue creado por el W3C proveniente de GML que surgió por la necesidad que tenía la empresa de almacenar grandes volúmenes de información. Su desarrollo comenzó en 1996 y la primera versión salió a la luz el 10 de febrero de 1998. Es un formato basado en texto con una sintaxis muy simple. Fue originalmente diseñado para resolver los problemas de la publicación electrónica.

Un documento XML es una estructura jerárquica autodescriptiva de información, que consiste en un conjunto de átomos, elementos compuestos y atributos (Dalamagas et al., 2006). A esto se añade que los documentos XML contienen su información en forma semiestructurada (Abiteboul, 1997) ya que incorporan estructura y datos en una misma entidad.

XML tiene un número de características que lo han hecho ser ampliamente utilizado como formato de representación de datos. Es extensible e independiente de la plataforma utilizada. Su extensibilidad se manifiesta de varias formas. Primeramente, a diferencia de HTML, no tiene un conjunto de tags[2] fijo, por tanto, las palabras claves del lenguaje no están definidas desde el principio. Así, con XML es posible definir un lenguaje específico para aplicaciones concretas. Las aplicaciones utilizadas para el procesamiento de los documentos en este formato son fácilmente extensibles, en el sentido que admiten cambios de carácter aditivo. Por ejemplo, una aplicación que dependa del procesamiento de elementos de tipo mensaje con un subelemento remitente seguiría funcionando correctamente si se añade otro atributo como destinatario. Esta flexibilidad no es común en otros lenguajes y es un beneficio derivado de usar XML.

Por otra parte, es independiente de la plataforma utilizada, del sistema operativo, o del fabricante de software. De hecho, es bastante posible producir (o procesar) XML utilizando una amplia variedad de productos y lenguajes de programación. Esta independencia lo hace ideal como un medio de alcanzar interoperabilidad entre plataformas diferentes de programación y sistemas operativos. El hecho de que sea un formato basado en texto posibilita leer y editar documentos XML usando tan sólo simples editores de texto.

[2] Indistintamente se utiliza el término tag o etiqueta para referirse a un elemento de un documento XML.

Finalmente, soporta internacionalización. Esto quiere decir que los archivos en este formato admiten cualquier tipo de codificación. Particularmente, se puede trabajar con archivos UTF-8, que permiten la representación de caracteres de cualquier idioma.

Al ser extensibles, con estructura de fácil análisis y procesamiento, XML se ha convertido en el formato de intercambio de datos estándar entre las aplicaciones Web (Dalamagas et al., 2006), teniendo un papel muy importante en la actualidad, ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

1.2 Estructura de un documento XML

La tecnología XML busca dar solución al problema de expresar información estructurada de la manera más abstracta y reutilizable posible. Que la información sea estructurada quiere decir que se compone de partes bien definidas, y que esas partes se componen a su vez de otras partes. En la Figura 1.1 se muestra un ejemplo de documento XML correspondiente a un artículo científico, el árbol que contiene la estructura de este documento se muestra en la Figura 1.2.

En este ejemplo es posible observar como el artículo está dividido en varias partes, que a su vez pueden estar divididas en subpartes. En un documento XML estas partes se llaman elementos, y se les señala mediante etiquetas.

Una etiqueta consiste en una marca hecha en el documento, que señala una porción de éste como un elemento, representando de esta forma un fragmento de información con un sentido claro y definido. Las etiquetas tienen la forma `<nombre> contenido </nombre>`, donde *`<nombre>`* es el nombre del elemento que se está señalando y *`</nombre>`* indica el fin de la misma.

Las etiquetas existentes en los documentos XML permiten la descripción semántica del contenido de los elementos. De este modo, la estructura de los documentos puede ser explotada para realizar recuperación de documentos relevantes (Guerrini et al., 2006).

Lo antes expuesto hace que cada día más datos electrónicos sean presentados en formato XML (Dalamagas et al., 2006), no obstante, la habilidad de las herramientas existentes para extraer conocimiento permanece constante (Dixon, 1997, Lanquillon, 2001). Con este continuo crecimiento de los datos semiestructurados, hay una necesidad inevitable de manejar

eficazmente estos grandes volúmenes datos (Dalamagas et al., 2006). El agrupamiento de los documentos XML basándose en su estructura y/o en su contenido contribuye a una eficiente organización y recuperación de los documentos relevantes (Tien T., 2007).

```
<?xml version="1.0" encoding="ISO-8859-1" ?>

<Artículo>
   <Título>
      "Agrupamiento de documentos estructurados"
   </Título>
   <Resumen>
      En este trabajo se propone realizar un agrupamiento…
   </Resumen>
   <Introducción>
      XML es el lenguaje mas utilizado en para…
   </Introducción>
   <Secciones>
      <Sección1>
         La estructura de los documentos XML juega un papel[1]…
      </Sección1>
      <Sección2>
         Un algoritmo de agrupamiento…
      </Sección2>
                                    …
      <Sección n>
         La estructura de los documentos XML juega un papel…
      </Sección n>
   </Secciones>
                                    …
   <Referencias>
      1. Autor, XML, su estructura…
   </Referencias>
</Artículo>
```

Figura 1.1 Ejemplo de un documento XML correspondiente a un artículo científico.

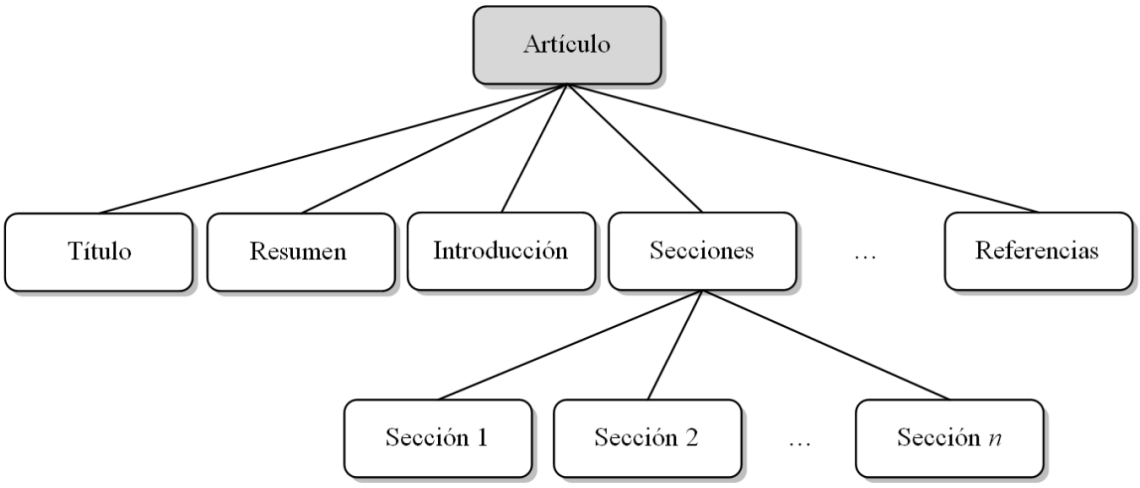


Figura 1.2 Ejemplo de un árbol correspondiente a un artículo científico.

Estos algoritmos de agrupamiento se mantienen con un gran auge en los últimos tiempos, como consecuencia del crecimiento de datos electrónicos en este formato. El agrupamiento es fundamental para una eficiente organización y recuperación de los documentos XML relevantes. Sin embargo, la mayoría de los métodos existentes explotan solo la información incluida en el contenido o sólo la información contenida en la estructura (Tien T., 2007).

1.3 Agrupamiento de documentos XML

El análisis de grupos[3] es descrito como una herramienta para el descubrimiento porque tiene la potencialidad de revelar relaciones basadas en datos complejos no detectadas previamente. Los algoritmos de agrupamiento son usados para encontrar una estructura de grupos que se ajuste al conjunto de datos, logrando homogeneidad dentro de los grupos y heterogeneidad entre ellos (Anderberg, 1973, Kruse et al., 2007).

El agrupamiento forma usualmente las bases del aprendizaje y el conocimiento, por tanto, para el correcto funcionamiento de muchos de los sistemas de minería de textos y recuperación de información una posible solución es el empleo de técnicas de agrupamiento (Dixon, 1997, Tan, 1999).

1.3.1 Agrupamiento

Un algoritmo de agrupamiento, como se mencionó anteriormente intenta encontrar grupos naturales basándose principalmente en la similitud y relaciones de los objetos, de forma tal que se obtenga una distribución interna del conjunto de datos mediante su particionamiento en grupos, es decir sigue el principio de maximizar la similitud dentro del grupo y minimizar la similitud entre los grupos (Anderberg, 1973, Kruse et al., 2007).

El concepto de "similitud" tiene que especificarse acorde a los datos. En la mayoría de los casos los datos son vectores de valores reales, entonces se requieren algunas medidas (distancias, similitudes, o disimilitudes) para cuantificar el grado de asociación entre ellos; lo más común es seleccionar la medida qué será utilizada con determinado método. Las medidas que se usan con mayor frecuencia se muestran en el Anexo 1.

[3] En esta tesis se emplean indistintamente los términos: grupos, conglomerados, clases, comunidades y subconjuntos.

Al mismo tiempo, es un reto descubrir grupos en datos que al relacionarse forman una estructura interesante para el análisis. Este tipo de datos ha tenido una mejor descripción cuando se representa como una colección de objetos interrelacionados y enlazados (Getoor and Diehl, 2005). El enlace entre objetos es un conocimiento que puede ser explotado en el agrupamiento, ya que rasgos de objetos enlazados están correlacionados, y es probable la existencia de enlaces entre objetos que tienen elementos comunes (Arco, 2009).

Representar los objetos y sus relaciones en un grafo y explotar su topología para descubrir los grupos es la idea de varios métodos. Estas propuestas ven el conjunto de datos desde la perspectiva de las conexiones entre los objetos más que los objetos en sí mismos (Girvan and Newman, 2002). Los conjuntos de datos pueden intrínsecamente formar un grafo o se pueden obtener grafos de similitud a partir de la matriz de similitud entre los objetos.

En la actualidad se presupone que el conocimiento de la estructura de los datos es tan importante como los objetos en sí. Ese conocimiento puede ayudar a descubrir grupos que se ocultan en las comunicaciones entre los objetos (Baumes et al., 2005, Tong and Faloutsos, 2006).

1.3.2 Clasificación de las técnicas de agrupamiento

Para realizar análisis de grupos han sido propuestos una gran variedad de algoritmos de agrupamiento. Estos pueden ser clasificados de diversas formas: tipo de los datos de entrada del algoritmo, criterios para definir la similitud entre los objetos, conceptos en los cuales se basa el análisis y forma de representación de los datos, entre otros.

En general, en esta clasificación se distinguen dos tipos: aquellos que forman particiones y los jerárquicos (Kruse et al., 2007).

Los métodos que forman particiones tienen como objetivo encontrar la mejor partición de los datos en k grupos ($k \in \mathbb{N}, k > 0$) basada en una medida de similitud dada y conservar el espacio de particiones posibles en k subconjuntos solamente. La mayoría de los algoritmos que siguen esta técnica son esencialmente basados en prototipos, comienzan con una partición inicial, usualmente aleatoria, y proceden con su refinamiento (Kruse et al., 2007). Uno de los algoritmos perteneciente a esta clasificación y que ha sido ampliamente utilizado es el *k-medias* (*k-means*) (Xiong et al., 2006).

Por otra parte, los algoritmos jerárquicos hacen una descomposición jerárquica de los objetos. Dentro de ellos, los aglomerativos (bottom-up), comienzan considerando que cada objeto constituye un grupo, por tanto inicialmente existen tantos grupos como objetos tiene la colección, y sucesivamente los van uniendo, hasta que todos los objetos formen un único grupo, generalmente considerando una medida de distancia. Mientras que los divisivos (top-down) consideran inicialmente que existe un único grupo al cual pertenecen todos los objetos y sucesivamente van dividiendo los grupos en grupos más pequeños, hasta que cada grupo contenga un único objeto. La construcción de la jerarquía se puede detener por criterios automáticos o del usuario. Trabajos como (Cheng et al., 2006, Cheng et al., 2005) combinan la estrategia divisiva y la aglomerativa.

Otra clasificación, no mutuamente excluyente a las ya presentadas, considera la forma de manipular la incertidumbre en términos del solapamiento de los grupos: agrupamiento duro y borroso (Höppner et al., 1999). Las técnicas duras pueden ser deterministas o con solapamiento. Las deterministas crean una partición, donde los grupos son mutuamente excluyentes y exhaustivos del universo de objetos. Los algoritmos con solapamiento crean un cubrimiento, donde un objeto puede pertenecer a más de un grupo. Las borrosas se subdividen en probabilísticas y posibilistas (Kruse et al., 2007).

1.3.3 Técnicas para el agrupamiento de documentos XML

Cuando se trata de documentos XML, los algoritmos de agrupamiento se clasifican principalmente en tres grupos: los que se centran solo en el contenido de los documentos (Kurgan et al., 2002, Shen and Wang, 2003), realizando un análisis solamente léxico, o incluyendo elementos sintácticos o semánticos en el estudio, aquellos que realizan análisis léxico generalmente consideran los documentos como una bolsa de palabras; sin embargo, un buen proceso de agrupamiento no puede descartar el uso de la estructura (Tran et al., 2008b), por lo que están los algoritmos que utilizan solo la estructura, considerando que esta juega un papel importante en el agrupamiento para ciertas aplicaciones específicas y los que combinan ambas componentes: estructura y contenido, lo cual constituye un nuevo desafío, ya que la mayoría de los enfoques existentes no utilizan estas dos dimensiones dada su gran complejidad (Tien T., 2007).

1.3.3.1 Algoritmos que utilizan solo la estructura de los documentos

La estructura jerárquica de los documentos XML juega un papel importante en el agrupamiento (Nayak, 2006). Tener en cuenta solo esta, tiene aplicaciones interesantes para la extracción de información, la integración de datos heterogéneos, entre otras (Guerrini et al., 2006). Varios trabajos utilizan la estructura en forma de árbol para realizar el agrupamiento, por lo que dado un documento XML, el agrupamiento se realizaría obviando el contenido, utilizando solamente la estructura correspondiente, ver Figuras 1.1 y 1.2. A continuación se tratarán algunos de estos.

Una de las variantes para comparar árboles es utilizar la distancia *Tree-Edit* (TE), que intenta transformar un árbol A_1 en un árbol A_2, realizando una secuencia de operaciones (inserción, eliminación y substitución de nodos), a las que le asigna un costo. De manera que mientras menor sea la cantidad de operaciones necesarias en la transformación, mayor será la similitud entre los árboles correspondientes a los documentos comparados. Una gran cantidad de trabajos utilizan TE (Flesca et al., 2005, Dalamagas et al., 2006, Nierman and Jagadish, 2002, Chawathe et al., 1996, Chawathe, 1999, Zhang and Shasha, 1989, Selkov, 1977) o alguna de sus variantes.

Debido a que los documentos XML presentan varios elementos repetidos y/o anidados, la diferencia en cuanto a tamaño y a estructura puede ser muy alta si se usa la distancia TE, aun cuando estos compartan el mismo DTD[4], por lo que se ha propuesto el cálculo del *Structural Summaries* (Dalamagas et al., 2006) para reducir el anidamiento y las repeticiones, obteniéndose una representación lo más reducida que conserve la relaciones jerárquicas entre los elementos del árbol asociado al documento y luego se aplica el cálculo de la distancia TE. Ver Figura 1.3.

Otra variante es a través del cálculo de los *Closed Frequent Subtrees* (CFST). Los autores del artículo referenciado en (Kutty et al., 2007) plantean que dado un conjunto de árboles T, existe para un árbol T_i un subárbol ST_i que mantiene la misma relación padre-hijo que T_i; se calcula la frecuencia $f(ST_i)$, que no es más que la cantidad de árboles pertenecientes a T de los que ST_i

[4] Mignet, L., et al. (2003) "The XML web: a first study", 12th International Conference on WWW.

es subárbol, y este es frecuente (*FST*) si $f(ST_i)$ es mayor que un umbral determinado. Se puede decir que dos árboles pertenecen a un mismo grupo si tienen el mismo *FST*.

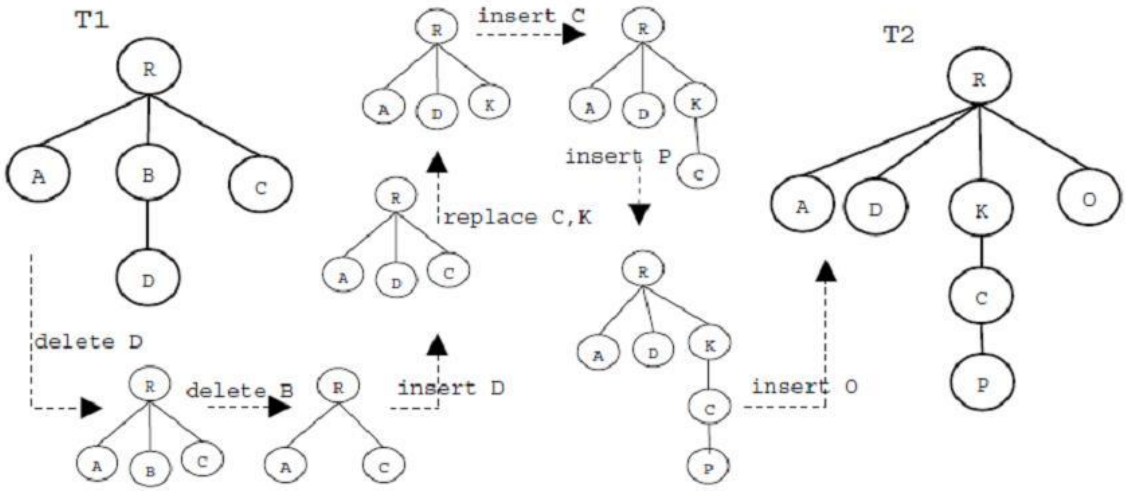


Figura 1.3 Uso de la distancia Tree-Edit.

Esto tiene como inconveniente que pueden existir muchos *FST*, por lo que se busca el *CFST* que es un FST_i que es superconjunto de otros *FST* que poseen la misma $f(FST_i)$ y no existe otro superconjunto para FST_i. Con esto se conforma una matriz (*T*x*CFST*) en la que cada celda va a tener la existencia o no de $CFST_i$ en T_j. Para buscar los grupos se sigue un criterio muy parecido al antes mencionado.

Otro trabajo basado en la estructura de los árboles correspondientes a los documentos y que no utiliza la distancia TE se presentó por (Lian et al., 2004), donde introducen el concepto de grafo estructurado (*s-graph*) de un documento *D*, el cual es muy similar a otra herramienta introducida por (Goldman and Widom, 1997) para datos semiestructurados. Los trabajos (Chawathe et al., 1996, Chawathe, 1999) se basan en técnicas de programación dinámica y el uso de *EditGraph* respectivamente, utilizan solo la estructura de los documentos para realizar agrupamientos. En (Xing et al., 2007) se utiliza la TE para comparar los DTD de los documentos. En (Kirsten and Wrobel, 2000) se aplicó un razonamiento basado en casos para integrar conocimientos previos para calcular una mejor similitud en los documentos.

1.3.3.2 Algoritmos que combinan estructura y contenido

Un nuevo desafío en el agrupamiento de documentos XML lo constituye el desarrollo de algoritmos a partir de la combinación de estructura y contenido. La mayoría de los enfoques existentes como se ha referido antes en este trabajo no utilizan estas dos dimensiones dada su gran complejidad. Sin embargo, para obtener mejores resultados en el agrupamiento, es esencial utilizar ambas dimensiones (Kutty et al., 2008). A continuación, se mencionan algunos trabajos existentes en la literatura.

Una primera variante muy sencilla es mezclar en una representación Espacio Vectorial (Vector Space Model*;* VSM) (Salton et al., 1975) el contenido y las etiquetas del documento y aplicar un algoritmo de agrupamiento conocido.

En la representación VSM cada documento se identifica como un vector de rango en el que cada dimensión corresponde a términos distintos que se han indexados con anterioridad.

En una matriz VSM, se almacena un valor numérico que indica la importancia de cada término en cada documento, utilizando para esto su frecuencia de aparición. Comúnmente este valor se identifica como una función que expresa cuán importante es un término *j* en un documento *i*, ignorándose la secuencia en la que los términos aparecen en el documento. Esta matriz es la que combina el contenido y estructura de los documentos, y se utiliza para realizar el agrupamiento de los mismos.

Existen varios pasos que permiten la transformación de una colección de documentos original a esta representación. Inicialmente cada documento se transforma en una secuencia de tokens. El número de tokens recuperados puede resultar extremadamente grande, por tanto es necesario reducir la dimensionalidad (Magdaleno, 2008).Este proceso se puede realizar a partir de la técnica de selección de rasgos que intenta encontrar el mejor subconjunto de todos los posibles que permita mejores resultados en el agrupamiento, lo cual puede verse como un problema de optimización guiada por heurística.

Otros trabajos realizan extensiones a la representación VSM, llamadas C-VSM y SLVM (Doucet and AhonenMyka, 2002, Giannopoulos and Veltkamp., 2002, Karmarkar, 1984, Yang and Chen, 2002). En ambas representaciones para cada documento se conforma una matriz M_{ext}, donde *e* es el número de elementos y *t* el número de términos; cada celda va a contener la frecuencia de cada término t_i en el elemento e_j. La diferencia radica en que para el caso de

C-VSM solo se comparan términos que pertenezcan a elementos comunes en dos documentos y en SLVM se realiza la comparación de términos de un elemento con los términos correspondientes en cualquier elemento del otro documento. C-VSM al ignorar la relación semántica entre diferentes elementos presenta el problema de "baja contribución" y SLVM al no tener en cuenta la relación entre elementos comunes puede presentar el problema de "sobre contribución".

Con el propósito de eliminar estas dificultades en (Wan and Yang, 2006) se propone la Proportional Transportation Similarity, donde se trabaja con comparaciones pesadas según la semejanza o no de los elementos a comparar en dos documentos.

Otro enfoque se muestra en (Tran et al., 2008a), donde realiza primeramente un agrupamiento teniendo en cuenta solo la estructura de los documentos, ver Figura 1.4, posteriormente proponen el uso del Latent Semantic Kernel (Cristianini et al., 2002) para determinar la similitud entre el contenido de los documentos y realiza un agrupamiento teniendo en cuenta el contenido. Además utilizan un método para reducir dimensionalidad basado en la información estructural común de los documentos, ya que el uso del singular vector decomposition es muy costoso computacionalmente cuando se tratan un gran número de rasgos (Cristianini et al., 2002).

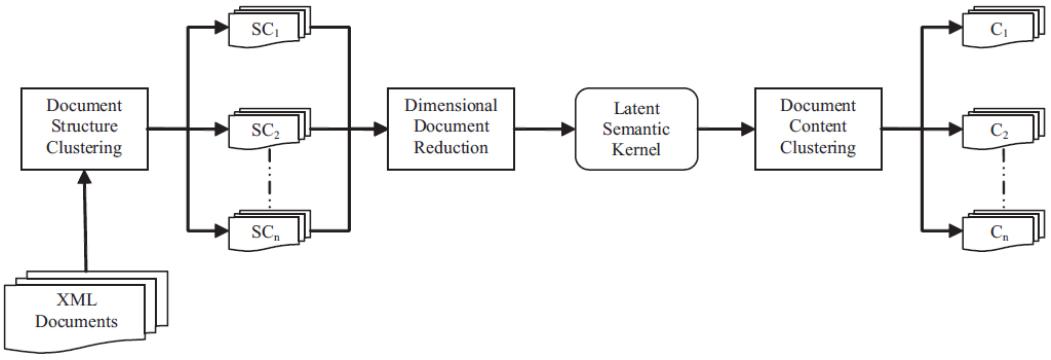


Figura 1.4 Esquema de algoritmo que utiliza estructura y contenido para agrupar documentos XML.[5]

En (Nayak, 2006) desarrollaron el XCLSE que es una modificación al algoritmo de agrupamiento que utiliza solo la estructura XCLS (Nayak and Xu., 2006), e incorpora una

[5] Tomado de TRAN, T., KUTTY, S. & NAYAK, R. 2008a. Utilizing the Structure and Data Information for XML Document Clustering. *INEX*, 402-410.

comparación a nivel semántico antes de realizar el agrupamiento. Esta propuesta no mejoró significativamente los resultados alcanzados por XCLS, aunque sí tuvo que realizar un importante esfuerzo de cálculo para la semántica de los datos.

1.4 Manipulación de documentos en formato XML

El surgimiento de los repositorios de datos electrónicos y la explosión de Internet han hecho posible que grandes volúmenes de información se encuentren disponibles al alcance de la sociedad. Consecuentemente, el crecimiento continuo de información hace prácticamente imposible su gestión mediante la aplicación de métodos tradicionales (Hatcher et al., 2009).

En la actualidad la diferencia competitiva entre las instituciones (se usará este término y el de organizaciones para referirse indistintamente a instituciones de servicios, empresas, etc.) se basa en un nuevo factor: la información y sobre todo, en su adecuada sistematización en orden a convertirla en conocimiento. Las instituciones ya saben que las ventajas competitivas, a medio y largo plazo, no van a venir de la información, algo que en mayor o menor medida es de acceso universal y no representará ningún valor diferenciador, sino del conocimiento, entendiéndole como el grado de incorporación, sistematización y utilización de esa información para mejorar los resultados de estas (Magdaleno et al., 2011a). La información en sí misma no supone ninguna ventaja, su sistematización es la que aporta ese valor añadido. A diferencia de hace unos pocos años, hoy existe una "invasión" de información, surgiendo un problema: cómo procesarla y usarla en beneficio propio. De todo lo anterior se deduce que la explotación del conocimiento en aras de la obtención de una ventaja competitiva sostenible requiere de alguna herramienta que lleve a cabo esa sistematización de la información.

1.4.1. Lucene

En el proceso de gestión de la información, la indexación y la búsqueda son pasos claves. *Lucene* es una potente biblioteca de búsqueda e indexación basada en *Java* y de código abierto, que fácilmente permite la integración con cualquier aplicación (Egüe, 2011).

En los últimos años *Lucene* se ha convertido en una popular biblioteca de recuperación de información que ha sido integrada a las funciones de búsquedas de muchas aplicaciones web y de escritorio. Aunque fue originalmente desarrollado en *Java*, debido a su popularidad ya se ha implementado en otros lenguajes de programación como (*C*/*C++*, *C#*, *Ruby*, *Perl*, *Python*,

PHP, etc.). Uno de los factores clave detrás de la popularidad de *Lucene* es su aparente simplicidad, pues realmente cuenta con algoritmos que implementan técnicas de recuperación de la información de última generación (MATTMANN and ZITTING, 2012). Además, para utilizarla no es necesario un conocimiento profundo acerca de cómo indexa y recupera información.

Esta biblioteca se adapta fácilmente a cualquier aplicación que requiera indexación y búsqueda completa de texto y es ampliamente conocida por su utilidad en la implementación de motores de búsqueda en Internet y locales.

El principio fundamental de la filosofía de trabajo de *Lucene* consiste en un documento compuesto por campos de texto. Textos de documentos en formato PDF, HTML, XML y muchos otros pueden indexarse, siempre que de ellos se pueda extraer información textual.

Resumiendo: *Lucene* constituye una novedosa herramienta para la gestión de la información. Creada bajo una metodología orientada a objetos e implementada completamente en *Java*, permite la búsqueda y recuperación de información, sobre una indexación. Sus fuentes se encuentran totalmente disponibles, elemento esencial para decidir utilizarla. *Lucene* es multiplataforma, tiene un alto rendimiento y es escalable, permite la creación incremental de índices, los algoritmos de búsqueda son potentes, fiables y eficientes, facilita: ordenar resultados por relevancia, utilizar un amplio lenguaje de consulta, realizar búsquedas por campos y por rangos de fechas, ordenar por cualquier campo, y buscar mientras se actualiza el índice. No se trata de una aplicación que pueda ser descargada, instalada y ejecutada sino de una interfaz de programación de aplicaciones (Application Program Interfaces; API) flexible, muy potente y realmente fácil de utilizar, a través de la cual se pueden añadir, con pocos esfuerzos de programación, capacidades de indexación y búsqueda a cualquier sistema para la gestión de la información.

Para la creación de los índices, *Lucene* necesita definir para cada documento un conjunto de campos. Una herramienta que facilita la confección de estos campos, es el API *Jdom*, especializada en la manipulación de documentos en formatos XML. Esta biblioteca permite identificar los *elementos* existentes en un documento.

Lucene confecciona el índice de términos utilizados en la creación de la representación VSM. Para realizar el preprocesamiento de la colección posee varias clases: *StandardAnalyzer*,

especializada en normalizar los tokens extraídos; *LowerCaseFilter*, convierte los tokens a minúsculas y *StopFilter* elimina palabras de parada (Lewis and Ringuette, 1994). Adicionalmente, *Analyzer* obtiene las raíces de las palabras mediante heurísticas, y tratar la sinonimia y polisemia.

1.5 Consideraciones finales del capítulo

Existen múltiples tipos de documentos que resulta más natural tratarlos como un conjunto de unidades estructurales y no como una unidad indivisible, entre ellos, los artículos científicos. Resulta XML el lenguaje más utilizado hoy en día para almacenar información estructurada ya que posibilita la fácil lectura y edición de los documentos con este formato, usando tan sólo simples editores de texto.

Ante el continuo crecimiento de los datos semiestructurados, hay una necesidad inevitable de manejar eficazmente estos grandes volúmenes datos.

El agrupamiento de los documentos XML basándose en su estructura y/o en su contenido contribuye a una eficiente organización y recuperación de los documentos relevantes.

2

MODELO DE AGRUPAMIENTO DE DOCUMENTOS XML

2. MODELO DE AGRUPAMIENTO DE DOCUMENTOS XML

Los métodos propuestos para agrupar documentos XML basados en las relaciones estructura o contenido por separado, no utilizan las conexiones entre ambas dimensiones; limitando la similitud existente entre estos documentos. El problema de estructurar un universo de objetos[6] presupone: la determinación del Espacio de Representación Inicial (ERI); la determinación de una función de semejanza y de un criterio de agrupamiento, que en buena medida significa la forma en que se utiliza esta semejanza en el espacio de representación para formar los agrupamientos. En este capítulo se presenta: (1) un modelo[7] general para la aplicación del agrupamiento, combinando la relación estructura-contenido; (2) una nueva medida de semejanza que facilita evaluar el grado de relación entre los documentos; (3) la implementación del procedimiento general que sustenta el modelo.

2.1 Modelo para el Agrupamiento

Se propone una metodología general para la aplicación del agrupamiento de documentos XML, con el propósito de facilitar a los usuarios enfrentarse a grandes colecciones de documentos, a partir de su organización; contribuyendo a la extracción de conocimiento relevante. El nuevo modelo se inicia a partir del resultado de un proceso de recuperación de información (Berry, 2004). Las salidas son grupos homogéneos de documentos afines, el resumen de cada documento, los documentos más representativos de cada grupo y la calidad del agrupamiento; garantizando el control para la evaluación de los resultados.

Una visión gráfica del esquema del modelo general presentado en este trabajo se muestra en la Figura 2.1. En la Figura 2.2 se muestra los cuatros módulos principales que contiene el modelo propuesto.

[6] En este trabajo específicamente son documentos XML.

[7] En esta tesis se utilizan indistintamente los términos modelo y metodología.

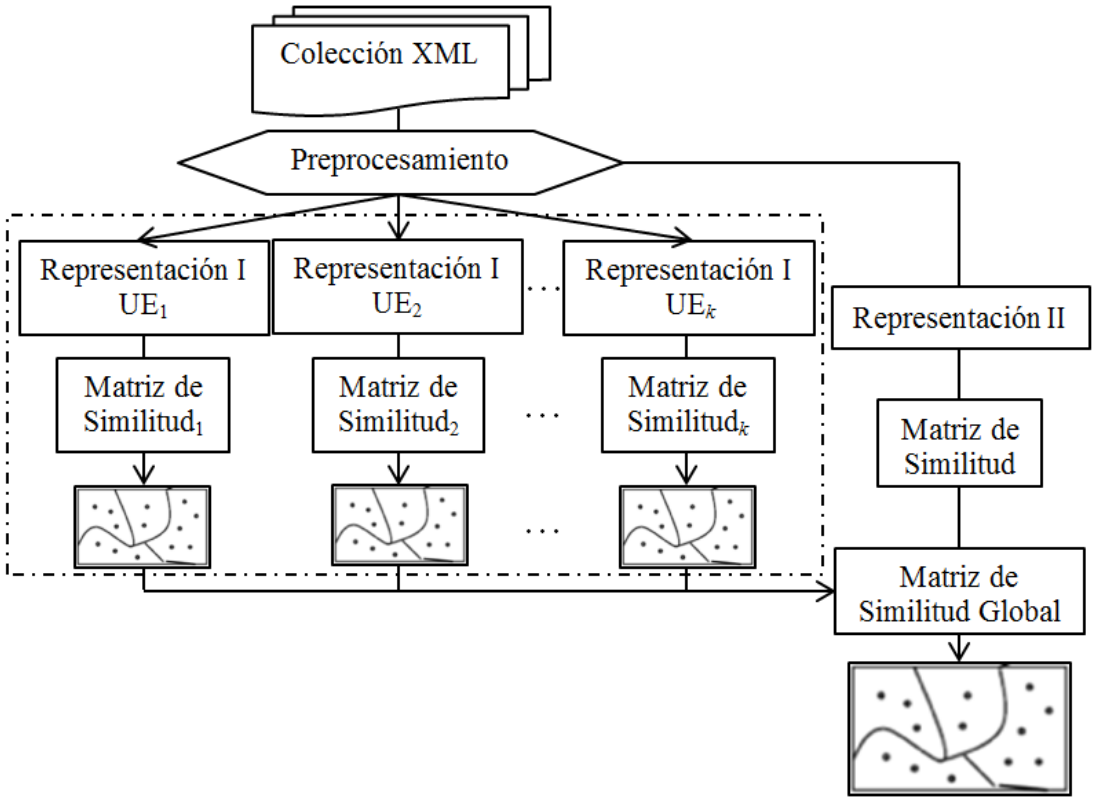


Figura 2.1 Esquema que muestra el modelo general propuesto.

Módulo 1. Recuperación de la información o especificación del corpus textual a procesar, identificando en cada documento recuperado las Unidades Estructurales (UE).

Módulo 2. Representación del corpus textual obtenido.

Submódulo 2.1. Tratar cada UE como una colección diferente. Obtener por cada UE una representación basada en la VSM clásica, denominada en este trabajo *Representación I*.

Submódulo 2.2 A partir de las UE identificadas. Obtener una representación global que tendrá en cuenta el contenido en función de la estructura. Esta representación, es denominada *Representación II*.

Módulo 3. Agrupamiento de los documentos.

Submódulo 3.1. Realizar un agrupamiento por cada UE, a partir de la matriz de similitud resultante de la *Representación I*.

Submódulo 3.2. Obtener la matriz de similitud, a partir de la *Representación II*.

Submódulo 3.3. Realizar el agrupamiento general a partir del cálculo de la función de semejanza, propuesta en esta investigación, que utiliza como entrada el resultado de los submódulos 3.1 y 3.2.

Módulo 4. Valoración (validación y verificación) de los grupos obtenidos.

Figura 2.2 Módulos principales del modelo propuesto.

2.1.1 Representación del corpus textual obtenido

La información contenida en los documentos XML se encuentra en formato semiestructurado, por lo que, la representación textual es indispensable para su procesamiento posterior. En este trabajo se seleccionó la representación VSM por ser efectiva para representar documentos y ser ampliamente reconocida en la comunidad de minería de textos. En VSM cada documento se identifica como un vector de rasgos en un espacio en el cual cada dimensión corresponde a términos indexados distintos (palabras). Un vector documento dado, en cada componente tiene un valor numérico para indicar su importancia. Según Lanquillon (Lanquillon, 2001), la representación textual está compuesta por la transformación del corpus, la extracción de términos, la reducción de la dimensionalidad, la normalización y el pesado de la matriz. *Lucene*[8] provee herramientas para manipularlas.

2.1.1.1 Transformación del corpus

En el modelo propuesto se transforma el corpus convirtiendo los ficheros de entrada en una secuencia de tokens de palabras. En el paso subsecuente a la extracción de términos, estos tokens se usan para generar rasgos significativos (índices de términos).

El primer paso en la trasformación del corpus, se concentra en procesar los documentos XML determinando en qué UE del documento se encuentra cada token, además se identifican *k* colecciones independientes. La Definición 2.1 denota la correspondencia entre colección y UE, a partir del concepto de *k*-colección.

Definición 2.1 (*k*-colección). Sea *D* un conjunto de documentos XML, entonces la *k*-colección del conjunto *D* está formada por el conjunto de documentos D_{UEk}, donde:

$$D_{UEk}: \{d_{ik} = UE_{ik} | \forall d_i \in D\} \quad (2.1)$$

Segundo, la secuencia resultante de tokens se transforma convirtiendo todas las letras a minúsculas, eliminando las marcas de puntuación al final de los tokens, omitiendo los tokens que contienen caracteres alfa-numéricos, y sustituyendo las contracciones por sus expresiones completas (Lanquillon, 2001).

[8] http://lucene.apache.org

2.1.1.2 Extracción de términos

Para obtener las representaciones que utiliza el modelo, se parte de una secuencia de tokens y se produce una secuencia de términos indexados basados en esos tokens. En este trabajo se realiza un análisis léxico de los textos, se identifican las palabras simples como rasgos. Así, se explota básicamente el plano estadístico de los textos y no se considera la secuencia de aparición de las palabras en un documento (modelo bolsa de palabras; bag-of-words model) (Lewis and Ringuette, 1994). El análisis léxico es ventajoso porque la definición de los términos es independiente del lenguaje y computacionalmente muy eficiente, la representación resultante es fácil de analizar por los humanos, no obstante debido a que el número de éstos puede ser innecesariamente grande, se hace necesario la reducción de dimensionalidad.

Por otra parte, un token tiene mayor o menor importancia para la comparación de dos documentos en dependencia del lugar que este ocupe dentro de los mismos (Magdaleno et al., 2011b). Esto es, dado tres documentos d_1, d_2, d_3 correspondientes a artículos científicos y las palabras w_1, w_2,…, w_n, donde w_1,…, w_{n-k} son comunes para d_1 y d_2 y están presentes en UE importantes de los documentos (por ejemplo resumen, palabras claves), y w_{n-k+1},…, w_n son comunes para d_1 y d_3, pero están presentes en UE menos importantes de estos; la relación que existe entre los documentos d_1 y d_2 es más fuerte que la existente entre d_1 y d_3, pues al pertenecer sus palabras comunes a partes claves del documento, la información de estos dos documentos es significativamente común, comparada con la de los documentos d_1 y d_3. Por lo tanto, la información se representa teniendo en cuenta esta idea. Por lo que en este trabajo se tiene en cuenta este criterio para realizar la *Representación II*.

En los algoritmos de agrupamientos que utilizan solo el contenido de los documentos XML y consideran estos como una bolsa de palabras eliminando las etiquetas, pierden la información estructural que brindan los documentos (Wan and Yang, 2006); la frecuencia (Fr_{ij}) mostrada en la Tabla 2.1 no es más que la frecuencia de aparición del token t_i en el documento d_j. Este es el criterio seleccionado para confeccionar la *Representación I*.

En este trabajo se agrega al análisis la estructura de los documentos, por tanto en la *Representación II* la frecuencia ($tf_{dj}(t_i)$) va a ser ponderada por la UE que ocupe el token analizado, y se define en la ecuación 2.2 para un token t_i en un documento d_j.

Tabla 2.1. *Representación I*. Donde $tf_{dj}(t_i)$ es la frecuencia de aparición absoluta del término t_i en el documento d_j.

	Término_1	Término_2	…	Término_m
Documento_1	$tf_{d1}(t_1)$	$tf_{d1}(t_2)$	…	$tf_{d1}(t_m)$
Documento_2	$tf_{d2}(t_1)$	$tf_{d2}(t_2)$	…	$tf_{d2}(t_m)$
…	…	…	…	…
Documento_n	$tf_{dn}(t_1)$	$tf_{dn}(t_2)$	…	$tf_{dn}(t_m)$

$$tf_{dj}(t_i) = \sum_{k=1}^{n} (w_k * fr_{ik}) \tag{2.2}$$

$$w_{kj} = \left(e^{\left(-Long_{UE}/Long_{Doc}\right)}\right)^{pot} \tag{2.3}$$

Donde n es la cantidad de UE presentes en d_j, fr_{ik} es la frecuencia de t_i en UE_k y w_k es el peso que se le calcula a UE_k en el documento d_j. El cálculo del peso de la UE_k para cada documento d_j se realiza como se expresó en la ecuación 2.3; aquí $Long_{UE}$ es la longitud de la UE_k, $Long_{Doc}$ es la longitud del documento d_j y *pot* es un valor suministrado. Quedando así formalizada la idea.

Como resultado del proceso de extracción de términos en los submódulos 2.1 y 2.2, se obtienen *k* representaciones VSM clásicas y una representación VSM global que considera el contenido en función de la estructura.

2.1.1.3 Reducción de la dimensionalidad

Controlar la dimensionalidad del espacio vector documento, se hace necesario. Las razones principales tienen su génesis en: (i) la complejidad de muchos algoritmos de agrupamiento depende crucialmente del número de rasgos y reducirlo hace tratables estos algoritmos y (ii) existen palabras que son irrelevantes y producen peores resultados, por tanto, eliminarlas puede aumentar la eficiencia del agrupamiento a realizar (Arco, 2009).

La eliminación de palabras de parada o gramaticales (stop word elimination) (Yang and Pedersen, 1997, Mladenic and Grobelnik, 1998), es una de las técnicas de selección utilizadas en este esquema de aplicación. También se utilizan métodos de filtrado para decidir cuándo incluir un término en el vocabulario o no. El vocabulario final se establece seleccionando los

m mejores rasgos, es decir, los *m* rasgos con mayor o menor puntuación acorde a la magnitud empleada, en este trabajo se considera esencialmente las expresiones I y II de calidad de términos, mostradas en el Anexo 1 (Berry, 2004). Además, se homogenizó la ortografía y se redujeron las palabras a su forma raíz (stemming), lo cual permite reducir la dimensionalidad del espacio de rasgos haciendo corresponder palabras morfológicamente similares con la palabra raíz asociada (Frakes and Baeza-Yates, 1992, Porter, 1980).

2.1.1.4 Normalización y pesado de la matriz

Se genera un vector pesado para cualquier documento basado en el vector de frecuencias de términos. En el esquema de aplicación propuesto se utiliza TF-IDF (Berry, 2004) para pesar los valores de la matriz y se normaliza dividiendo la frecuencia de aparición de los términos por la longitud de los documentos. En la Figura 2.3 se muestra el esquema de la representación del corpus textual.

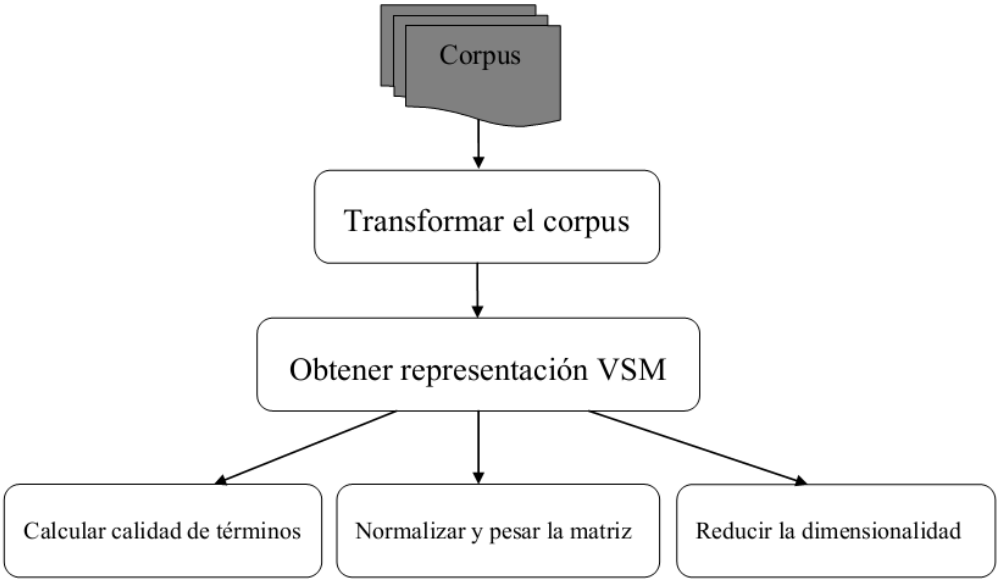


Figura 2.3 Esquema de la representación del corpus textual.

2.1.2 Similitud Coseno, función de semejanza OverallSimSUX

El problema del Reconocimiento de Patrones sin aprendizaje consiste en: dado un conjunto de objetos (muestra inicial) *MI* y β una función de semejanza entre los objetos, identificar a éstos en diferentes grupos, que responden o se generan de manera "natural" según el comportamiento global o particular de las semejanzas entre los objetos o atendiendo al cumplimiento de una cierta propiedad. (Ruiz-Shulcloper, 1995)

Resolver este problema consiste en esencia en hallar la estructura interna de los objetos en el ERI, que depende de la forma en que los objetos se comparen, es decir, del concepto de similaridad que se utilice y de la forma en que éste se emplee.

A partir de la función de semejanza β y de *MI*, se construye una matriz de similitud que refleja las relaciones de semejanza entre todos los objetos sujetos a estudio. En esta investigación se considera la pertenencia a un grupo analizando el comportamiento global de las semejanzas entre los objetos. Esto se logra siguiendo el criterio β-semejantes que se describe en la Definición 2.2 (Ruiz-Shulcloper, 1995).

Definición 2.2 (β-semejantes) Dos descripciones[9] (objetos) $I(O_i)$, $I(O_j)$ se denominan β-semejantes si $\beta(O_i,O_j)>\beta_0$, tal que consideramos β_0 como el umbral de semejanza.

2.1.2.1 Similitud Coseno

Una de las formas para definir el criterio de semejanza entre dos objetos, es la similitud coseno[10]; que ha sido ampliamente utilizada para comparar documentos que son sometidos a un proceso de agrupamiento. No obstante, existen colecciones donde dos documentos pueden tener alta similitud coseno y tratar temas diferentes. Estos documentos puede que no sean similares a sus vecinos respectivos y sólo el uso de esta similitud no logra identificarlos en grupos diferentes.

En casos como estos y particularmente en el contexto de documentos semiestructurados, se hace necesario explotar la estructura de los documentos, de forma tal que se pueda refinar este valor de similitud a partir de la relación de semejanza entre cada una de las unidades estructurales de los documentos y verificar si efectivamente existe divergencia entre las asociaciones establecidas en cada una de las UE por separado. Este análisis permitirá emitir un criterio definitivo sobre grado de relación entre los documentos, objetos de estudio.

2.1.2.2 Función de Similitud OverallSimSUX

La relación estructural existente entre los documentos XML puede aportar mejores resultados al agrupamiento, cuando se utiliza el contenido en función de la relación entre sus unidades

[9] Indistintamente se utiliza el término objeto y descriptores de objetos.
[10] En el Anexo 1 se muestra la expresión de la similitud coseno.

estructurales. En este trabajo se propone una nueva medida de similitud que facilita capturar el grado se semejanza entre estos documentos, tomando como génesis la relación existente entre sus unidades estructurales, cuando se manipulan como colecciones independientes y la similitud global.

Las consideraciones antes expuestas son el punto de partida de la medida de similitud *OverallSimSUX*, precisada formalmente a través de la Definición 2.4. Se parte de los resultados de los agrupamientos realizados a las *k*-colecciones y la matriz de similitud basada en el cálculo de la similitud coseno, a partir de la *Representación II*.

La Definición 2.3 introduce la relación λ en este enfoque. Los resultados que aquí se presentan son válidos con independencia del algoritmo que se utilice para obtener los grupos.

Definición 2.3 (λ-pertenencia) Dados los objetos *i*, *j* se define la λ-pertenencia como la relación de pertenencia de ambos objetos a un mismo grupo, a partir de los resultados del agrupamiento. Esta pertenencia se formaliza en la ecuación 2.4.

$$\lambda(i,j) = \begin{cases} 1, & \{i,j\} \epsilon grupo_k \\ 0, & i \epsilon grupo_n \ \wedge \ j \epsilon grupo_m \end{cases} \quad m \neq n \tag{2.4}$$

Definición 2.4 (*OverallSimSUX*) La similitud *OverallSimSUX* entre objetos *i*, *j* está dada por la expresión 2.5, en esta: $A = \{a_1, a_2, \dots, a_k\}$, donde a_k es el resultado del agrupamiento para la *Representación* I_k; s_g es la matriz de similitud coseno que se obtiene a partir de la *Representación II* y w_k es la ponderación de la UE_k.

$$f\left(A, s_g, i, j\right) = \frac{\sum_{k=1}^{m} (w_k * \lambda_{k(i,j)} + s_{g(i,j)})}{\sum_{k=1}^{m} w_k + 1} \tag{2.5}$$

OverallSimSUX considera *m* como la cantidad de UE identificadas en los documentos. Esta función de similitud está normalizada por la sumatoria de los pesos de las *m* UE y el máximo valor de la similitud global s_g (i.e. 1). Por tanto, su máximo (i.e. 1) se alcanza cuando los documentos *i*, *j* pertenecen al mismo grupo en todos los *k*-agrupamientos (i.e. $\lambda_k = 1$) y el valor de s_g es máximo.

2.2 Un algoritmo de agrupamiento basado en la similitud OverallSimSUX

En esta sección se explica un algoritmo de agrupamiento basado en la estrategia del algoritmo de agrupamiento *K*-Star (Shin and Han, 2003, Pinto et al., 2009) y la matriz similitud *OverallSimSUX*.

Algoritmo 1. Algoritmo de agrupamiento basado en *K*-Star.

1. Construcción de la matriz de similitud OverallSimSUX.
2. Estimación del umbral de similitud.
3. Determinación de los núcleos iniciales del agrupamiento mediante el cálculo de la máxima similitud entre dos objetos, no asignados.
4. Asignación de los objetos que no pertenecen a los núcleos a partir de su umbral de pertenencia a los grupos ya formados.

2.2.1 Construcción de la matriz de similitud OverallSimSUX

La matriz de similitud, contiene el valor de similitud *OverallSimSUX*, que existe entre los objetos. Esta función de semejanza captura de forma implícita el comportamiento global de las semejanzas de los documentos a nivel de UE, dependiente de la relación de pertenencia o no a los agrupamientos simples[11]. Así, es necesario determinar primeramente la matriz de similitud coseno asociada a cada una de las colecciones de documentos que representan a cada UE a partir de la *Representación I* y obtener los grupos asociados a los *k* agrupamientos simples, para lo cual se determinan los mismos parámetros de entrada que describe este algoritmo, exceptuando la forma de cálculo de la matriz de similitud, que constituye el único criterio de divergencia en el procedimiento para obtener los agrupamientos por UE.

2.2.2 Estimación del umbral de similitud

La estimación del umbral de similitud, permite determinar la relación mínima de semejanza que debe existir entre un objeto y un grupo ya formado, para decidir o no incorporarlo como miembro de este. Sobre la base del cálculo de la similitud definida (similitud coseno para los agrupamientos simples o similitud *OverallSimSUX* para el agrupamiento general), se define una función booleana de semejanza de la siguiente manera:

[11] Indistintamente se utiliza en esta tesis el término agrupamiento simple para hacer referencia al agrupamiento asociado a una unidad estructural en específico.

$$\Gamma(d_k, g_i) = \begin{cases} 1, & \varphi(d_k, g_i) \geq \gamma \\ 0, & \varphi(d_k, g_i) < \gamma \end{cases} \quad (2.6)$$

Donde γ es un parámetro numérico que funciona como una evaluación del umbral.

Existen diversos criterios para el cálculo del umbral que se abordarán en la sección 2.3.

2.2.3 Determinación de los núcleos iniciales

La conformación de los núcleos iniciales del agrupamiento está determinada por el cálculo inicial el máximo valor de similitud α, de esta forma el grupo inicial contiene los elementos que reportaron este máximo valor de semejanza, intuitivamente los de mayor grado de relación. Cuando no existe un nivel de similitud mayor que el umbral entre los objetos no asignados y los grupos formados, se decide recalcular este máximo y crear un nuevo grupo.

2.2.4 Asignación de los objetos que no pertenecen a los núcleos

Asociar los objetos a un grupo determinado depende del cálculo del umbral de pertenencia al grupo (umbral de similitud grupal), existen varios criterios para esto, los cuales se abordan en la siguiente sección. En este paso se asocian los documentos que no pertenecen a ninguno de los grupos antes creados si el grado de pertenencia a un grupo es mayor o igual que el umbral general calculado anteriormente.

2.3 Variantes para el cálculo del umbral de similitud entre objetos

La forma de medir la similitud y qué umbral utilizar para formar conjuntos de relaciones, es una tarea difícil que depende del dominio donde fue aplicado, cómo fueron descritos los objetos y qué nivel de granularidad se desea evaluar en los resultados. Otros elementos que influyen en la estimación del umbral son la variabilidad en la densidad de los grupos y la varianza y desviación estándar de las similitudes. Por otro lado, el umbral, en algunos casos, constituye una herramienta que tiene el usuario para hacer que el método se ajuste a sus requerimientos y características del problema (Arco, 2009).

2.3.1 Cálculo del umbral de similitud global

A continuación se exponen algunas variantes para el cálculo del umbral de similitud inicial, que requiere el algoritmo de agrupamiento propuesto en la sección anterior. El cálculo en cada

uno de los criterios se realiza a partir de la matriz de similitud y no se requiere información adicional del conjunto de datos que se procesa.

Se considera *m* como la cantidad de objetos de la colección y $s(O_i, O_j)$ el valor de similitud entre los objetos O_i y O_j (Ruiz-Shulcloper, 1995).

Definición 2.5 (Umbral de Semejanza). La magnitud γ se denominará umbral de semejanza y puede ser calculada de la siguiente manera:

1. La media de las similitudes entre todos los pares de objetos posibles; expresión 2.7:

$$\bar{X} = \frac{1}{m(m-1)} \sum_{i=1}^{m-1} \sum_{j=i+1}^{n} s(o_i, o_j) \tag{2.7}$$

2. La media de los valores máximos de las similitudes entre cualquier par de objetos; expresión 2.8:

$$\bar{X}_{max} = \frac{1}{m} \sum_{i=1}^{m-1} {}_{\substack{j=1..m \\ i \neq j}} \max[s(o_i, o_j)] \tag{2.8}$$

3. La media de los valores mínimos de las similitudes entre cualquier par de objetos; expresión 2.9:

$$\bar{X}_{min} = \frac{1}{m} \sum_{i=1}^{m-1} {}_{\substack{j=1..m \\ i \neq j}} \min[s(o_i, o_j)] \tag{2.9}$$

2.3.2 Cálculo del umbral de similitud grupal

El algoritmo de agrupamiento propuesto en la sección 2.2 requiere del cálculo de similitud grupal, o sea, determinar el nivel de semejanza de un objeto no asignado con cada uno de los grupos formados. Considerando que requiere además el cálculo del umbral inicial para formar un grupo, se aconseja utilizar el mismo criterio para el cálculo del umbral en ambos casos, debido a la semántica que expresan sus contextos. A continuación se exponen algunos criterios para el cálculo del umbral de similitud grupal.

En este contexto se considera *m* como la cantidad de objetos del grupo_i.

Definición 2.6. La magnitud $\varphi(d_k, c_i)$ se denomina umbral de similitud grupal y puede calcularse como se muestra a continuación:

1. La media de las similitudes entre todos los pares de objetos posibles que pertenecen al grupo; expresión 2.10:

$$\overline{X(o_k\,, c_i)} = \frac{1}{m}\sum_{i=1}^{m} s(o_i\,, o_k) \tag{2.10}$$

2. El valor máximo de similitud que alcanza con uno de los elementos del grupo; expresión 2.11:

$$\bar{X}_{max(o_k\,, c_i)} = \max[s(o_i\,, o_{jk})] \tag{2.11}$$

3. El valor mínimo de similitud que alcanza con uno de los elementos del grupo; expresión 2.12:

$$\bar{X}_{min(o_k\,, c_i)} = min[s(o_i\,, o_k)] \tag{2.12}$$

2.4 Procedimiento general para el agrupamiento de documentos XML

Como parte del modelo de agrupamiento propuesto se desarrolla un procedimiento general que incluye varios módulos específicos, estructurados en cuatro etapas con sus fases correspondientes que en su conjunto resumen el contenido del modelo. Cada módulo del procedimiento general se corresponde con los módulos mencionados en el modelo de agrupamiento, y se describen siguiendo el mismo orden.

Los módulos del procedimiento general son (observe el Anexo 2):

1. Recuperación y creación de índices a partir del corpus de documentos XML.
2. Representación de la colección.
3. Agrupamiento General a partir de la matriz de similitud basada en el cálculo de la similitud *OverallSimSUX*.
4. Evaluación local y global de los resultados del agrupamiento.

A continuación se describen cada uno de los módulos que conforman el procedimiento general para el agrupamiento de documentos XML, se enfatizará en cada una de las técnicas empleadas que responden al modelo, enunciado en la sección 2.1.

2.4.1 Módulo 1: Recuperación y creación de índices a partir del corpus de documentos XML

Como ya se ha mencionado, la entrada al modelo lo constituye la colección de documentos XML que se desea procesar, resultado de una búsqueda o un repositorio personal. A partir de esta especificación se comienza el proceso de recuperación utilizando primeramente el API *Jdom* de Java destinada al trabajo con documentos XML, que permite identificar las UE que se incorporan al índice creado introduciéndose las facilidades de *Lucene* (Hatcher et al., 2009).

2.4.2 Módulo 2: Representación de la colección

Se reutilizan las facilidades de *Lucene* para la representación del corpus: análisis léxico, eliminación de palabras vacías, segmentación por eliminación de afijos basada en el método heurístico de *Porter*, contenido en esta poderosa biblioteca de recuperación de información. En esta etapa se obtiene la *Representación I* asociada a cada UE y la *Representación II*. Específicamente para obtener la *Representación I* se construye la matriz VSM clásica, que contiene en sus filas el índice de términos construido utilizando *Lucene* y los documentos de la colección en sus columnas, las celdas representan la frecuencia de aparición de cada término en la UE del documento que se procesa. A partir de cada *representación* (indistintamente tipo *I* y *II*) el sistema realiza el proceso de normalización, para la *Representación I* esta razón se calcula, utilizando la frecuencia absoluta de aparición del término y la longitud del documento (se asume como longitud del documento la longitud de la UE) y para la *Representación II*, se calcula utilizando las ecuaciones 2.2 y 2.3. Luego, se aplica la medida TF-IDF (Berry, 2004) clásica y se reduce la dimensionalidad basándose en la medida de calidad de términos, asociándole a cada término el valor de su calidad. La cantidad máxima de términos utilizada es 600, según se propone en (Berry, 2004).

La combinación de la representación de cada UE con la representación global del contenido del documento en función de su estructura, constituye un aspecto novedoso que garantiza el uso de las dos dimensiones.

2.4.3 Módulo 3: Agrupamiento General a partir de la matriz de similitud basada en el cálculo de la función OverallSimSUX.

Para cada representación resultante se calcula una matriz de similitud utilizando como medida la similitud coseno, esta se muestra en la expresión 2.13. Se genera un agrupamiento para cada UE a partir de la similitud asociada a la *Representación I*.

$$S_{coseno(o_i,o_j)} = \frac{\sum_{k=1}^{m}(o_{ik} * o_{jk})}{\sqrt{\sum_{k=1}^{m} o_{ik}^2 * \sum_{k=1}^{m} o_{jk}^2}} \quad (2.13)$$

La matriz de similitud global se obtiene a partir del resultado de cada agrupamiento y la matriz de similitud asociada a la *Representación II*, utilizando como medida de similitud *OverallSimSUX*, ver ecuación 2.5. Finalmente el agrupamiento general es el resultado de aplicar el Algoritmo 1, que se describió detalladamente en la sección 2.2, utilizando la matriz de similitud confeccionada con *OverallSimSUX*.

Como resultado se obtiene una partición de la colección inicial en grupos homogéneos de documentos.

2.4.4 Módulo 4: Evaluación local y global de los resultados del agrupamiento.

Para la evaluación de los resultados se han implementado medidas externas, internas y las propuestas por *INEX*[12] para la evaluación de técnicas de clasificación supervisadas y no supervisadas de documentos XML. La medida externa implementada es *Overall F-measure* (Steinbach et al., 2000a), basada en Precisión (Pr) y cubrimiento[13] (Re) (Frakes and Baeza-Yates, 1992). Las medidas internas utilizadas son *Overall Similarity* (Steinbach et al., 2000a) y los índices *Dunn* (Dunn, 1974). Por último se incluye el cálculo de medidas basadas en *Purity*, *Micro-Purity* y *Macro-Purity* (Pinto et al., 2009).

En la Figura 2.4 se muestra la combinación de los cuatro módulos que componen el procedimiento general descrito con anterioridad.

[12] Initiative for the Evaluation of XML Retrieval.

[13] En este documento se utiliza cubrimiento como traducción de la medida recall. Adicionalmente, se utiliza el término cubrimiento para nombrar una forma de la división de los objetos después de un agrupamiento duro y con solapamiento.

Entrada: Colección de documentos en formato XML

Salida: Grupos homogéneos de documentos afines, resumen de cada documento, documentos más representativos de cada grupo y calidad del agrupamiento.

Inicio

1. Realizar Preprocesamiento /* *normalización, convertir tokens a minúsculas, eliminar palabras de parada, Stemming* */
   - Identificar Unidades Estructurales (UE)
2. **Para cada** ue ∈ UE
   - Representación-I←Realizar_Representación-I(ue)
   - Matriz_Similitud←Similitud_Coseno(Representación-I)
   - Agrupamientos[ue]←Realizar_Agrupamiento$_{ue}$(Matriz_Similitud)
3. **Fin_Para**
4. Representación-II←Realizar_Representación-II
5. Matriz_Similitud←Similitud_Coseno(Representación-II)
6. Matriz_Similitud_Global←OverallSimSUX(Agrupamientos,Matriz_Similitud)
7. Realizar Agrupamiento_General(Matriz_Similitud_Global)
8. Realizar Evaluación_Agrupamiento(Agrupamiento_General)

Fin

Figura 2.4 Procedimiento general para el agrupamiento usando OverallSimSUX.

2.5 Complejidad Computacional del Modelo Propuesto

Un creciente número de herramientas soporta la creación y diseminación de la información provocando su proliferación. Estas son razones por las cuales los problemas de la minería de texto requieren el agrupamiento de documentos. Debido a que el problema del agrupamiento no ha sido aún resuelto; continuamente emergen nuevos algoritmos de agrupamiento efectivos, pero con enfoques que en su gran mayoría no abordan el procesamiento de documentos semiestructurados y que típicamente no tienen una complejidad lineal como refiere (Chen and Liu, 2004). No obstante, nuevos enfoques que abordan el agrupamiento de documentos XML basados en estructura y contenido, como el referido por (Tran et al., 2008a), tienen como desventaja su alto costo computacional, al proponer el uso del *Latent Semantic Kernel* (Cristianini et al., 2002) para determinar la similitud entre el contenido de los documentos.

El Algoritmo 1, propuesto en esta investigación para el agrupamiento de documentos XML, basado en la relación estructura-contenido tiene una complejidad computacional aceptable. En su cálculo se considera: *k* número de grupos, *n* número de documentos de la colección y *m*

número de rasgos. La estimación del umbral de similitud y la determinación de los núcleos iniciales del agrupamiento, mediante el cálculo de la máxima similitud entre los objetos no asignados, tiene en el peor de los casos complejidad $O(nlogn)$. La complejidad de asociar los objetos a un grupo determinado asume la complejidad del algoritmo *K*-Star, que en el peor de los casos tiene complejidad $O(kn^2)$ (Shin and Han, 2003). El cálculo de la similitud *OverallSimSUX* tiene una complejidad computacional $O(mn^2)$ considerando que este depende de la complejidad de obtener la matriz de similitud coseno y de los agrupamientos simples. Finalmente la complejidad computacional del Algoritmo 1 es $O(mn^2)$, pues en el contexto de agrupamiento de documentos *m* es mayor que *k.*

2.6 Diseño del Sistema LucXML

En este trabajo se propone el Sistema para el agrupamiento de artículos científicos en formato XML usando *Lucene* (LucXML), el cual implementa el procedimiento general basado en el modelo para el agrupamiento propuesto.

El diseño del sistema LucXML se dividió en tres capas fundamentales como se muestra en la Figura 2.5. La primera capa o inferior es la capa del dominio, la segunda o intermedia es la capa controladora y la tercera o superior es la capa de interfaz de usuario.

Figura 2.5 Diseño general del sistema LucXML.

En la capa inferior están las clases del dominio, agrupadas en dos tipos de clases diferentes: en el primer tipo están aquellas clases que permiten la representación y manipulación de los datos (ej. el analizador, la representación VSM, la manipulación de los documentos XML); el segundo tipo incluye las clases correspondientes al algoritmo de agrupamiento que operan sobre estos datos (ej. el algoritmo de agrupamiento, las medidas para la evaluación de los resultados del agrupamiento, reductores de dimensionalidad, los algoritmos para normalizar y

pesar la representación VSM). Por otra parte, la tercera capa es la encargada de la interfaz visual y contiene todas las clases relacionadas con las formas visuales y la interacción con el usuario. La capa intermedia es la que empaqueta todas las clases controladoras y es la encargada de establecer la comunicación entre las clases de las dos capas mencionadas. Observe anexos 3-8, donde se modelan las clases que intervienen en los procesos descritos en cada uno de los módulos del modelo de agrupamiento para documentos XML presentado.

2.7 Conclusiones parciales

La metodología propuesta muestra que el agrupamiento, y su evaluación, permiten la manipulación de documentos XML, y con ello, contribuyen a la gestión de información y conocimiento.

Se ha presentado una nueva medida de similitud *OverallSimSUX* que facilita evaluar el grado de similitud de dos documentos XML a nivel de unidades estructurales.

Se implementó un nuevo algoritmo de agrupamiento de documentos XML, que utiliza las dimensiones estructura y contenido, reportada en la literatura como la variante de mejores resultados para el agrupamiento de documentos semiestructurados.

3

EVALUACIÓN DEL MODELO DE AGRUPAMIENTO Y DESCRIPCIÓN A NIVEL DE USUARIO DEL SISTEMA LUCXML

3. EVALUACIÓN DEL MODELO DE AGRUPAMIENTO Y DESCRIPCIÓN A NIVEL DE USUARIO DEL SISTEMA LUCXML

La evaluación de los resultados de un agrupamiento es una tarea ardua; debido a que "El agrupamiento es un proceso subjetivo; el mismo conjunto de datos usualmente necesita ser agrupado de formas diferentes dependiendo de las aplicaciones" (Jain et al., 1999). En este capítulo, se presentan los resultados de los experimentos diseñados para evaluar el modelo de agrupamiento propuesto en esta investigación. Además se realiza una descripción del sistema a nivel de usuario con el propósito de explicar cómo utilizar LucXML para el agrupamiento de artículos científicos en formato XML, recuperados usando *Lucene*.

3.1 Evaluación de los resultados del modelo de agrupamiento de documentos XML

Para chequear la validez de los resultados obtenidos a partir del modelo de agrupamiento propuesto, se han diseñado dos experimentos, aplicados a tres casos de estudio; con el propósito de realizar un análisis estadístico, que permita verificar si existen diferencias significativas entre la metodología propuesta y otras variantes de algoritmos reportados en la literatura. La evaluación incluye la verificación y validación del modelo. Se debe verificar que el sistema está correctamente construido y que efectivamente es el producto que satisface los requerimientos.

3.1.1 Definición de los casos de estudio para la aplicación del modelo de agrupamiento de documentos XML a través de LucXML

En el CEI-UCLV existe un gran número de artículos científicos y documentos relacionados con diversos temas de investigación, disponibles para la red del Ministerio de Educación Superior (MES). Este repositorio de información es visitado con frecuencia por los investigadores, profesores y estudiantes del centro con el propósito de seleccionar documentos relacionados con algún tema en específico o descubrir conocimiento entre los mismos, cuando comienzan la revisión del estado del arte en un área específica. Teniendo en cuenta estos antecedentes se decide conformar el primer caso de estudio a partir de archivos provenientes

del sitio ICT[14], para comprobar las bondades de la nueva metodología a la recuperación de información y extracción de conocimiento que solicitan estos usuarios.

El segundo caso de estudio definido constituye una recopilación de documentos del repositorio IDE-Alliance, internacionalmente utilizados para evaluar el agrupamiento. Proporcionados por la Universidad de Granada, España.

Entre los corpus textuales publicados en Internet que se referencian en los artículos para evaluar algoritmos en el área de la minería de textos aplicados a los documentos XML, se destacan los experimentos que utilizan documentos de la colección de la Wikipedia, según se expone en (Denoyer and Gallinari, 2009) y (Campos et al., 2009), los que son publicados cada año por la **IN**iciativa para la **E**valuación de la recuperación de documentos **X**ML (INEX), entre otros. El tercer caso de estudio constituye una selección aleatoria de estos artículos, debido a que la colección contiene documentos clasificados en categorías y éstas a su vez se asocian a temas de diferentes áreas. Esta colección tiene el problema que los textos contienen mucha información no útil y el formato en que se presentan es muy difícil de preprocesar.

En la Tabla A12.1 se muestra la descripción y la fuente de cada uno de los archivos de datos que conforman los casos de estudio antes mencionados. Todos los conjuntos de datos constan de un rasgo objetivo, por tanto existe la clasificación de referencia para cada uno de ellos, en específico para el primer caso de estudio este rasgo se obtuvo basado en el criterio de expertos. Las colecciones restantes fueron adquiridas con la clasificación de referencia.

3.1.2 Validación del agrupamiento

La validación del agrupamiento se conoce por el procedimiento de evaluar los resultados de algoritmos de agrupamiento (Theodoridis and Koutroubas, 1999, Halkidi et al., 2002). Se dice medida de validación de grupos a una función que hace corresponder un número real a un agrupamiento, indicando en qué grado el agrupamiento es correcto o no (Höppner et al., 1999). Estas medidas en su mayoría son heurísticas por naturaleza para conservar una complejidad computacional plausible.

Atendiendo a la clasificación de las medidas para la evaluación del agrupamiento de (Höppner et al., 1999, Silberschatz and Tuzhilin, 1996, Kaufman and Rousseeuw, 1990), en esta

[14] ftp://ict.cei.uclv.edu.cu

investigación se seleccionaron medidas internas: *Overall Similarity* y los índices *Dunn*, externa: *Overall F-measure* y el cálculo de medidas propuestas por INEX: *Purity*, *Micro-Purity* y *Macro-Purity*. En los Anexo 10 y Anexo 11 se puede observar: un esquema de esta clasificación y cada una de las expresiones correspondientes a las medidas seleccionadas para la evaluación respectivamente.

Las medidas externas fueron seleccionadas para el estudio comparativo que se realiza, debido a que describen la calidad del resultado completo del agrupamiento usando un único valor real y se basan en una estructura previamente especificada que refleja la intuición que se tiene del agrupamiento de los datos (i.e. clasificación de referencia). *Overall F-measure* (Steinbach et al., 2000a) como medida externa, utiliza los criterios de Precisión (Pr) y cubrimiento[15] (Re) (Frakes and Baeza-Yates, 1992), que se calculan para un grupo *j* y una clase *i* dados, usando las expresiones $Pr(i,j)=n_{ij}/n_j$ y $Re(i,j)=n_{ij}/n_i$, respectivamente; donde n_{ij} es el número de objetos de la clase *i* en el grupo *j*, n_j es el número de objetos del grupo *j* y n_i es el número de objetos de la clase *i*. La medida-*F* (*F*-measure) se obtiene calculando la media armónica de precisión y cubrimiento. Se puede variar el umbral α $(0 \leq \alpha \leq 1)$ para regular la influencia de precisión y cubrimiento en el cálculo de esta medida (Frakes and Baeza-Yates, 1992). Se utiliza α=0.5, para lograr una equidad en la importancia de estos criterios. Finalmente el valor global de la medida-*F* (Overall *F*-measure; OFM), se calcula usando el promedio ponderado de los valores máximos por clase de la medida-*F* sobre todos los grupos (Steinbach et al., 2000a).

La medida OFM fue seleccionada pues logra capturar de forma eficiente la correspondencia entre los resultados del agrupamiento con las clases de tomadas como referencia (Rosell et al., 2004a).

Por otra parte, las medidas internas evalúan considerando solamente los resultados del agrupamiento en términos de cantidades que involucran los vectores de datos. Con este fin, se utilizaron las medidas internas: *Overall Similarity* (Steinbach et al., 2000a) y los índices *Dunn* (Dunn, 1974). La medida interna nombrada similitud global (*Overall Similarity*; OS) se utiliza para medir la cohesión basándose en la media de la similitud de los pares de objetos en un grupo (Steinbach et al., 2000a). Los índices se utilizan para evaluar particiones y estiman cuán

[15] En este documento se utiliza cubrimiento como traducción de la medida recall. Adicionalmente, se utiliza el término cubrimiento para nombrar una forma de la división de los objetos después de un agrupamiento duro y con solapamiento.

compactos y bien separados están los grupos, entre ellos los índices Dunn (Dunn, 1974) Así, esta medida tienden a producir valores elevados en agrupamientos con grupos compactos y muy bien separados (Bezdek and Pal, 1995).

Por último, se incluye el cálculo de medidas basadas en *Purity*: *Micro-Purity* y *Macro-Purity*. El criterio *Purity* es utilizado para determinar la calidad de los grupos del agrupamiento, se basa en la idea de maximizar su valor, para lo cual se desea que todos los elementos del grupo pertenezcan a una sola clase. *Purity* es una medida del mayor número de documentos con la misma etiqueta clase en el grupo, respecto al total de documentos. *Micro-Purity* y *Macro-Purity* se calculan para la solución completa del agrupamiento según se muestra en el Anexo 11. En general, en (Pinto et al., 2009) consideran que mayores valores de *Purity*, reportan mejores resultados del agrupamiento.

3.1.3 Verificación de los resultados

Verificar los resultados pretende asegurarse que el sistema sea consistente y correcto en cuanto a sintaxis. Para comprobar los resultados del modelo propuesto se diseñaron dos experimentos. En ambos se usa el criterio para el cálculo del umbral, basado en la media de las similitudes y se tienen en cuenta todas las unidades estructurales de las colecciones de documentos para el análisis.

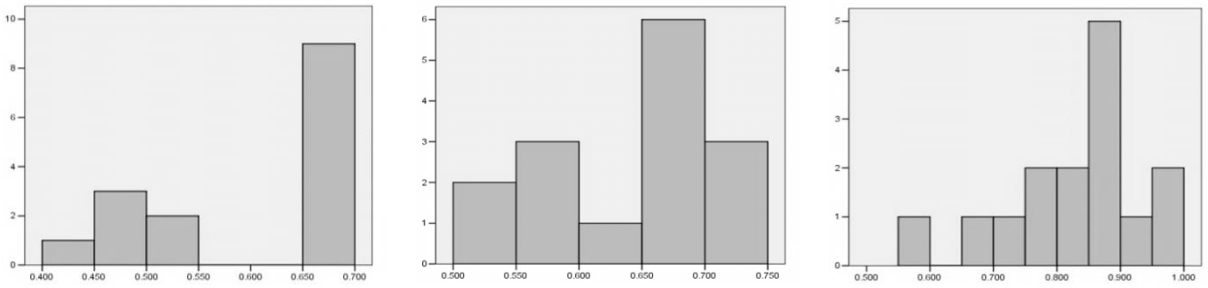
Criterios de selección del umbral

Las variaciones en el umbral de similitud permiten restringir o no el conjunto de objetos más representativos para caracterizar los grupos. En la literatura han sido propuestos diversos criterios que pueden variar los resultados del agrupamiento.

Una de estas variantes es la media de las similitudes entre todos los pares posibles de objetos. También se puede utilizar la media de los valores máximos de las similitudes entre cualquier par de objetos. Esta forma de cálculo puede provocar la obtención de un umbral muy alto, conduciendo a que exista un número mayor de grupos, al proponer criterios más restrictivos para la pertenencia al grupo. Esta situación puede arrojar valores de precisión y calidad cercanos a uno, cuando en realidad el resultado del agrupamiento no sea tan bueno. Por el contrario, la media de los valores mínimos de las similitudes entre cualquier par de objetos permite obtener umbrales de similitud muy bajos. De esta forma, el criterio de pertenencia al

grupo será mucho más flexible. Esto provoca que se obtengan valores muy bajos de precisión y calidad cuando en realidad el resultado del agrupamiento no sea de tan baja calidad.

Los archivos recopilados en los dos primeros casos de estudio fueron utilizados para comparar la calidad de los resultados del agrupamiento utilizando los criterios: Media de los máximos, Media de los mínimos y Media de todos los valores de similitud. En la Figura 3.1 se observan histogramas de frecuencias diferentes atendiendo al criterio para el cálculo del umbral.



(a) (b) (c)

Figura 3.1 Histograma de frecuencias de calidad del agrupamiento basadas en OFM. Procedentes de los casos de estudio 1 y 2, según el criterio de selección para el cálculo del umbral: (a) media de los valores máximos, (b) media de los valores mínimos y (c) media de las similitudes entre todos los pares de objetos posibles.

Para realizar un análisis comparativo de los valores de la medida OFM, aplicada al resultado de los agrupamientos, utilizando cada uno de los criterios; se escoge la prueba no paramétrica de Wilcoxon[16]. En el Anexo 13 se muestran los valores de significación de esta prueba, que refleja en todos los casos valores de significación siempre inferiores a 0.05.

Esto indica que existen diferencias significativas entre las poblaciones comparadas horizontalmente (medida OFM utilizando los criterios máximo y media de las similitudes, mínimo de las similitudes y media de todos los valores posibles, máximo de las similitudes y media de todos los valores de similitud posibles). Es importante señalar que siempre el criterio basado en el cálculo de las medias de las similitudes, aporta resultados positivos altamente significativos de calidad del agrupamiento. Por tanto, en los experimentos realizados como parte de esta investigación se ha utilizado la media de las similitudes, como criterio para el cálculo del umbral.

[16] Se utilizó SPSS 13.0 para Windows

3.1.4 Diseño de los experimentos

El primer experimento consistió en verificar cómo se comporta globalmente, sobre los tres casos de estudio descritos previamente, una de las variantes ampliamente utilizadas para el agrupamiento de documentos XML, propuesto en INEX por (Pinto et al., 2009) y el algoritmo *K*-Star por ser un antecesor de la propuesta que se hace en esta investigación. Finalmente se mostrará el estudio comparativo con el Algoritmo 1 propuesto.

En el análisis se hace referencia a la propuesta de (Pinto et al., 2009) a través las siglas INEX*K*-Star.

Tanto para la aplicación de una u otra variante (*K*-Star e INEX*K*-Star) utilizadas para comparar los resultados del agrupamiento con el Algoritmo 1; fue necesario preprocesar los corpus textuales, asociados a los tres casos de estudio que se muestran en el Anexo 12. Los experimentos realizados en esta investigación incluyeron en la transformación del corpus las operaciones siguientes: convertir todos los caracteres a mayúscula, la sustitución de las contracciones por sus expansiones, de las abreviaturas por sus formas completas y la eliminación de números y símbolos y la segmentación por eliminación de afijos, basada en el método heurístico de Porter. Las formas de pesado se basan en la fórmula TF-IDF. La idea de una expresión TF-IDF es que el peso de los términos refleje la importancia relativa de un término en un documento con respecto a los otros términos en el documento. La reducción de la dimensionalidad del espacio de rasgos haciendo corresponder palabras morfológicamente similares con la palabra raíz asociada (Frakes and Baeza-Yates, 1992, Porter, 1980) y la selección de aquellos 600 mejores términos, es decir, con calidad superior a determinado umbral considerando esencialmente las expresiones I y II de calidad de términos (Berry, 2004).

Para aplicar el Algoritmo 1 se siguió la metodología propuesta en esta investigación para el agrupamiento de documentos XML. Lo anterior, incluyó: identificar en cada documento las UE; tratadas como colecciones diferentes. Para cada UE se obtuvo una representación (*Representación I*) basada en la Representación VSM clásica; se obtuvo además una representación global (*Representación II)* que tiene en cuenta el contenido en función de la estructura.

Como entrada el Algoritmo 1 tomó los agrupamientos realizados por cada UE, a partir de la matriz de similitud coseno resultante de la *Representación I* y la matriz de similitud basada en el cálculo de la similitud coseno, a partir de la *Representación II*.

Los documentos pertenecientes a cada una de las colecciones que se utilizan en los experimentos están etiquetados y se hace uso de esa clasificación para comparar los resultados del agrupamiento respecto a la clasificación de referencia. Por eso, como criterios para la validación de los resultados de los algoritmos de agrupamiento que se comparan en el experimento 1 se utilizaron las medidas externas precisión, cubrimiento y OFM.

Para el algoritmo propuesto en (Pinto et al., 2009) que se analizó en el estudio comparativo; se utilizó la configuración más óptima para el desempeño del algoritmo, garantizando que los valores obtenidos para las medidas Pr, Re y OFM sean los mejores posibles.

Para verificar que existen diferencias significativas entre el Algoritmo 1, propuesto en esta investigación y las variantes restantes, se emplearon los resultados obtenidos por la medida de evaluación OFM, que se muestran en la Tabla A14.1. La prueba no paramétrica de Wilcoxon fue aplicada a partir de los valores registrados en la Tabla A14.1 del Algoritmo 1 propuesto y la variante original basada en *K*-Star. La Tabla A14.2 del Anexo 14 contiene los valores de significación de esta prueba, que refleja valores de significación inferiores a 0.05. Esto indica que existen diferencias significativas entre las poblaciones comparadas horizontalmente (medida OFM para el Algoritmo 1 y medida OFM para el algoritmo *K*-Star). Es importante señalar que en 15 de los 16 casos utilizados en la prueba el Algoritmo 1 se comporta mejor, aportando resultados altamente significativos con relación a la calidad del agrupamiento que reporta la medida OFM, basada en precisión y cubrimiento con factor α=0.5.

En Tabla A14.3 se muestran los valores de significación de esta prueba no paramétrica ahora aplicada utilizando los valores de OFM en la Tabla A14.1 para el Algoritmo 1 y el algoritmo INEX*K*-Star. Se observan valores de significación inferiores a 0.05 que indican las diferencias significativas existentes entre ambos algoritmos. De ellos el Algoritmo 1, aporta resultados altamente significativos respecto a la calidad del agrupamiento reportado por la medida OFM.

Las estadísticas descriptivas de los valores de la Tabla A14.1 muestran el valor promedio de OFM para *K*-Star (0.678). Este valor indica que los resultados de *K*-Star distan de la clasificación de referencia; no obstante, este algoritmo alcanzó para el corpus 12 el valor

0.923. Se observa en esta tabla que el algoritmo INEX*K*-Star tiene un comportamiento similar con valores OFM promedio (0.679) aun cuando su mejor valor es 0.819 en el corpus 13. Es importante señalar que en general el Algoritmo 1 siempre se comporta mejor que el resto de las propuestas, que aquí se analizan, alcanzando sus mejores resultados también para los corpus 12, 13 y 14 con valores OFM 0.947, 0.977 y 0.966 respectivamente. Se destaca que los buenos resultados que alcanza la medida externa OFM, se corresponden con los valores deseados para las medidas internas, ya que se obtienen valores cercanos a cero para *Overall Similarity*, que indican la buena calidad de los grupos obtenidos y los valores Dunn para los grupos asociados a cada uno de estos corpus son cercanos a uno, lo que refleja que existe separabilidad entre los grupos, obtenidos por el agrupamiento. En la Figura 3.2 se puede observar valores superiores de OFM para la metodología propuesta en la mayoría de los casos analizados.

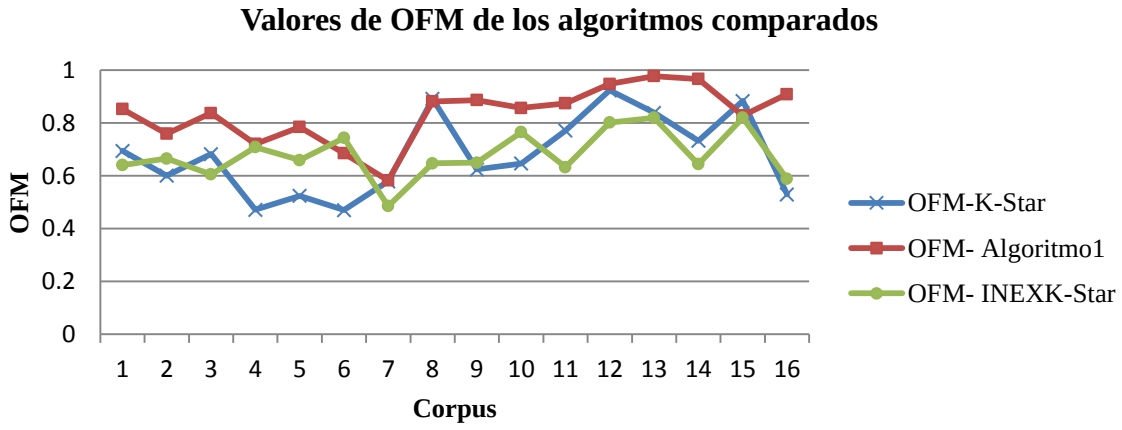


Figura 3.2. Valores de OFM de los algoritmos utilizados en el experimento 1.

En el segundo experimento se buscar verificar cómo se comporta globalmente, sobre los tres casos de estudio descritos previamente, la propuesta de INEX (Pinto et al., 2009) y el Algoritmo 1, a través de un estudio comparativo, basado en las medidas *Micro-Purity* y *Macro-Purity* que utilizan en (Vries et al., 2011) para mostrar la calidad de los grupos obtenidas en cada agrupamiento.

Las Figuras 3.3 y 3.4 muestran valores superiores de *Micro-Purity* y *Macro-Purity* para la metodología propuesta en la mayoría de los casos analizados, respectivamente.

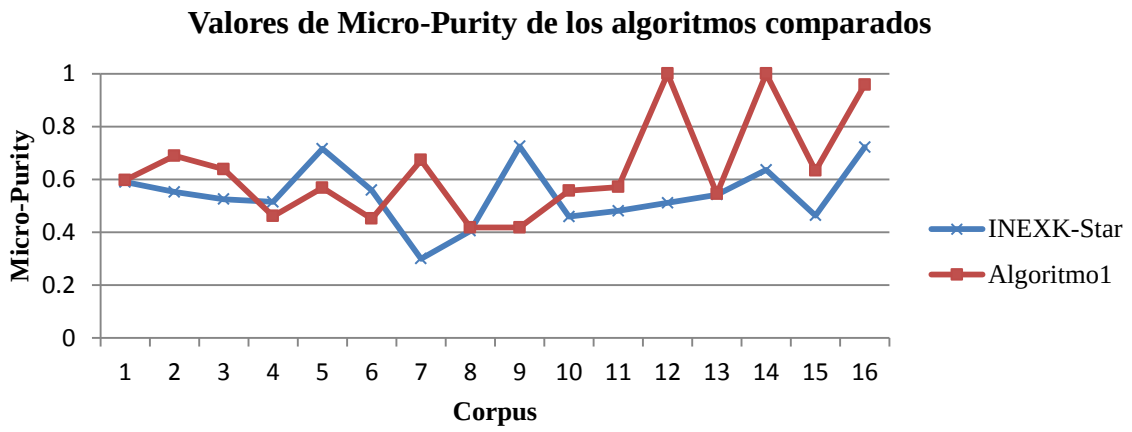


Figura 3.3. Valores de Micro-Purity de los algoritmos utilizados en el experimento 2.

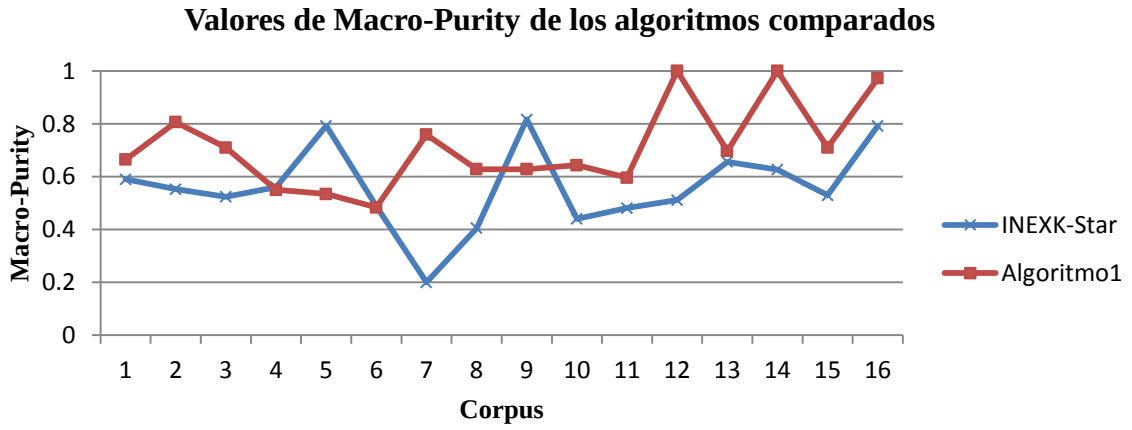


Figura 3.4. Valores de Macro-Purity de los algoritmos utilizados en el experimento 2.

La prueba no paramétrica de Wilcoxon fue aplicada a partir de los valores de *Micro-Purity* registrados en la Tabla A15.1 para el Algoritmo 1 propuesto y el algoritmo INEX*K*-Star. Reflejándose diferencias significativas en la calidad de los grupos (*Purity*) obtenida con ambos algoritmos. Los valores de significación indican que en 12 de los 16 casos utilizados en la prueba, el Algoritmo 1 se comporta mejor, aportando resultados significativos con relación a la calidad del agrupamiento que reporta la medida *Micro-Purity*, alcanzando incluso para los corpus 12, 14 y 16 valores de 1.0, 1.0 y 0.958 respectivamente. Ver Tabla A15.2.

Por otra parte, los valores de significación de la prueba de Wilcoxon que muestra la Tabla A15.3 asociada a los valores de *Macro-Purity*, refleja valores de significación inferiores a

0.05. Esto indica que existen diferencias significativas entre las poblaciones comparadas horizontalmente. El Algoritmo 1 logra comportarse mejor que la variante INEX*K*-Star y alcanza sus mejores valores para los corpus 12, 14 y 16 con *Macro-Purity* óptimas (1.0, 1.0 y 0.972 respectivamente).

3.2 Interfaz de usuarios de LucXML para la recuperación, indexación y agrupamiento de documentos XML

En este epígrafe se describe cómo utilizar en LucXML las opciones asociadas al procedimiento general para el agrupamiento de documentos XML. Se hará énfasis en las opciones de configuración del algoritmo y las bondades para la recuperación de documentos a partir de las opciones de búsqueda que provee el sistema.

3.2.1 ¿Cómo indexar colecciones de documentos XML?

LucXML permite indexar colecciones de documentos XML para su posterior recuperación, durante este proceso el sistema utiliza las facilidades del API *Jdom* de Java para extraer las unidades estructuradas seleccionadas por el usuario, que se le suministran a *Lucene* para crear los índices. El sistema permite decidir crear un nuevo índice o seleccionar uno existente, resultado de un procesamiento anterior, a través de la opción *Load Index*. La Figura 3.5 muestra el orden de ejecución del sistema para crear un nuevo índice a partir de una colección personal de documentos XML.

3.2.2 ¿Cómo configurar el agrupamiento de documentos XML?

La implementación modular utilizada en LucXML favorece la incorporación de nuevos algoritmos de agrupamiento a la metodología sin cambios perceptibles, pues basta con cambiar el método de agrupamiento a utilizar, por defecto asume el Algoritmo 1, que constituye la novedad de esta investigación. Fueron incorporados al sistema otros dos algoritmos, ampliamente utilizados para el agrupamiento: *K*-Star e INEX*K*-Star, con el propósito de facilitar la validación y verificación de la nueva metodología. En el diálogo de la Figura 3.6. LucXML da la opción de elegir cual variante utilizar.

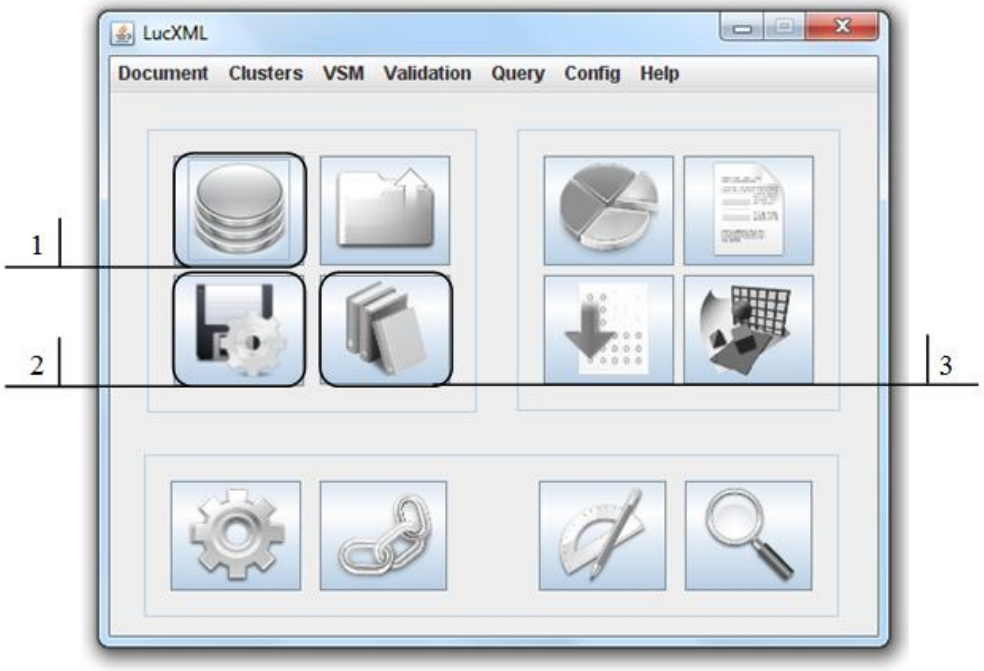


Figura 3.5. Ventana principal, donde: (1). Seleccionar la colección (2). Seleccionar el directorio para salvar el índice a crear (3) Indexar.

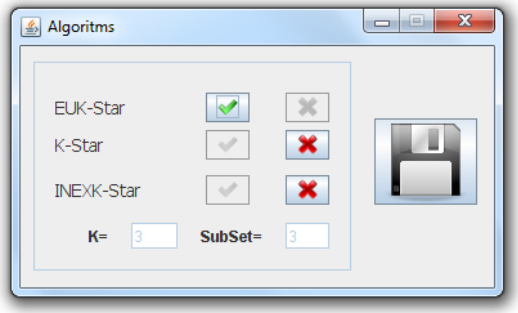


Figura 3.6. Seleccionar Algoritmo de agrupamiento, donde: EUK-Star, Nuevo Modelo para el Agrupamiento de documentos XML, propuesta de esta tesis; *K*-Star, agrupamiento basado en *K*-Star; INEXK-Star Propuesta de INEX para el agrupamiento de documentos XML.

El Algoritmo 1, incluye el cálculo de la matriz de similitud global a partir de la función *OverallSimSUX*. El sistema permite modificar los valores asociados a los pesos de cada unidad estructural, que utiliza esta función para el agrupamiento. No obstante, se recomienda, utilizar los propuestos por defecto obtenidos con la expresión encargada de calcular el peso de las UE (Ver epígrafe 2.1.1.2). Esto es posible de la forma que se muestra en la Figura 3.7.

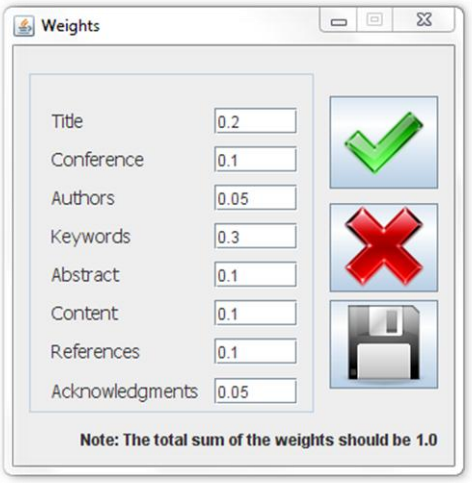


Figura 3.7. Opción que permite modificar los pesos asociados a cada unidad estructural, utilizados en la función *OverallSimSUX*.

El modelo para el agrupamiento de documentos XML propuesto, combina las dimensiones estructura y contenido. En el módulo 2 del procedimiento general que soporta esta metodología, cada UE es tratada como una colección independiente (se obtiene una *Representación I* por cada UE), resultados que se combinan a partir del cálculo de la similitud *OverallSimSUX*. Los agrupamientos asociados a cada *Representación I*, son la entrada del Algoritmo 1 que utiliza el sistema para el agrupamiento. LucXML por defecto considera todas las unidades estructurales en el procesamiento aunque permite modificar este criterio, a partir del diálogo que se muestra en la Figura 3.8.

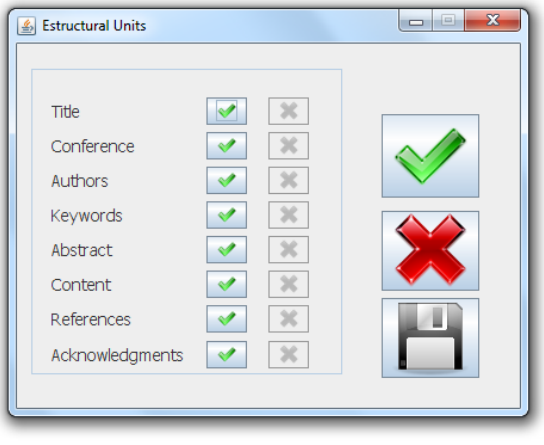


Figura 3.8. Opción para seleccionar las unidades estructurales que se consideran en el agrupamiento.

Los criterios para el cálculo del umbral que utilizan los algoritmos de agrupamiento en LucXML pueden modificarse, según lo muestra la Figura 3.9. Este diálogo además permite,

indicar si los resultados del agrupamiento desean solo visualizarse o realizar este proceso de forma física.

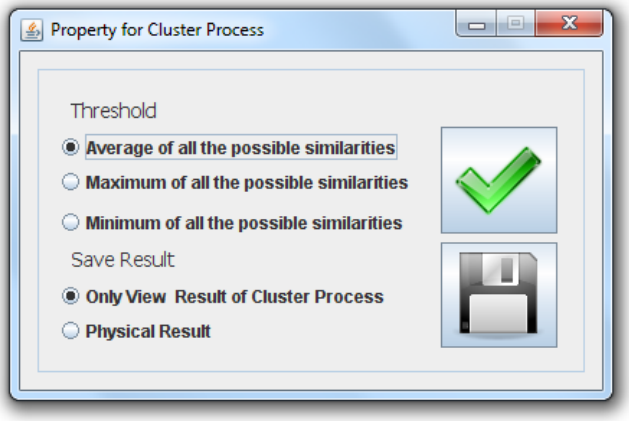


Figura 3.9. Opción para seleccionar el criterio para el cálculo del umbral y la forma de presentación de los resultados del agrupamiento.

3.2.3 ¿Cómo agrupar una colección de documentos XML y validar los resultados del agrupamiento?

La Figura 3.10 muestra el orden de ejecución del sistema para obtener un agrupamiento basado en el modelo propuesto y su evaluación a partir de una colección personal de documentos XML.

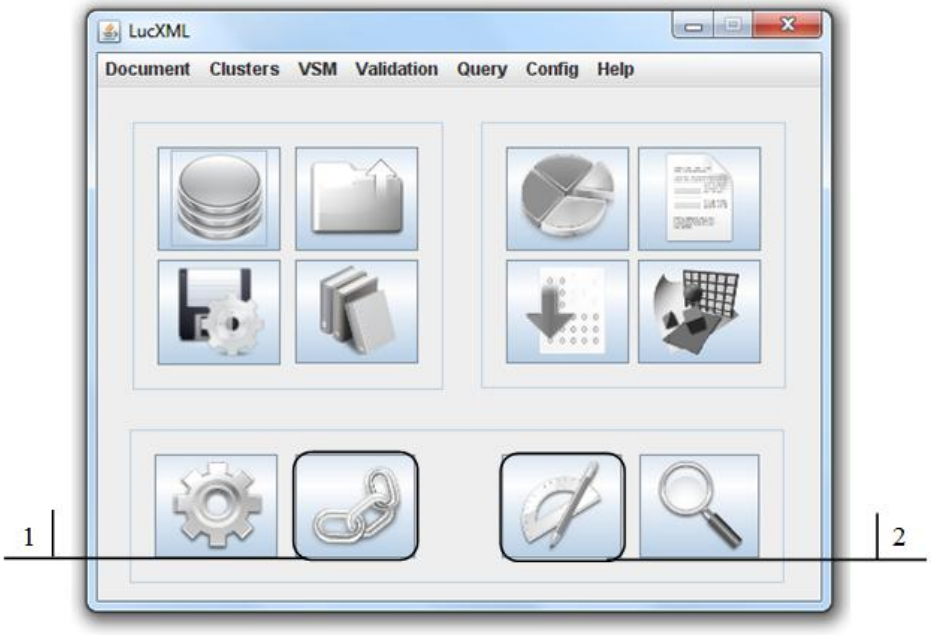


Figura 3.10. Ventana principal, donde: (1). Ejecutar el agrupamiento (2). Validación del agrupamiento.

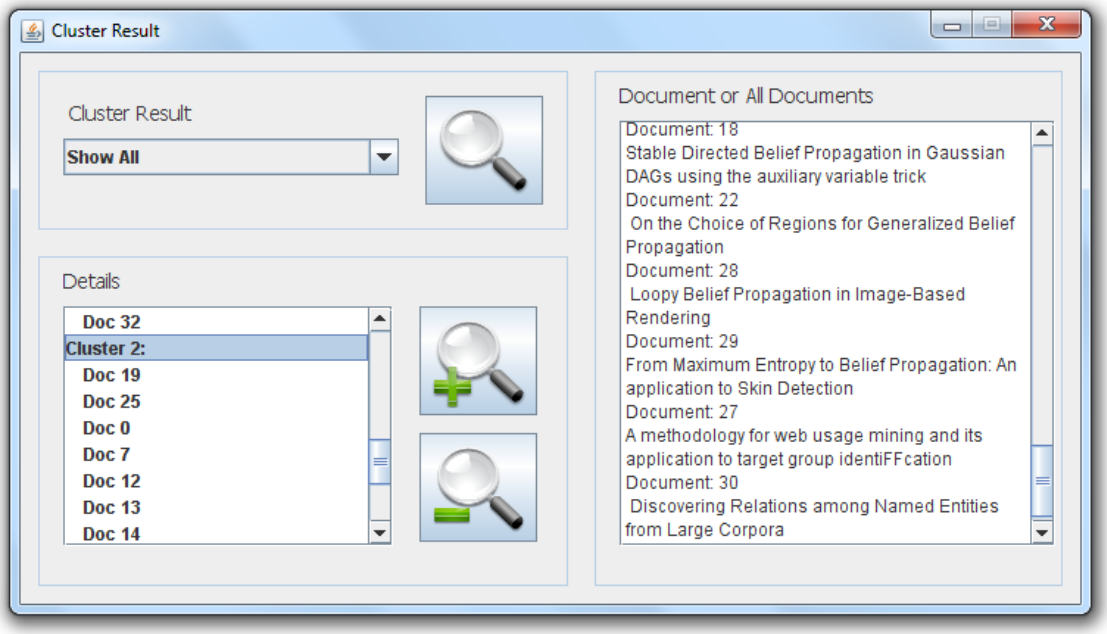


Figura 3.11. Ventana con los resultados del agrupamiento.

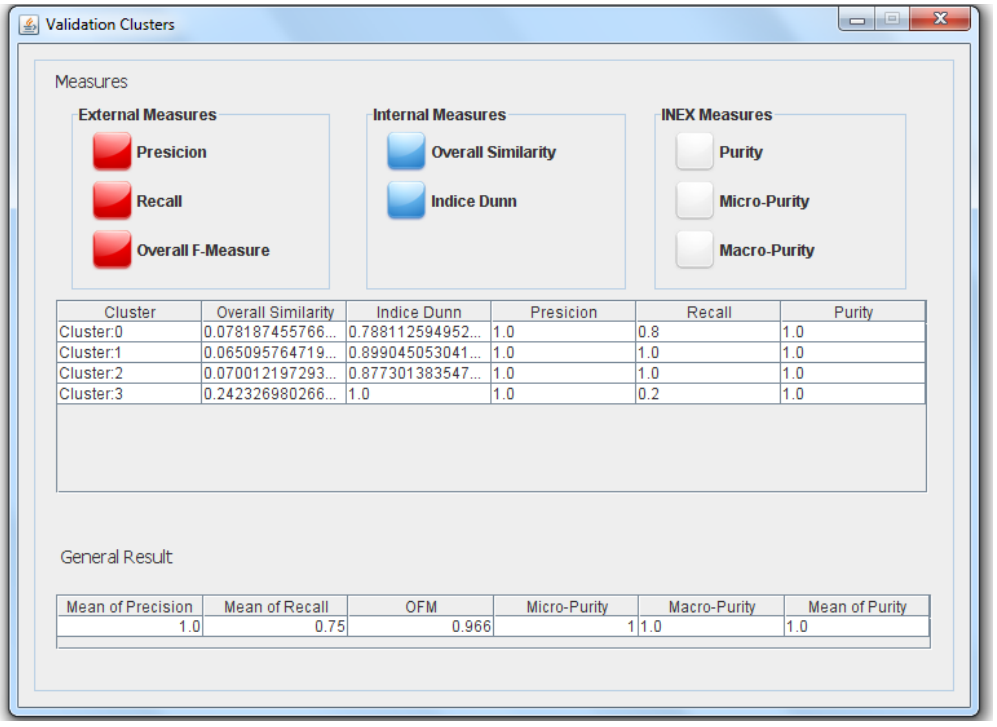


Figura 3.12. Ventana con los resultados de las medidas de evaluación del agrupamiento.

3.2.4 ¿Cómo realizar búsquedas a partir una colección de documentos?

LucXML facilita el proceso de búsqueda y consulta de la colección, permitiendo organizar los resultados por la relevancia y el criterio del agrupamiento. El sistema admite realizar búsquedas por múltiples campos, al estilo *Lucene*. En este contexto se establece una asociación entre campo y unidad estructural de un documento XML. La Figura 3.13 muestra el diálogo que permite editar las UE que se desean incluir en el procesamiento de una consulta.

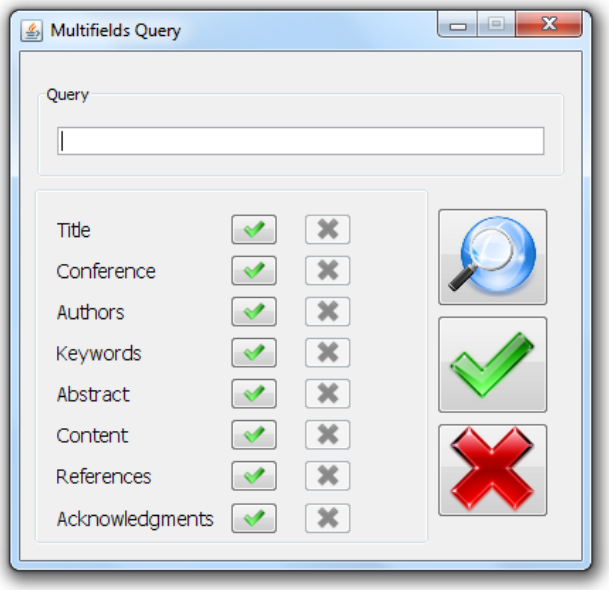


Figura 3.13. Diálogo para editar búsquedas utilizando múltiples campos (UE).

3.3 Conclusiones parciales

Los casos de estudio definidos permitieron demostrar la factibilidad del modelo propuesto para el agrupamiento de documentos XML.

Se mostró que el algoritmo presentado para agrupamiento de documentos XML tiene un buen desempeño. Los resultados fueron comparados con aquellos producidos por los algoritmos *K*-Star y la propuesta INEX*K*-Star (Pinto et al., 2009). Para los tres casos de estudios considerados, el algoritmo basado en la metodología propuesta obtuvo resultados superiores a los alcanzados por los algoritmos citados. Estos resultados se deben en gran parte a las potencialidades de la nueva medida de similitud para capturar las relaciones topológicas entre los documentos XML a nivel de las unidades estructurales que en ocasiones la similitud coseno, no permite obtener de manera precisa.

La interfaz de usuario que incluye la implementación del procedimiento general propuesto es amigable y facilita el proceso de búsqueda y consulta de la colección, permite organizar los resultados por la relevancia y el criterio del agrupamiento. La implementación modular utilizada en LucXML favorece la incorporación de nuevos algoritmos de agrupamiento a la metodología sin cambios perceptibles.

CONCLUSIONES

Como resultado de esta investigación se implementó un nuevo método de agrupamiento automático de documentos XML, utilizando el contenido y la estructura existentes en los mismos. Cumpliéndose de esta forma el objetivo general planteado, ya que:

- La variante de agrupamiento implementada es la de mejores resultados reportados.
- Se definió una nueva función de similitud denominada *OverallSimSUX* que permite capturar el grado de semejanza entre los documentos tomando como génesis la relación existente entre las unidades estructurales, cuando se manipulan como colecciones independientes y la similitud global.
- Se implementó el sistema LucXML utilizando tecnología Java; el API *Jdom*, encargada del manejo de documentos XML y la biblioteca para la recuperación de información *Lucene*. El mismo facilita el proceso de búsqueda y consulta de la colección, permite organizar los resultados por la relevancia y el criterio del agrupamiento. La implementación modular utilizada en LucXML favorece la incorporación de nuevos algoritmos de agrupamiento a la metodología sin cambios perceptibles, pues basta con cambiar el método de agrupamiento a utilizar, por defecto asume el EUK-Star.
- La evaluación a través de los experimentos y los casos de estudios definidos, utilizando el algoritmo de agrupamiento *K*-Star, arrojaron mejores resultados con la metodología propuesta, que con otras variante existentes en la literatura.

RECOMENDACIONES

Se recomienda:

- Evaluar la metodología propuesta con otras existentes en la literatura.
- Agregar otros algoritmos de agrupamiento a la metodología propuesta.

REFERENCIAS BIBLIOGRÁFICAS

ABITEBOUL, S. 1997. Querying semi-structured data. *Proceedings of the ICDT Conference, Delphi, Greece*.

ANDERBERG, M. R. 1973. *Clustering Analysis for Applications*, New York: Academic.

ARCO, L. 2009. *Agrupamiento basado en la intermediación diferencial y su valoración utilizando la teoría de los conjuntos aproximados.* Doctorado en Ciencias Técnicas, Universidad Central "Marta Abreu" de Las Villas.

BATCHELOR, B. 1978. *Pattern Recognition: Ideas in Practice,* New York, Plenum Press.

BAUMES, J., GOLDBERG, A. & MAGDON-ISMAIL, M. 2005. Efficient identification of overlapping communities. *Intelligence and Security Informatics.* Berlin: Springer Berlin / Heidelberg.

BERRY, M. W. 2004. *Survey of Text mining: Clustering, Classification, and Retrieval,* New York, USA, Springer Verlag.

BEZDEK, J. & PAL, N. Cluster validation with generalized Dunn's indices. *In:* KASABOV, N. & COGHILL, G., eds. Proceedings of the 2nd International two-stream Conference on ANNES, 1995 Piscataway, NJ. IEEE Press, 190-193.

BRUN, M., SIMA, C., HUA, J., LOWEY, J., CARROLL, B., SUH, E. & DOUGHERTY, E. R. 2007. Model-based evaluation of clustering validation measures. *Pattern Recognition,* 40**,** 807-824.

BUENO, E. Estado del arte y tendencias en creación y gestión del conocimiento. Congreso Iberoamericano de Gestión del Conocimiento y la Tecnología (IBERGECYT 2001, 2001 La Habana, Cuba.

C.D., M., RAGHAN, P. & SCHÜTZE, H. Introduction to Information Retrieval. 2008 Cambridge University Press.

CAMPOS, L. M. D., FERN´ANDEZ-LUNA, J. M. & J.F. HUETE, A. E. R. 2009. Probabilistic methods for link-based classification at INEX'08. *Proceedings of Initiative for the Evaluation of XML Retrieval*

5631**,** 453–459.

CANALS, A., BOISOT, M. & CORNELLA, A. 2003. Gestión del conocimiento. Barcelona, España: Gestión: 2000.

CHAWATHE, S. S. Comparing Hierarchical Data in External Memory. In Proceedings of International Conference on Very Large Databases, 1999. 90-101.

CHAWATHE, S. S., RAJARAMAN, A., GARCIA-MOLINA, H. & WIDOM, J. Change Detection in Hierarchically Structured Information. In Proceedings of the ACM International Conference on Management of Data, 1996. 493-504.

CHEN, K. & LIU, L. ClusterMap: labeling clusters in large datasets via visualization. Proceedings of the ACM IEEE 13th Conference on Information and Knowledge Management (CIKM 2004), 2004 Washington, D.C., 285-293.

CHENG, D., KANNAN, R., VEMPALA, S. & WANG, G. 2006. A divide-and-merge methodology for clustering. *ACM Transaction on Database Systems (TODS),* 31**,** 1499-1525.

CHENG, D., VEMPALA, S., KANNAN, R. & WANG, G. A divide-and-merge methodology for clustering. Proceedings of the 24th ACM SIGMOD SIGACT SIGART Symposium on Principles of Database Systems (PODS 2005), 2005 Baltimore, Maryland. ACM Press, 196-205.

CRISTIANINI, N., SHAWE-TAYLOR, J. & LODHI, H. 2002. Latent semantic kernels. *JJIS'2002,* 18.

DALAMAGAS, T., CHENG, T., WINKEL, K.-J. & SELLIS, T. 2006. A Methodology for Clustering XML Documents by Structure. *Information Systems*.

DALKIR, K. 2005. *Knowledge Management in Theory and Practice,* Burlington, USA, Elsevier.

DENOYER, L. & GALLINARI, P. 2009. Overview of the inex 2008 xml mining track. In Advances in Focused Retrieval. *Proceedings of Initiative for the Evaluation of XML Retrieval,* 5631**,** 401–411.

DIXON, M. 1997. An overview of document mining technology. http://www.geocities.com/ResearchTriangle/Thinktank/1997/mark/writings/dixm97_dm.ps.

DOUCET, A. & AHONENMYKA, H. 2002. Naive clustering of a large XML document collection. *INEX***,** 84-89.

DUCH, W. 2002. Similarity-based methods: a general framework for classification. *Control and Cybernetics,* 29**,** 937-968.

DUNN, J. 1974. A fuzzy relative isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics,* 3**,** 32-57.

EGÜE, M. A. 2011. *Herramientas de Minería de Textos e Inteligencia Artificial aplicadas a la gestión de la información científico-técnica.* Máster en Ciencia de la Computación, Universidad Central "Marta Abreu" de Las Villas.

FLESCA, S., MANCO, G., MASCIARI, E., PONTIERI, L. & PUGLIESE, A. 2005. Fast detection of XML structural similarities. *IEEE Trans. Knowl. Data Engin.,* 7**,** 160-175.

FRAKES, W. B. & BAEZA-YATES, R. 1992. *Information Retrieval. Data Structure & Algorithms,* New York, Prentice Hall.

GETOOR, L. & DIEHL, C. P. 2005. Link mining: a survey. *SIGKDD Exploration Newsletter,* 7**,** 3-12.

GIANNOPOULOS, P. & VELTKAMP., R. C. A Pseudo-Metric for Weighted Point Sets. In Proceedings of the 7th European Conference on Computer Vision (ECCV), 2002. 715–730.

GIRVAN, M. & NEWMAN, M. E. J. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences (PNAS USA),* 99**,** 7821-7826.

GOLDMAN, R. & WIDOM, J. DataGuides: Enabling Query Formulation and Optimization in Semistructured Databases. In Proceedings of International Conference on Very Large Databases., 1997. 436-445.

GUERRINI, G., MESITI, M. & SANZ, I. 2006. An Overview of Similarity Measures for Clustering XML Documents.

HALKIDI, M., BATISTAKIS, Y. & VAZIRGIANNIS, M. 2002. Clustering validity checking methods: Part II. *ACM SIGMOD Record,* 31**,** 19-27.

HAND, D. J. 1981. *Discrimination and classification*, John Wiley and Sons.

HATCHER, E., GOSPODNETIC, O. & MCCANDLESS, M. 2009. *Lucene in Action*.

HÖPPNER, F., KLAWONN, F., KRUSE, R. & RUNKLER, T. 1999. *Fuzzy cluster analysis: methods for classification, data analysis and image recognition.*, West Sussex, England, John Wiley & Sons Ltd.

JAIN, A. K., MURTY, M. N. & FLYNN, P. J. 1999. Data clustering: a review. *ACM Computing Surveys,* 31**,** 264-323.

KARMARKAR, N. 1984. A new polynomial-time algorithm for linear programming. *Proceedings of the 16th Annual ACM Symposium on the Theory of Computing.*

KAUFMAN, L. & ROUSSEEUW, P. J. 1990. *Finding groups in data: an introduction to cluster analysis*, John Wiley and Sons.

KIRSTEN, M. & WROBEL, S. Extending k-means clustering to first-order representations. Proceedings of the 10th International Conference on Inductive Logic Programming., 2000.

KRUSE, R., DÖRING, C. & LESOR, M.-J. 2007. Fundamentals of Fuzzy Clustering. *In:* OLIVEIRA, J. V. D. & PEDRYCZ, W. (eds.) *Advances in Fuzzy Clustering and its Applications.* Est Sussex, England: John Wiley and Sons.

KURGAN, L., SWIERCZ, W. & CIOS, K. J. Semantic mapping of xml tags using inductive machine learning. 11th International Conference on Information and Knowledge Management., 2002 Virginia, USA.

KUTTY, S., TRAN, T., NAYAK, R. & LI, Y. 2008. Combining the structure and content of XML documents for clustering using frequent subtrees. *INEX***,** 391-401.

LANQUILLON, C. 2001. *Enhancing Text Classification to Improve Information Filtering.* PhD. thesis, University of Magdeburg "Otto von Guericke".

LIAN, W., CHEUNG, D., MAMOULIS, N. & YIU, S.-M. 2004. An Efficient and Scalable Algorithm for Clustering XML Documents by Structure. *TKDEE*

MAGDALENO, D. 2008. *Refinamiento, evaluación y etiquetamiento de grupos textuales basados en la teoría de los conjuntos aproximados.* Máster en Ciencia de la Computación, Universidad Central "Marta Abreu" de Las Villas.

MAGDALENO, D., ARCO, L. & ARTILES, M. 2011a. Representación de Documentos XML, un Enfoque para el Agrupamiento, Aplicaciones en la Gestión del Conocimiento. *In:* FEIJÓ (ed.). UCLV.

MAGDALENO, D., FUENTES, I. E., ARCO, L., ARTILES, M., FERNANDEZ, J. M. & HUETE, J. 2011b. New Textual Representation using Structure and Contents. *Research in Computing Science*, 54**,** 117-130.

MARTÍN, C. 2007. *Aprendizaje Automático y Minería de Datos con Modelos Gráficos Probababilísticos.* DEA DEA, Universidad de Granada.

MATTMANN, C. A. & ZITTING, J. L. 2012. *Tika in Action,* 20 Baldwin Road

PO Box 261

Shelter Island, NY 11964, Manning Publications Co.

MICHALSKI, R. S., STEPP, R. E. & DIDAY, E. 1981. A recent advance in data analysis: clustering objects into classes characterized by conjuntive concepts. *Progress in Pattern Recognition,* 1**,** 33-56.

MLADENIC, D. & GROBELNIK, M. Feature selection for classification based on text hierarchy. Working Notes of Learning from Text and the Web: Conference on Automatic Learning and Discovery (CONALD-98), 1998 Carnegie Mellon University, Pittsburgh, PA.

NAYAK, R. Investigating Semantic Measures in XML Clustering. Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, 2006. IEEE.

NAYAK, R. & XU., S. XCLS: A Fast and Effective Clustering Algorithm for Heterogenous XML Documents. 10th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD). , 2006 Singapore. LNCS, p 292-302.

NIERMAN, A. & JAGADISH, H. V. 2002. Evaluating structural similarity in XML documents. *5th Int. Conf. Computational Science (ICCS'05).*

NIU, Z.-Y., JI, D.-H. & TAN, C.-L. Document clustering based on cluster validation. *In:* EVANS, D. A., ed. Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management (CIKM 2004), 2004 Washington, D.C., USA. ACM Press, 501-506.

PASSONI, L. 2005. Gestión del conocimiento: una aplicación en departamentos académicos. *Gestión y Política Pública,* XIV**,** 57-74.

PINTO, D., TOVAR, M. & VILARIÑO, D. BUAP: Performance of K-Star at the INEX'09 Clustering Task. *In:* GEVA, S., KAMPS, J. & TROTMAN, A., eds. INEX 2009 Workshop Pre-proceedings, 2009 Woodlands of Marburg, Ipswich, Queensland,Australia. 391-398.

PORTER, M. F. 1980. An algorithm for suffix stripping. *Program,* 14**,** 130-137.

ROSELL, M., KANN, V. & LITTON, J.-E. Comparing comparisons: document clustering evaluation using two manual classifications. *In:* SANGAL, R. & BENDRE, S. M., eds. Proceedings of the International Conference on Natural Language Processing (ICON 2004), 2004a Hyderabad, India. Allied Publishers, 207-216.

ROSELL, M., KANN, V. & LITTON, J. E. Comparing comparisons: document clustering evaluation using two manual classifications. Proceedings of International Conference on Natural Language processings ICON, 2004b Hyderabad, India.

RUIZ-SHULCLOPER, J. 1995. *Introducción al reconocimiento de patrones. Enfoque lógico combinatorio,* México, CINVESTAV IPN.

SELKOV, S. M. 1977. The Tree-to-Tree Editing Problem. *Information Processing Letters,* 6**,** 184-186.

SHEN, Y. & WANG, B. Clustering schemaless xml document. 11th international conference on Cooperative Information System., 2003.

SHIN, K. & HAN, S. Y. 2003. Fast clustering algorithm for information organization. *In:Proc. of the CICLing Conference.* Lecture Notes in Computer Science.Springer-Verlag (2003).

SILBERSCHATZ, A. & TUZHILIN, A. 1996. What makes patterns interesting in knowledge discovery systems. *IEEE Transactions on Knowledge and Data Engineering,* 8**,** 940-974.

STEINBACH, M., KARYPIS, G. & KUMAR, V. A comparison of document clustering techniques. Proceedings of 6th ACM SIGKDD World Text Mining Conference, 2000a Boston. ACM Press, 1-20.

STEINBACH, M., KARYPIS, G. & KUMAR, V. A comparison of document clustering techniques. Proceedings of 6th ACM SIGKDD World Text Mining Conference, 2000b Boston. ACM Press.

STREHL, A., GHOSH, J. & MOONEY, R. Impact of similarity measures on Web-page clustering. Proceedings of the 17th National Conference on Artificial Intelligence (AAAI 2000): Workshop of Artificial Intelligence for Web Search, 2000 Austin, Texas. 58-64.

TAN, A. Text Mining: The state of the art and the challenges. Proceedings of the Conference Knowledge Discovery and Data Mining (PAKDD'99): Workshop Knowledge Discovery from Advanced Databases, 1999 Pacific Asia. 65-70.

THEODORIDIS, S. & KOUTROUBAS, K. 1999. *Pattern Recognition*, Academic Press.

TIEN T., R. N. 2007. Evaluating the Performance of XML Document Clustering by Structure only. *5th International Workshop of the Initiative for the Evaluation of XML Retrieval*.

TONG, H. & FALOUTSOS, C. Center-piece subgraphs: problem definition and fast solutions. Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006 Philadelphia, PA, USA. ACM Press, 404-413.

TRAN, T., KUTTY, S. & NAYAK, R. 2008a. Utilizing the Structure and Data Information for XML Document Clustering. *INEX***,** 402-410.

TRAN, T., NAYAK, R. & BRUZA, P. Combining Structure and Content Similarities for XML Document Clustering. Seventh Australasian Data Mining Conference, 2008b Glenelg, Australia.

VRIES, C. M. D., NAYAK, R., KUTTY, S., GEVA, S. & TAGARELLI, A. 2011. Overview of the INEX 2010 XML mining track : clustering and classification of XML documents. *In Lecture Notes in Computer Science, Springer*. Amsterdam.

WAN, X. & YANG, J. 2006. Using Proportional Transportation Similarity with Learned Element Semantics for XML Document Clustering. *International World Wide Web Conference Committee*.

WILSON, D. R. & MARTÍNEZ, T. R. 1997. Improved heterogeneous distance functions. *Journal of Artificial Intelligence Research,* 6**,** 1-34.

XING, G., XIA, Z. & GUO, J. (eds.) 2007. *Clustering XML Documents Based on Structural Similarity, LNCS 4443, pp. 905–911*: Springer.

XIONG, H., WU, J. & CHEN, J. K-means clustering versus validation measures: a data distribution perspective. Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD 2006), 2006 Philadelphia, PA, USA. ACM Press, 779-784.

YANG, W. & CHEN, X. O. 2002. A semi-structured document model for text mining. *Journal of Computer Science and Technology*, 17**,** 603-610.

YANG, Y. & PEDERSEN, J. O. A comparative study on feature selection in text categorization. *In:* FISHER, D. H., ed. Proceedings of the Fourteenth International Conference on Machine Learning, 1997 San Francisco, US. Morgan Kaufmann Publishers, 412-420.

ZHANG, K. & SHASHA, D. 1989. Simple Fast Algorithms for the Editing Distance Between Trees and Related Problems. . *SIAM Journal of Computing*, 18**,** 1245-1262.

ANEXOS

Anexo 1. Similitudes, distancias más usadas para comparar objetos y medidas de calidad

Sean los objetos O_i y O_j descritos por m rasgos, donde $O_i=(o_{i1}, \ldots, o_{im})$ y $O_j=(o_{j1}, \ldots, o_{jm})$

Distancia Euclidiana

$$D_{Euclideana}(O_i, O_j) = \sqrt{\sum_{k=1}^{m}(o_{ik} - o_{jk})^2} \qquad \text{(A1.1)}$$

Distancia Minkowski (Batchelor, 1978)

$$D_{Minkowski}(O_i, O_j) = \left(\sum_{k=1}^{m}|o_{ik} - o_{jk}|^{\gamma}\right)^{\frac{1}{\gamma}} \quad \text{donde } \gamma \geq 1 \qquad \text{(A1.2)}$$

La distancia Minkowsky es equivalente a la distancia Manhattan o city-block,y a la distancia Euclidiana cuando γ es 1 y 2, respectivamente(Batchelor, 1978). Para los valores de $\gamma \geq 2$, la distancia Minkowsky equivale a Supermum (Hand, 1981).

Distancia Euclidiana heterogénea (HeterogenousEuclidean – OverlapMetric; HEOM)

$$D_{HEOM}(O_i, O_j) = \sqrt{\sum_{k=1}^{m} d_{local}(o_{ik}, o_{jk})^2} \text{ , donde}$$

$$d_{local}(o_{ik}, o_{jk}) = \begin{cases} d_{Overlap}(o_{ik}, o_{jk}) & si\ k\ simbólico \\ d_{NormEuclidean}(o_{ik}, o_{jk}) & si\ k\ numérico \end{cases} \qquad \text{(A1.3)}$$

$$d_{Overlap}(o_{ik}, o_{jk}) = \begin{cases} 0, & si\ o_{ik} = o_{jk} \\ 1, & en\ otro\ caso \end{cases} \quad \text{y} \quad d_{NormEuclidean}(o_{ik}, o_{jk}) = \frac{|o_{ik} - o_{jk}|}{\max_k - \min_k}$$

Distancia Camberra (Michalski et al., 1981)

$$D_{Camberra}(O_i, O_j) = \sum_{k=1}^{m} \frac{|o_{ik} - o_{jk}|}{|o_{ik} + o_{jk}|} \qquad \text{(A3.3.4)}$$

Correlación de Pearson(Wilson and Martínez, 1997)

$$D_{Pearson}(O_i, O_j) = \frac{\sum_{k=1}^{m} \left(o_{ik} - \overline{atributo_k}\right)\left(o_{jk} - \overline{atributo_k}\right)}{\sqrt{\sum_{k=1}^{m} \left(o_{ik} - \overline{atributo_k}\right)^2 \sum_{k=1}^{m} \left(o_{jk} - \overline{atributo_k}\right)^2}} \quad \text{(A1.5)}$$

donde $\overline{atributo_k}$ es el valor promedio que toma el $atributo_k$ en el conjunto de datos.

Las expresiones de Chebychev, Mahalanobis, distancia de Hamming y la máxima distancia son otras variantes de cálculo de distancias entre objetos (Wilson and Martínez, 1997). En(Duch, 2002) se presentan formas de medir la similitud considerando una relación asimétrica entre los objetos y teniendo en cuenta la probabilidad condicional de un objeto respecto al otro.

A partir de estudios realizados, se ha demostrado que al agrupar documentos, los coeficientes Dice, Jaccard y Coseno, han reportado los mejores resultados(Frakes and Baeza-Yates, 1992). Una valoración del impacto de la distancia Euclidiana y los coeficientes Dice, Jaccard y Coseno en dominios textuales se presenta en(Strehl et al., 2000).

Coeficiente Dice

$$S_{Dice}(O_i, O_j) = \frac{2\sum_{k=1}^{m} \left(o_{ik} \cdot o_{jk}\right)}{\sum_{k=1}^{m} o_{ik}^2 + \sum_{k=1}^{m} o_{jk}^2} \quad \text{(A1.6)}$$

Coeficiente de Jaccard

$$S_{Jaccad}(O_i, O_j) = \frac{\sum_{k=1}^{m} \left(o_{ik} \cdot o_{jk}\right)}{\sum_{k=1}^{m} o_{ik}^2 + \sum_{k=1}^{m} o_{jk}^2 - \sum_{k=1}^{m} \left(o_{ik} \cdot o_{jk}\right)} \quad \text{(A1.7)}$$

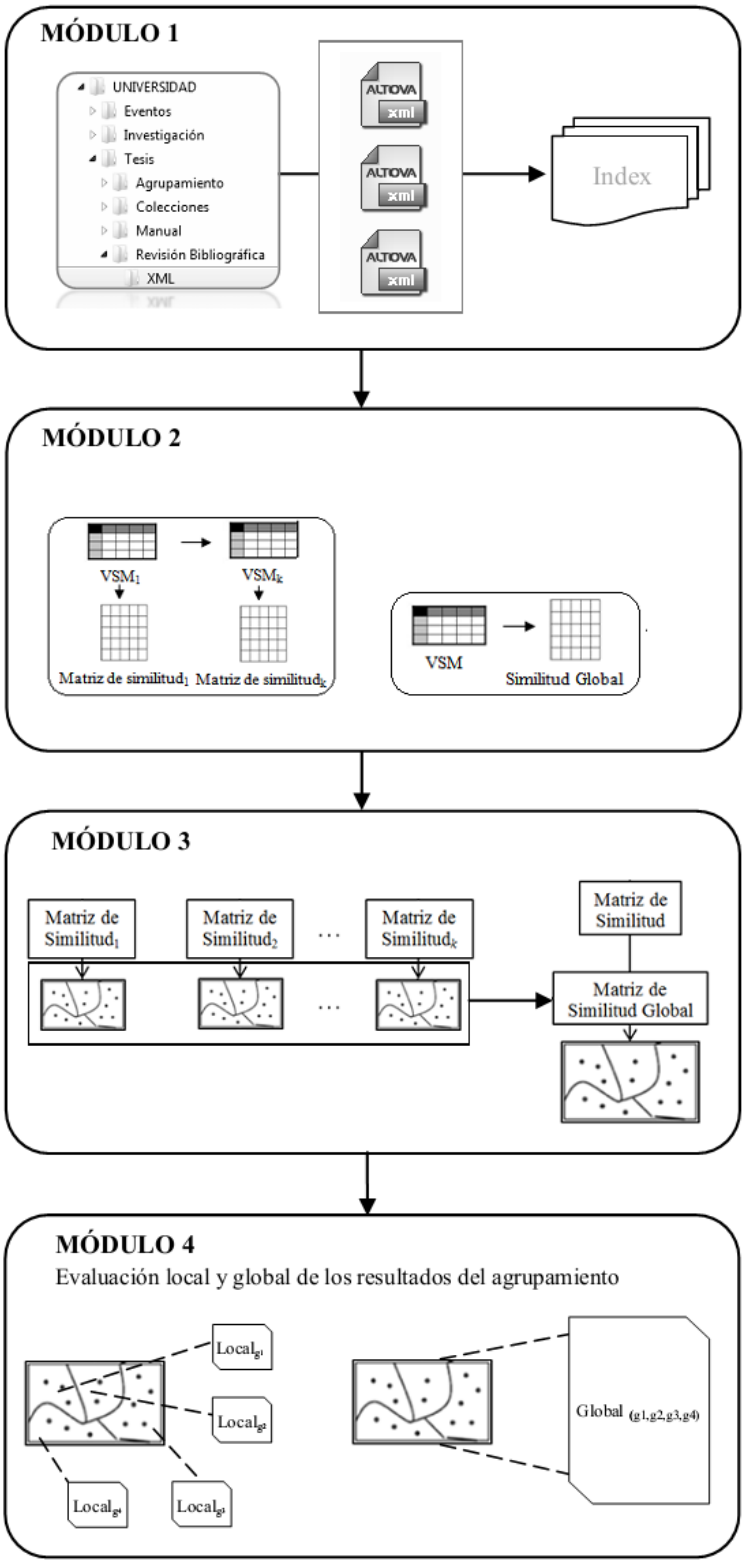
Coeficiente Coseno

$$S_{Coseno}(O_i, O_j) = \frac{\sum_{k=1}^{m} (o_{ik} \cdot o_{jk})}{\sqrt{\sum_{k=1}^{m} o_{ik}^2 \cdot \sum_{k=1}^{m} o_{jk}^2}} \qquad \text{(A1.8)}$$

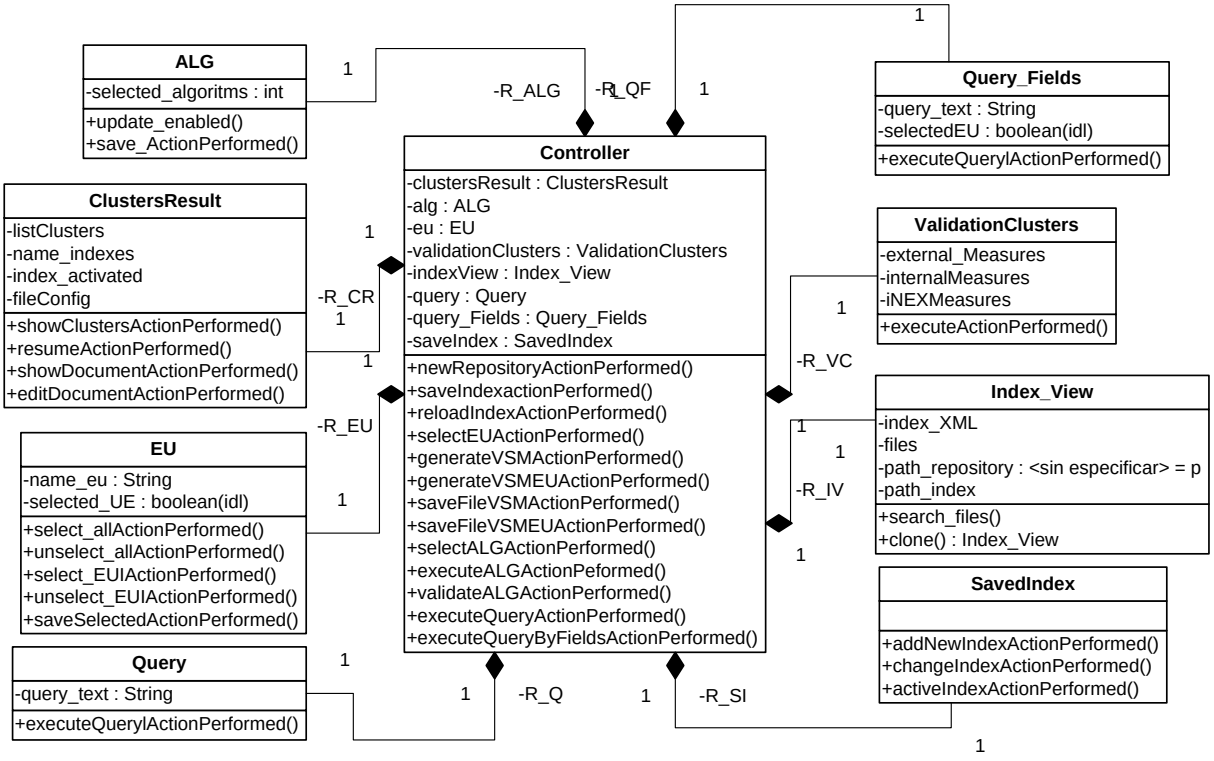
Calidad de términos. Medir la calidad de los términos según las expresiones q_0 y q_1, la segunda constituye una variante de la primera donde n_1 es el número de documentos en los cuales *t* ocurre al menos una vez (Berry, 2004).

$$q_0(t) = \sum_{j=1}^{n} (tf_{d\,j}(t))^2 - \frac{1}{n}\left[\sum_{j=1}^{n} tf_{d\,j}(t)\right]^2 \quad q_1(t) = \sum_{j=1}^{n_1} (tf_{d\,j}(t))^2 - \frac{1}{n_1}\left[\sum_{j=1}^{n_1} tf_{d\,j}(t)\right]^2 \qquad \text{(A1.9)}$$

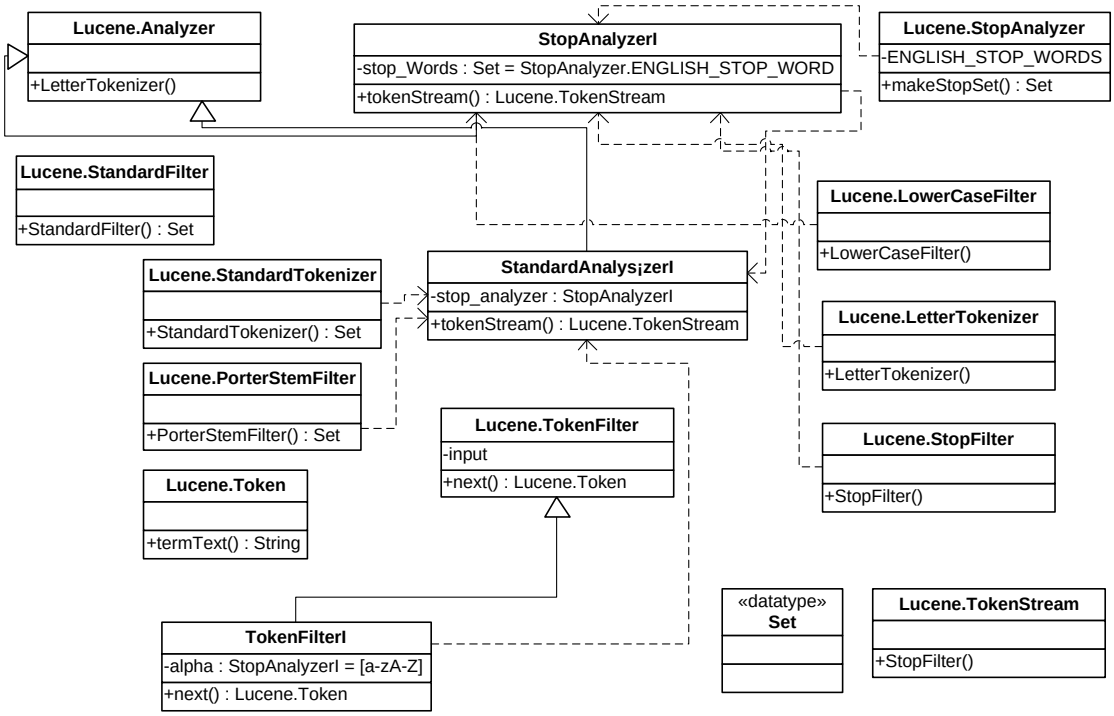
Anexo 2. Modelo general para el agrupamiento de documentos XML



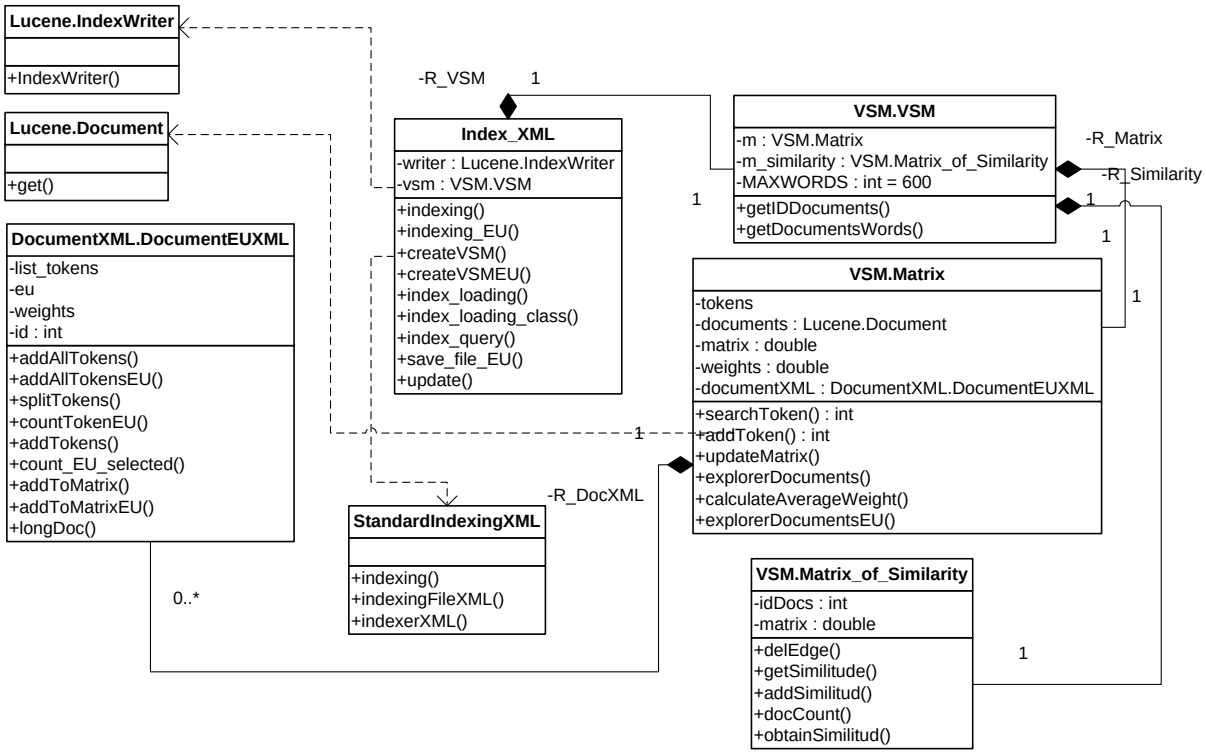
Anexo 3. Diseño de clases controladoras del sistema LucXML



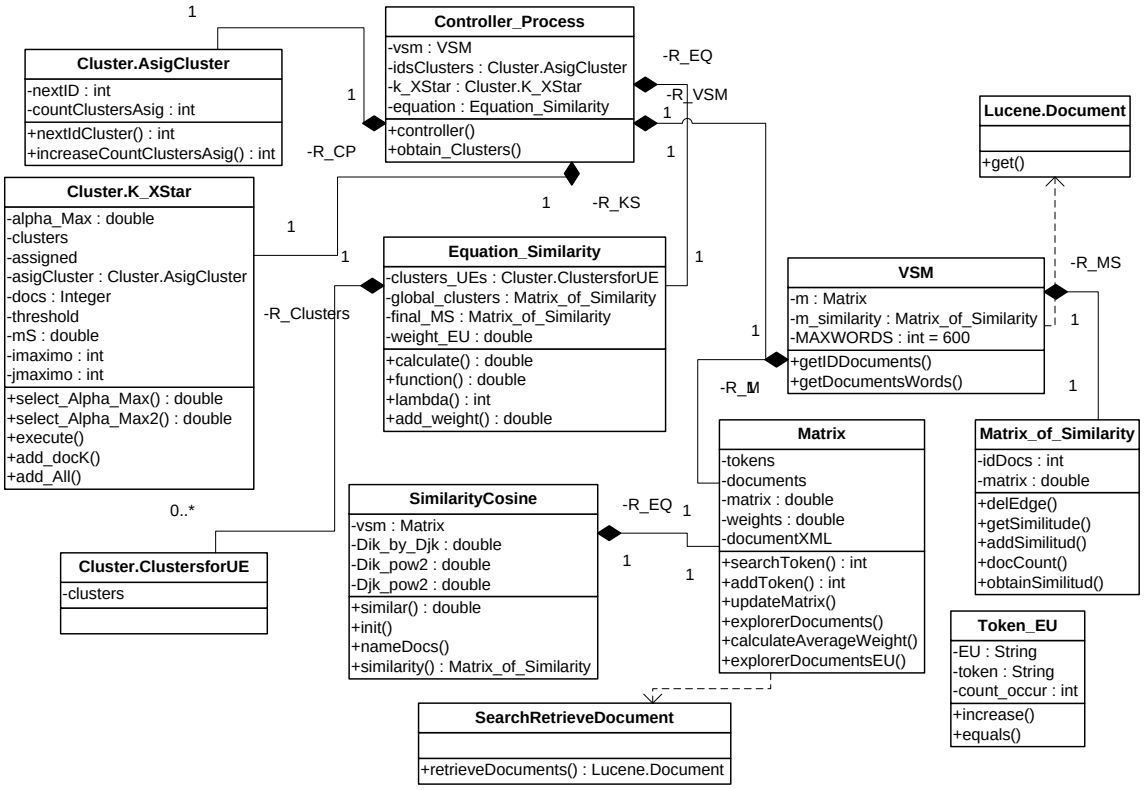
Anexo 4. Diseño de clases relacionadas con el proceso de análisis



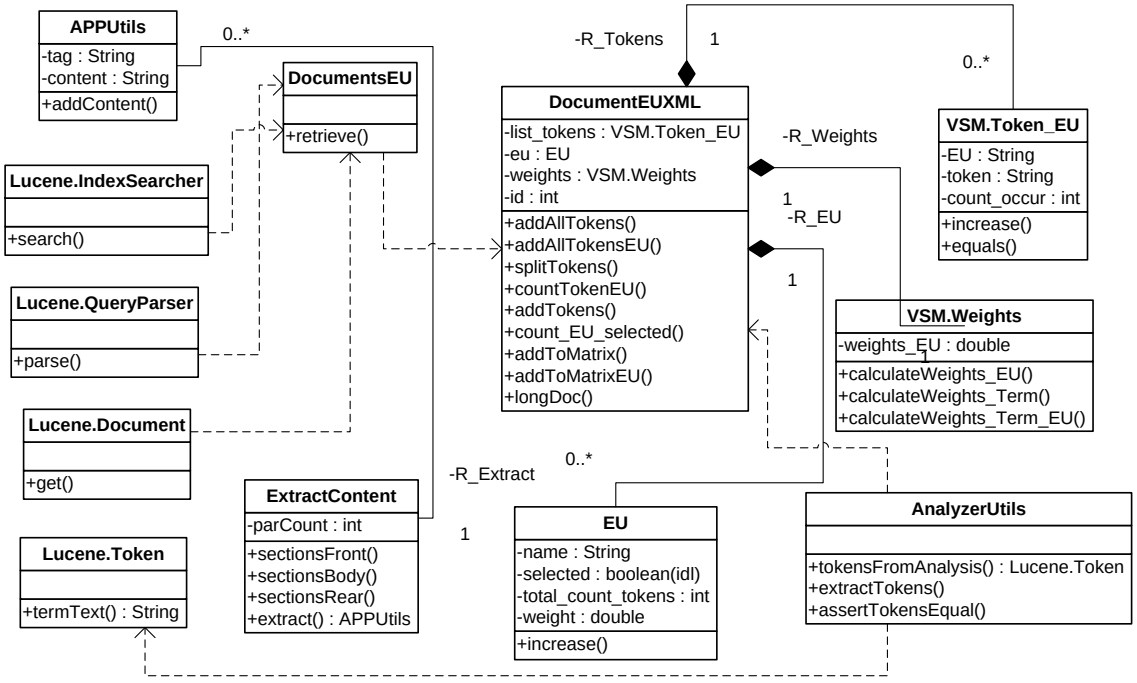
Anexo 5. Diseño de clases relacionadas con el proceso de indexación



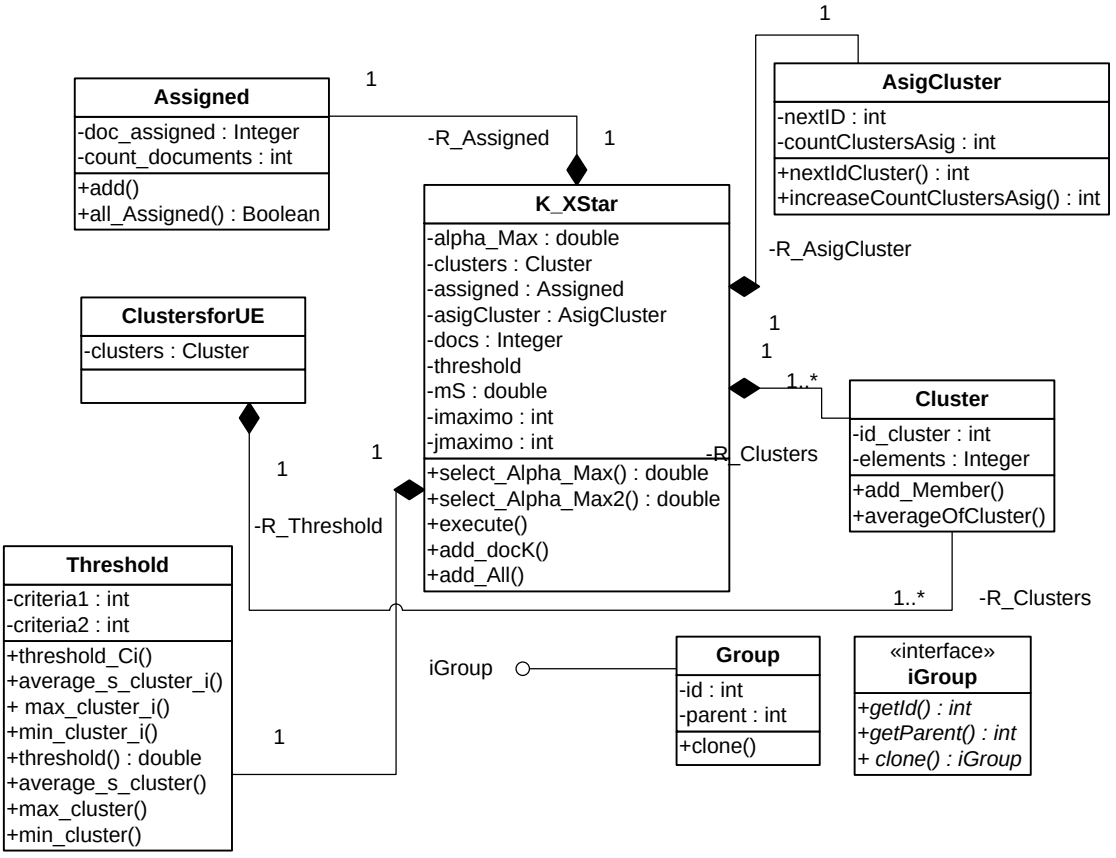
Anexo 6. Diseño de clases relacionadas con el proceso de representación VSM



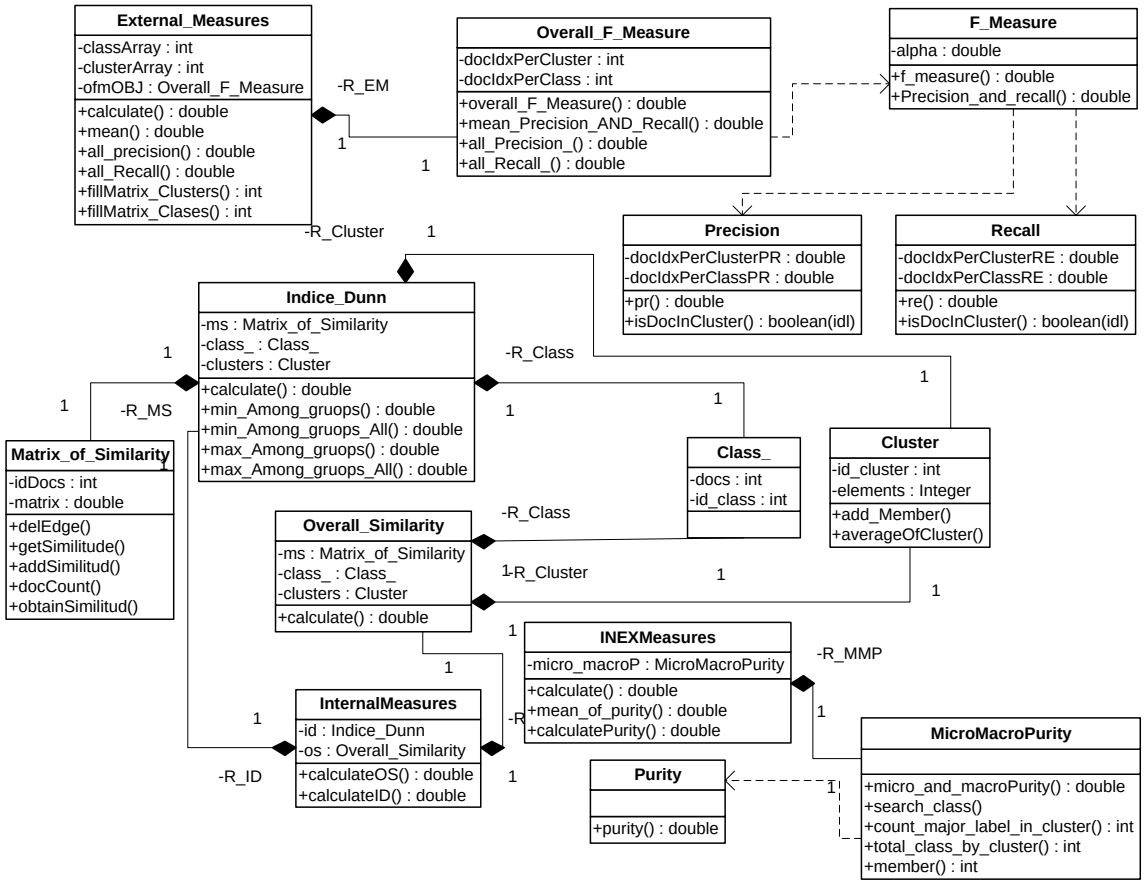
Anexo 7. Diseño de clases relacionadas con la manipulación de documentos XML



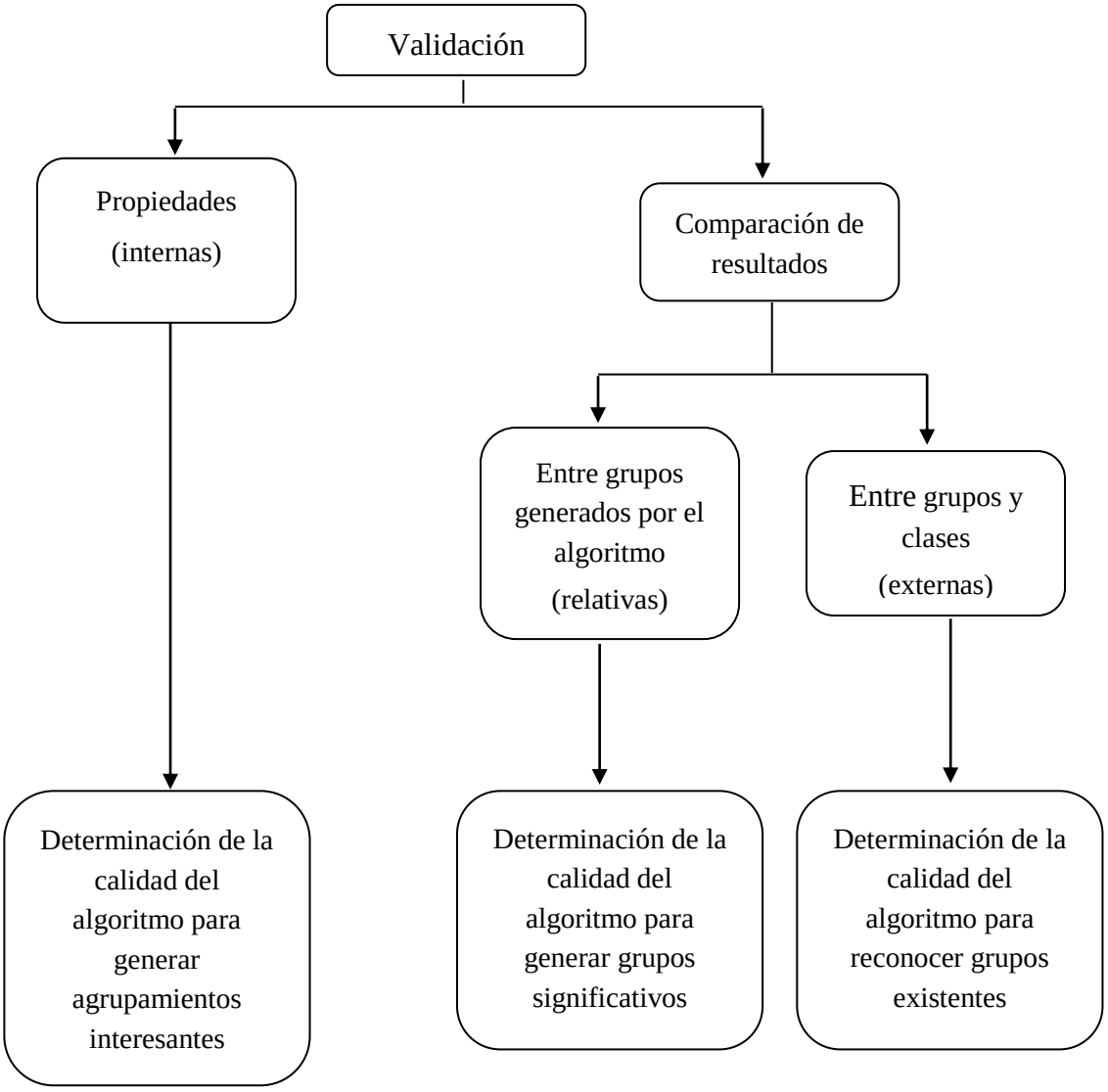
Anexo 8. Diseño de clases relacionadas con el proceso de agrupamiento de documentos XML



Anexo 9. Diseño de clases relacionadas con el proceso de evaluación de los resultados



Anexo 10. Clasificación simplificada de algunas técnicas para la validación de agrupamientos[17]



[17]Tomado de BRUN, M., SIMA, C., HUA, J., LOWEY, J., CARROLL, B., SUH, E. & DOUGHERTY, E. R. 2007. Model-based evaluation of clustering validation measures. *Pattern Recognition,* 40**,** 807-824.

Anexo 11. Algunas medidas externas e internas para la validación del agrupamiento

Medidas externas

Medida-*F* Global (Overall *F*-Measure; OFM)(Steinbach et al., 2000b)

$$Overall\ F-Measure = \sum_{i=1}^{k} \frac{n_i}{n} \max\{F-Measure(i,j)\} \qquad \text{(A3.1.1)}$$

donde k es el número de clases, n_i es el tamaño de la clase i, n es el número total de objetos agrupados y *F-Measure*(i, j) se calcula según la expresión siguiente. Si α= 1, entonces OFM se nombra Purity(Rosell et al., 2004b).

Medida-*F*(*F*-Measure) de la clase *i* respecto al grupo *j*

$$F-Measure(i,j) = \frac{1}{\alpha(1/\Pr(i,j)) + (1-\alpha)(1/\mathrm{Re}(i,j))} \qquad \text{(A3.1.2)}$$

Si α= 1 entonces *F-Measure*(i,j)coincide con precision, si α= 0 entonces *F-Measure*(i,j)coincide con cubrimiento. α= 0.5 significa igual peso para precisión y cubrimiento.

Micro-averaged precision y micro-averaged recall (Niu et al., 2004)

$$\mathrm{MA\text{-}Pr} = \frac{\sum_{i=1}^{k} \alpha_i}{\sum_{i=1}^{k} (\alpha_i + \beta_i)} \quad \text{y} \quad \mathrm{MA\text{-}Re} = \frac{\sum_{i=1}^{k} \alpha_i}{\sum_{i=1}^{k} (\alpha_i + \gamma_i)} \qquad \text{(A3.1.3)}$$

donde α_i es el número de objetos correctamente asignados a la clase i, β_i es el número de objetos incorrectamente asignados a la clase i y γ_i es el número de objetos incorrectamente no asignados a la clase i. MA-Pr = MA-Re si cada objeto pertenece a sólo un grupo y la clasificación de referencia también tiene una clasificación única para cada objeto.

Medidas internas

Similitud global (Overall Similarity)(Steinbach et al., 2000a)

$$OverallSimilarity(Grupo) = \frac{1}{|Grupo|^2} \sum_{O_i, O_j \in Grupo} \text{distancia}(O_i, O_j) \qquad \text{(A3.2.1)}$$

Índices Dunn

$$I(C) = \frac{\min_{i \neq j} \{\delta(C_i, C_j)\}}{\max_{1 \leq l \leq k} \{\Delta(C_l)\}} \qquad \text{(A3.2.1)}$$

donde $C=\{C_1, \ldots, C_k\}$ es el agrupamiento de un conjunto de objetos O, $\delta: C \times C \rightarrow R$ es una medida de distancia de grupo a grupo y $\triangle: C \rightarrow R$ es una medida de diámetro del grupo.

$$\delta(C_i, C_j) = \min_{x \in C_i, y \in C_j} d(x, y) \text{ y } \Delta(C_i) = \max_{x, y \in C_i} d(x, y) \qquad \text{(A3.2.3)}$$

donde $d: C \times C \rightarrow R$ es una función que mide la distancia entre los objetos de O.

Una de las propuestas de Bezdek para el cálculo de $\delta(C_i, C_j)$ y $\Delta(C_i)$

$$\delta(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{x \in C_i, y \in C_j} d(x, y) \text{ y } \Delta(C_i) = 2\left(\frac{\sum_{x \in C_i} d(x, c_i)}{|C_i|} \right) \qquad \text{(A3.2.2)}$$

donde c_i es el centro del grupo C_i.

Medidas propuestas por INEX

$$Purity(k) = \frac{NDMLC_k}{NDC_k} \qquad \text{(A3.3.1)}$$

$$Micro - Purity(k) = \frac{\sum_{k=0}^{n} Purity(k) * TotalFoundByClass(k)}{\sum_{k=0}^{n} TotalFoundByClass(k)} \qquad \text{(A3.3.2)}$$

$$Macro - Purity(k) = \frac{\sum_{k=0}^{n} Purity(k)}{TotalofCategories} \qquad \text{(A3.3.3)}$$

Donde: se asume el total de categorías como la cantidad de grupos encontrados.

Anexo 12. Descripción de los casos de estudio utilizados

Tabla A12.1 Descripción de los archivos utilizados para evaluar la calidad del agrupamiento.

No. Corpus	Cantidad de documentos	Cantidad de clases	Valores ausentes
Conjuntos de documentos XML confeccionados a partir de documentos recuperados del sitio de ICT del Centro de Estudios de Informática de la Universidad Central "Marta Abreu" de Las Villas http://ict.cei.uclv.edu.cu			
5	34	2	Fuzzy Logic, SVM
6	30	2	Association Rules, SVM
7	30	2	Fuzzy Logic, Rough Set
8	49	3	Association Rules, SVM, Fuzzy Logic
9	49	3	Rough Set, SVM, Fuzzy Logic
10	64	4	Rough Set, Association Rules, SVM, Fuzzy Logic
11	45	3	Rough Set, Association Rules, Fuzzy Logic
12	34	2	Association Rules, SVM
13	35	2	Rough Set, SVM
14	49	3	Rough Set, Association Rules, SVM
15	30	2	Rough Set, Association Rules
Recopilación de documentos del repositorio IDE-Alliance, internacionalmente utilizados para evaluar agrupamiento. Proporcionados por la Universidad de Granada, España.			
1	21	2	Belief Propagation, CL
2	23	2	Belief Propagation, Copula
3	33	3	Copula, Belief Propagation, CL
4	22	2	CL, Copula
Recopilación de documentos del repositorio Wikipedia[18], internacionalmente utilizados para evaluar agrupamiento, a través de INEX.			
16	17	2	Politics, Party

18 http://www.inex.otago.ac.nz

Anexo 13. Comparación de la calidad del agrupamiento para el cálculo del umbral

Tabla A13.1 Valores de la medida *Overall F-Measure* obtenidos al aplicar el agrupamiento propuesto para los casos de estudio 1 y 2, utilizando los criterios para el cálculo del umbral: media de los mínimos (OFM-Min), media de los máximos (OFM-Max) y media de todas las similitudes (OFM-Media).

Corpus	OFM-**Min**	OFM-**Max**	OFM-**Media**
1	0.684	0.726	0.852
2	0.659	0.5	0.759
3	0.659	0.714	0.837
4	0.502	0.559	0.72
5	0.5	0.691	0.784
6	0.402	0.633	0.685
7	0.497	0.556	0.582
8	0.663	0.667	0.881
9	0.663	0.703	0.886
10	0.502	0.676	0.856
11	0.659	0.681	0.874
12	0.657	0.685	0.947
13	0.657	0.539	0.828
14	0.656	0.696	0.977
15	0.497	0.598	0.966

Tabla A13.2 Estadísticas descriptivas basadas en la medida OFM. según el criterio utilizado para el cálculo del umbral.

Criterios	Min	Max	$\bar{x}$
OFM-Min	.402	.684	.591
OFM-Max	.500	.726	.642
OFM-Media	.582	**.977**	**.823**

Tabla A13.3 Valores de significación de la prueba no paramétrica de Wilcoxon para comparar la calidad de los resultados del agrupamientos basada en la medida OFM. según el criterio utilizado para el cálculo del umbral.

Significación de la prueba Wilcoxon		Positive Ranks	Negative Ranks	Ties
Min - Max	**.036**	13[a]	2[b]	0[c]
Media - Min	**.001**	**15[d]**	0[e]	0[f]
Max - Media	**.001**	**15[g]**	0[h]	0[i]

a. Max>Min b. Max<Min c. Max=Min d. Media>Min e. Media<Min f. Media=Min g. Media>Max h. Media<Max i. Media=Max

Anexo 14. Resultados del experimento 1

Tabla A14.1 Valores de la medida *Overall F-Measure* al aplicar el algoritmo *K*-Star. INEX*K*-Star y el Algoritmo 1. propuesta de esta investigación.

Corpus	OFM-***K*-Star**	OFM- **Algoritmo1**	OFM- **INEX*K*-Star**
1	0.694	0.852	0.64
2	0.6	0.759	0.665
3	0.682	0.837	0.605
4	0.471	0.72	0.708
5	0.524	0.784	0.659
6	0.47	0.685	0.743
7	0.579	0.582	0.485
8	0.891	0.881	0.647
9	0.624	0.886	0.649
10	0.646	0.856	0.765
11	0.771	0.874	0.632
12	0.923	0.947	0.801
13	0.838	0.977	0.819
14	0.732	0.966	0.644
15	0.883	0.828	0.819
16	0.529	0.908	0.588

Tabla A14.2 Resultados de la prueba estadística de Wilcoxon con los valores de la Tabla 14.1 del algoritmo *K*-Star y el Algoritmo 1. propuesta de esta investigación.

		N	**Mean Rank**	**Sum of Ranks**		**Alg1-K-Star**
ofm_Algoritmo1 - ofm_K-Star	Negative Ranks	2[a]	3.00	6.00	Z	-3.206[a]
	Positive Ranks	**14**[b]	9.29	130.00	Aymp. Sig (2-tailed)	**0.001**
	Ties	0[c]				
	Total	16				

(*a. Alg1 < KStar b. Alg1 > KStar c. Alg1 = KStar a. Base on positive ranks*)

Tabla A14.3 Resultados de la prueba estadística de Wilcoxon con los valores de la Tabla 14.1 del algoritmo INEX*K*-Star y el Algoritmo 1. propuesta de esta investigación.

		N	**Mean Rank**	**Sum of Ranks**		**Inex-Alg1**
ofm_INEXKStar - ofm_Algoritmo1	Negative Ranks	**15**[a]	8.87	133.00	Z	-3.361[a]
	Positive Ranks	1[b]	3.00	3.00	Aymp. Sig (2-tailed)	**0.001**
	Ties	0[c]				
	Total	16				

(*a. INEX < Alg1 b. INEX >Alg1 c. INEX = Alg1 a. Base on positive ranks*)

Anexo 15. Resultados del experimento 2

Tabla A15.1 Valores de las medidas *Micro-Purity* y *Macro-Purity* al aplicar el algoritmo INEX*K*-Star y el Algoritmo 1. propuesta de esta investigación.

Corpus	*Micro-Purity*		*Macro-Purity*	
	INEX*K*-Star	**Algoritmo1**	**INEX*K*-Star**	**Algoritmo1**
1	0.59	0.597	0.59	0.664
2	0.552	0.689	0.552	0.806
3	0.525	0.638	0.524	0.71
4	0.515	0.461	0.560	0.55
5	0.716	0.568	0.792	0.534
6	0.559	0.451	0.488	0.483
7	0.3	0.673	0.2	0.759
8	0.405	0.418	0.405	0.628
9	0.725	0.418	0.816	0.628
10	0.459	0.557	0.44	0.643
11	0.481	0.571	0.481	0.596
12	0.511	1	0.511	1
13	0.542	0.545	0.656	0.697
14	0.636	1	0.627	1
15	0.463	0.633	0.529	0.71
16	0.722	0.958	0.792	0.972

Tabla A15.2 Resultados de la prueba estadística de Wilcoxon con los valores de *Micro-Purity* de la Tabla 15.1 del algoritmo INEX*K*-Star y el Algoritmo 1. propuesta de esta investigación.

		N	**Mean Rank**	**Sum of Ranks**		**Inex-Alg1**
microP_INEXKStar - microP_Algoritmo1	Negative Ranks	**12**[a]	8.96	107.50	Z	-2.043[a]
	Positive Ranks	4[b]	7.13	28.50	Aymp. Sig (2-tailed)	**0.04**
	Ties	0[c]				
	Total	16				

(*a. INEX < Alg1 b. INEX >Alg1 c. INEX = Alg1 a. Base on positive ranks*)

Tabla A15.3 Resultados de la prueba estadística de Wilcoxon con los valores de *Macro-Purity* de la Tabla 15.1 del algoritmo INEX*K*-Star y el Algoritmo 1. propuesta de esta investigación.

		N	**Mean Rank**	**Sum of Ranks**		**Inex-Alg1**
macroP_INEXKStar - macroP_Algoritmo1	Negative Ranks	**12**[a]	9.25	111.00	Z	-2.223[a]
	Positive Ranks	4[b]	6.25	25.00	Aymp. Sig (2-tailed)	**0.026**
	Ties	0[c]				
	Total	16				

(*a. INEX < Alg1 b. INEX >Alg1 c. INEX = Alg1 a. Base on positive ranks*)