UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS
FACULTAD DE MATEMÁTICA, FÍSICA Y COMPUTACIÓN
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

Metodología para el agrupamiento de documentos semiestructurados

Tesis presentada en opción al grado científico de Doctor en Ciencias Técnicas

MSc. Damny Magdaleno Guevara

Santa Clara, 2015

UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS
FACULTAD DE MATEMÁTICA, FÍSICA Y COMPUTACIÓN
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

Metodología para el agrupamiento de documentos semiestructurados

Tesis presentada en opción al grado científico de Doctor en Ciencias Técnicas

Autor: MSc. Damny Magdaleno Guevara

Tutor: Dr. María M. García Lorenzo

Santa Clara, 2015

Siempre dije que tenía que terminar este trabajo, para poder dedicárselo a una personita que apareció en mi vida y desde ese momento cambió toda, llenándola de alegría.

Para David.

AGRADECIMIENTOS

Aunque no lo parezca, este es el fragmento del documento que me es más difícil de escribir. Son muchos los que estuvieron cerca a lo largo de este trabajo, y que me tendieron su mano de una forma u otra. A todos, muchas gracias, en especial:

A Dios, .

A mi mamá, por estar siempre al tanto de todo, sintiendo mis alegrías y mis pesares, aunque muchas veces no se daba cuenta de todo el enredo que tenía en mi cabeza.

A mi tío Eduardo, el día que existan las palabras que lleva mi agradecimiento, te prometo que vuelvo a hacer otra tesis (que dure menos, claro) para ponerlas.

A mi tío Rafael, por estar siempre al tanto, aconsejarme y recordarme que esta es mi prioridad.

A Diego, donde quiera que estés, sé que te sentirías orgulloso.

A mis amigos en especial a Norma, por su preocupación y Maikel, por todos sus regaños, consejos y ayuda.

A Amanda, te mereces una hoja en blanco para cada vez que me acuerde de algo de todo lo que me has apoyado en este tiempo, mayor que tres años, correr y ponerlo. ¿Estaría escribiendo esto ahora?

A mis compañeros de trabajo, en especial (es una *bolsa de nombres*) a Michel, Gheisa, Magalys, Greta, Gonzalo, Isel, Danel, Gladita, Olguita.

A todos los estudiantes con los que he compartido esta investigación, me ayudaron mucho.

A Leticia, todo lo que he aprendido de esto es gracias a ti.

A Morell, gracias por “invitarme a fumar”.

A Ivett, cuando todo esto se hundía te dije que te fueras de mi lado y nunca se me van a olvidar tus palabras, gracias.

A Marilyn, no sé si darte las gracias o pedirte disculpas, por darte esta carga. Me guiaste, me orientaste, en fin, gracias a esto me encuentro escribiendo los agradecimientos de mi tesis.

SÍNTESIS

Los documentos con formato semiestructurado –destacándose el XML– juegan un papel fundamental a nivel mundial dado el crecimiento exponencial de las publicaciones científicas en Internet y la necesidad de almacenar los artículos científicos en formatos que permitan una mejor manipulación de los mismos y aumentar de esta manera la eficacia de los sistemas de recuperación de información. 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 propone una metodología para el agrupamiento de documentos científicos en formato semiestructurado utilizando el contenido y la estructura de los mismos. Los principales resultados son: la metodología para el agrupamiento; la función de similitud *OverallSimSUX*, que permite capturar eficientemente la semejanza entre los documentos; las aplicaciones: trabajo con documentos científicos en formato XML; aplicación WEB, incorporando documentos científicos en diferentes formatos y una aplicación en el área de la Salud. Al evaluar las propuestas con datos representativos se obtuvieron resultados favorables con la utilización de la metodología y su extensión.

ABSTRACT

The semi-structured format documents - highlighting the XML - play a major global role given the exponential growth of scientific publications on the Internet and the need to store scientific articles in formats that enable better handling of them and thus increase the effectiveness of information retrieval systems. Scientific information management becomes increasingly complex and challenging, especially since document collections are generally heterogeneous, large, diverse and dynamic. Overcoming these challenges is essential to give scientists better conditions to manage the time required to process scientific information.

In this work a methodology for the clustering of scientific documents in semi-structured format using their content and structure is proposed. The main results are: the methodology for the grouping; OverallSimSUX similarity function, which allows efficiently capture the similarity between documents; applications: working with scientific documents in XML format; WEB application, incorporating scientific documents in different formats and an application in the area of Health. In assessing the proposals with representative data, favorable results were obtained with the use of the methodology and its extension.

TABLA DE CONTENIDOS

INTRODUCCIÓN

En el siglo XX surge un fenómeno conocido como la explosión de la información dado el inmenso volumen existente en la red, lo cual estableció un desafío interesante al tratar de beneficiar a los usuarios con las facilidades de acceso a la misma. Para dar respuesta a esto se desarrolla un área del conocimiento denominada Recuperación de Información, que estudia y propone soluciones al escenario presentado y plantea modelos, algoritmos, heurísticas, entre otras. Diversas herramientas documentales como buscadores y directorios se han desarrollado con el objetivo de lograr una recuperación rápida, efectiva y eficiente.

Teniendo en cuenta la estructura de los documentos, estos se clasifican en: estructurados, no estructurados y semiestructurados. Los que presentan formato semiestructurado, destacándose el XML, juegan un papel fundamental a nivel mundial dado el crecimiento exponencial de las publicaciones en Internet y la necesidad de almacenar los artículos en formatos que permitan una mejor manipulación de los mismos y aumentar de esta manera la eficacia de los sistemas de recuperación de información.

El metalenguaje XML (Extensible Markup Language) desarrollado por el W3C[1] proveniente de GML (Generalized Markup Language) 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 [1].

A esto se añade que los documentos XML contienen su información en forma semiestructurada [2] ya que incorporan estructura y datos en una misma entidad y son extensibles, con estructura de fácil análisis y procesamiento, por lo que se convirtieron en el formato de intercambio de datos estándar entre las aplicaciones Web [1]. Las etiquetas existentes en estos documentos permiten la descripción semántica del contenido de los

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

elementos. De este modo, la estructura de ellos se puede explotar en la recuperación de documentos relevantes [3].

Según Guerrini (2006), la información existente en formato semiestructurado se usa cada día más [3]. Por ejemplo, en el caso de las publicaciones científicas, esto se debe a que generalmente los artículos científicos, tienen una estructura bien definida, (*título*, *autor*, *palabras claves*, *resumen*, *contenido*, *referencias bibliográficas*, entre otras) la cual es fácilmente adaptable a formatos semiestructurados, delimitándose esta estructura en Unidades Estructurales de los documentos XML. De ahí que las revistas de contenido científico han migrado hacia documentos con formato semiestructurados, dada las ventajas que estos brindan al poder etiquetar los documentos para el acceso a partes específicas de los mismos. Así revistas científicas de prestigio internacional y bases de datos internacionales adoptan el formato XML como estándar; por ejemplo, los libros y revistas de *Elsevier*[2]; uno de los repositorios de bibliografía en Ciencias de la Computación en línea más reconocidos (*DBLP*[3]) también posee sus datos en formato XML; *Scielo* [4], propone normas[4] basadas en este metalenguaje, para el almacenamiento de los trabajos científicos.

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 [5-7].

Existen varias formas de gestionar el conocimiento: la categorización, la clasificación y el agrupamiento [8, 9]. Particularmente, el agrupamiento 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 [10]. 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 [11].

[2] www.elsevier.com/journals/title/a

[3] dblp.uni-trie.de

[4] ISO 12083-1994 (Electronic Manuscript Preparation and Markup)

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 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 permite descubrir una estructura previamente oculta en los datos; sin embargo, la asignación de los objetos a las clases y la descripción de esas clases son desconocidas [12].

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. Explotar la estructura específica que tienen los artículos científicos puede ofrecer resultados favorables en el agrupamiento de este tipo de documentos.

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 [13, 14]; sin embargo, un buen proceso de agrupamiento no puede descartar el uso de la estructura [15], 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 [11].

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 en formato semiestructurados se centran en tratar de forma independiente el contenido y la estructura. 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 perfeccionar o crear técnicas que utilicen ambas.

El **objetivo general** de esta investigación consiste en diseñar una metodología para el agrupamiento automático de documentos almacenados en formato semiestructurado,

combinando eficazmente el contenido y la estructura existente en los mismos, para contribuir a la organización de la información recuperada.

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

1. Diseñar e implementar la metodología para el agrupamiento de documentos con información semiestructurada, combinando el contenido y la estructura presente en los mismos.
2. Definir una función de similitud que permita capturar el grado de semejanza entre los documentos tomando la relación existente entre las unidades estructurales para facilitar la eficacia del agrupamiento.
3. Analizar de forma diferenciada la unidad estructural *Referencias Bibliográficas* para documentos semiestructurados que representan trabajo científico.
4. Evaluar la metodología propuesta a partir de corpus representativos del universo investigado.
5. Aplicar la metodología a diferentes problemáticas.

Las **preguntas de investigación** planteadas son:

1. ¿Cómo combinar la relación estructura-contenido de los documentos XML, a nivel de las unidades estructurales existentes en los documentos?
2. ¿En qué medida el nuevo modelo aporta mejores resultados al agrupamiento de documentos XML que otras variantes propuestas?
3. ¿Cómo procesar eficientemente la información existente en las *Referencias Bibliográficas*, para obtener información útil para el agrupamiento?

Como respuestas a las preguntas de investigación y después de haber realizado el marco teórico se formuló la siguiente **hipótesis de investigación**: La metodología para el agrupamiento de documentos científicos que combina la relación estructura-contenido y a su vez incorpora una función de similitud que trate las referencias bibliográficas de manera diferenciada, mejora la eficacia del agrupamiento sobre documentos semiestructurados.

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

1. Análisis de los métodos de agrupamiento de documentos XML y colecciones textuales.

2. Definición de una función de similitud que permita capturar la relaciones de dependencia entre los documentos recuperados, mediante la combinación de los resultados de agrupamientos para cada unidad estructural con la información global del documento.
3. Implementación de métodos de agrupamiento de documentos XML, que utilicen la relación estructura-contenido.
4. Estudio de las herramientas *Lucene, Tika, SolR* y otras especializadas en la manipulación documental.
5. Evaluación de la metodología.
6. Análisis de la Unidad Estructural *Referencias Bibliográficas* y su efecto en la metodología propuesta.

La **novedad científica** de la investigación radica en: La nueva metodología para el agrupamiento de documentos en formato semiestructurado, utilizando la función de similitud propuesta *OverallSimSUX*. La metodología propuesta considera un tratamiento diferenciado de las *referencias bibliográficas* al resto de las unidades estructurales de artículos científicos, que permite una mayor profundización en el análisis de la similitud entre este tipo de documento.

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

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

La aplicación de los resultados obtenidos a través de sistemas de recuperación de información (*LucXML* y *ScientificSolR*) que soportan la metodología. Estos sistemas permiten agrupar grandes volúmenes de datos, con el propósito de facilitar a los investigadores y docentes el inicio de una revisión del estado del arte, 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. El sistema *LucXML* permite indexar, recuperar y agrupar colecciones personales de documentos XML existentes en una estación local; por su parte *ScientifcSolr* presenta una interfaz WEB, con un servidor alojado en un repositorio remoto de información, permitiendo realizar recuperación sobre la información existente y agrupar los documentos relevantes a la consulta.

Entre los métodos de trabajo científico utilizados se destacan los siguientes:

Analítico-sintético al descomponer el problema de investigación en elementos por separado y profundizar en el estudio de cada uno de ellos, para luego sintetizarlos en la solución de las propuestas.

Histórico-lógico y el dialéctico para el estudio crítico de los trabajos anteriores, y para utilizar estos como punto de referencia y comparación de los resultados alcanzados.

Hipotético-deductivo para la elaboración de las hipótesis de la investigación y para proponer nuevas líneas de trabajo a partir de los resultados parciales que se obtuvieron.

Modelación para el desarrollo de los algoritmos.

Sistémico para el desarrollo de los diferentes sistemas computacionales y lograr que los elementos que formen parte de la aplicación real sean un todo que funcione de manera armónica.

Experimental para comprobar la utilidad de los resultados obtenidos a partir de las propuestas definidas y la comparación con otros métodos reportados.

Matemáticos-estadísticos para la validación de los aportes fundamentales de la investigación.

Análisis-síntesis e Inducción-deducción como vía de constatación teórica a lo largo de la tesis.

Coloquial para la presentación y discusión de los resultados en sesiones científicas.

La tesis está estructurada en tres capítulos. En el Capítulo 1 se tratan los documentos semiestructurados para el almacenamiento de la información, específicamente documentos XML. Se aborda el agrupamiento y las principales medidas para validarlo. Se profundiza en el análisis de los métodos que combinan contenido y estructura. En el Capítulo 2 se presenta una metodología general para la aplicación del agrupamiento, combinando la relación estructura-contenido; obteniéndose una nueva medida de semejanza que facilita evaluar el grado de relación entre los documentos, en este capítulo se presentan dos sistemas en los que se implementó la metodología (*XMLearning* y *LucXML*), el primero para realizar la evaluación de la metodología a través de la comparación de varios algoritmos de agrupamiento incorporados, el segundo recupera información desde un repositorio local. El Capítulo 3 muestra varias aplicaciones de la metodología: la primera para documentos

científicos en formato XML; una segunda aplicación que consiste en un sistema con interfaz WEB, analizando documentos científicos en diferentes formatos; una tercera aplicación de la metodología está enfocada al área de la salud, específicamente considerando las Historias Clínicas Electrónicas. Este documento culmina con las conclusiones, recomendaciones, referencias bibliográficas, producción científica del autor sobre el tema de la tesis y los anexos.

1

MÉTODOS DE AGRUPAMIENTO DE DOCUMENTOS SEMIESTRUCTURADOS

1. MÉTODOS DE AGRUPAMIENTO DE DOCUMENTOS SEMIESTRUCTURADOS

En la actualidad hay disponible una gran cantidad de información, la cual se incrementa cada día producto al avance científico-técnico. La gran cantidad de información científica disponible para los usuarios, hace que sea difícil para los motores de búsqueda identificar la información relevante [16]. Por ejemplo, solo en el ámbito biomédico alrededor de 1 800 nuevos documentos se publican diariamente [17]. Por lo que se hace necesario la confección de herramientas que se enfrenten a esta problemática [18, 19].

Los formatos semiestructurados, específicamente XML, juegan un papel fundamental a nivel mundial dado el crecimiento exponencial de las publicaciones científicas en Internet provocando la necesidad de almacenar los artículos científicos en formatos que permitan una mejor manipulación de los mismos y aumentar de esta manera la eficacia de los sistemas de recuperación de información [20].

Por esta razón, es necesario 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 auto-descriptiva de estos documentos. Particularmente, a partir de la información disponible, el agrupamiento permite organizar, delimitar relevancia y descubrir nuevo conocimiento [8, 9].

En este capítulo se describen los documentos XML; se relacionan las principales técnicas de agrupamiento, específicamente para documentos XML; las funciones de similitud más utilizadas en dominios textuales; así como las principales medidas externas para la evaluación de los resultados del agrupamiento. Por último, se mencionan varias herramientas actuales destinadas a la manipulación de documentos.

1.1 Documentos XML para almacenar información

El procesamiento automático de textos ha alcanzado un desarrollo considerable, facilitando a los usuarios la gestión de grandes volúmenes de información. Se han creado herramientas que permiten filtrar información relevante o interesante a partir de información no relevante acorde a intereses especificados por los usuarios, permitiendo que la información pueda ser resumida y visualmente presentada [21].

Los documentos se clasifican de acuerdo a su estructura como: estructurados, no estructurados y semi-estructurados [22, 23].

- Documentos no estructurados: La información contenida en el documento no tiene ningún orden de estructura, esta información tiene mayor riesgo de no ser encontrada por los buscadores, pues no contiene parámetros establecidos que proporcionen la información a buscar.
- Documentos semiestructurados: Se definen como aquellos documentos que en la mayoría de su contexto contienen elementos de un documento estructurado, dejando algunas partes del documento sin estructurar.
- Documentos estructurados: A diferencia de los documentos semiestructurados y no estructurados contienen una estructura predefinida la cual está definida por etiquetas cuyo objetivo es mostrar información relevante del documento.

Los documentos semiestructurados son muy utilizados en la literatura [24-26] ya que incorporan estructura y contenido en una misma entidad. Debido a la importancia de lograr un almacenamiento correcto de este tipo de documentos, se usan lenguajes específicos (RSS [27], HL7[5]AIML[6], WSDL[7], XML[8], entre otros), en su mayoría basados en XML, siendo este uno de los lenguajes de mayor uso hoy en día para el almacenamiento de la información semiestructurada [1, 28, 29].

[5] http://www.hl7.org/

[6] http://www.alicebot.org/aiml.html

[7] www.w3schools.com/webservices/ws_wsdl_intro.asp

[8] www.w3.org/XML/

1.1.1 Lenguaje XML

En las Figura 1.1 se puede observar un ejemplo de documento XML correspondiente a un artículo científico y el árbol que contiene su estructura se muestra en la Figura 1.2. Un documento XML está compuesto por elementos, los que se señalan mediante etiquetas.

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

<Artículo>
   <Titulo>
      "Agrupamiento de documentos estructurados"
   </Titulo>
   <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

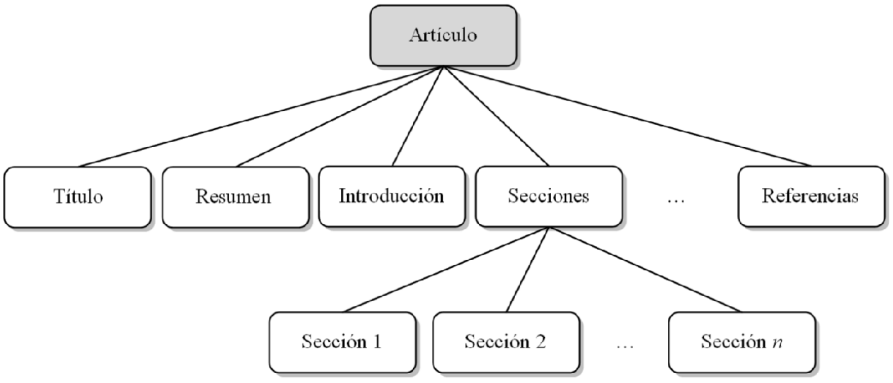


Figura 1.2: Ejemplo de un árbol correspondiente a un artículo científico

XML tiene un número de características que lo hace ampliamente utilizable como formato de representación de datos, entre las que se encuentran: su extensibilidad e independencia de la plataforma utilizada [1, 29]. Su extensibilidad se debe al hecho que no tiene un conjunto

de etiquetas fijas, 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. Por otra parte, es independiente de la plataforma utilizada, del sistema operativo, o del fabricante de software. Esta independencia permite interoperabilidad entre plataformas diferentes de programación y sistemas operativos.

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 [3].

Se dice que un documento XML está bien formado si cumple con todas reglas de sintaxis definidas para este tipo de documento, por otra parte un documentos es válido si además de estar bien formado, cumple con determinadas reglas y normas. Para establecer las reglas de construcción de documentos XML se utilizan los DTD y los Esquemas XML.

- DTD: definen los elementos, atributos, entidades y notaciones que pueden utilizarse para construir un tipo de documento, así como las reglas para su utilización. Mediante estas reglas se comprueba la validez de un documento. Los DTD poseen una sintaxis especializada.
- Esquemas XML: tienen la misma finalidad que los DTD, describen los elementos y atributos que se pueden utilizar para construir documentos XML y las reglas de utilización, pero permiten asociar tipos de datos con los elementos.

Algunas de las principales técnicas utilizadas para el análisis de documentos XML se basan en dos interfaces de aplicación: SAX (Simple API for XML) y DOM (Document Object Model) [30], las cuales analizan el documento y son capaces de construir su árbol de objetos (elementos, atributos, etc.) para poder realizar búsquedas o transformaciones en el mismo [31].

- SAX: Interfaz de programación de aplicaciones (API), únicamente para el lenguaje de programación Java, convirtiéndose en el API estándar para usar XML en Java. Existen actualmente versiones de SAX para otros lenguajes de programación como por ejemplo Python. Una de las ventajas de usar SAX es la eficiencia en cuanto al tiempo y la memoria empleados en el análisis. Como desventaja, tiene que realizar una lectura

secuencial del documento por lo que una vez leído no se puede volver atrás, algo que DOM sí permite.

- DOM: API que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos. A través del DOM los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML. DOM es en esencia, una interfaz de programación de aplicaciones para acceder, añadir y cambiar dinámicamente contenido estructurado en documentos.

A partir de DOM surge *jdom*[9], biblioteca de código abierto para manipulaciones de datos XML optimizados para *Java*. La principal diferencia entre DOM y *jdom* es que el primero se creó para ser un lenguaje neutral e inicialmente se usó para manipulación de páginas HTML con *JavaScript* sin embargo, *jdom* se creó para usarse con *Java* y por tanto beneficiarse de las características de este lenguaje, incluyendo sobrecarga de métodos, colecciones, entre otras.

Existen otras herramientas para el trabajo con documentos XML como las bases de datos XML (XMLDB) y XPath[10] (XML Path Language), siendo este último un lenguaje que permite construir expresiones que recorren y procesan un documento XML.

1.1.2 Artículos científicos en formato XML

Los artículos científicos, en su mayor parte, aparecen en Internet en formato PDF en las revistas electrónicas correspondientes. Aunque los datos de título, autoría, publicación, dirección, filiación y resumen son rastreables a través de las bases de datos y de algunos buscadores académicos de Internet, no lo son así los contenidos del artículo. Sólo se puede buscar dentro del contenido del artículo una vez abierto el PDF, utilizando una herramienta de búsqueda de Acrobat Reader[11], Foxit[12] o similares [32].

[9] http://www.jdom.org/

[10] http://www.w3.org/TR/xpath

[11] https://get.adobe.com/es/reader/

[12] https://www.foxitsoftware.com/products/pdf-reader/

La información de un artículo científico suele ser de interés para los investigadores. Estos documentos, tienen una estructura básica que se debe respetar y que contiene al menos los elementos siguientes: *autor*, *título*, *resumen*, *palabras claves*, *contenido principal del artículo*, *notas* y *referencias bibliográficas*. La búsqueda de información en Internet se hace a través de robots de búsqueda que se limitan a leer códigos de información, por lo cual si la información rastreada no está marcada de tal manera que se diferencien cada uno de los elementos que componen un artículo científico, los robots no distinguirán entre los distintos elementos del documento; así no sabrán que el título o las palabras claves tienen más importancia que un párrafo del contenido, o que un mismo autor referenciado en dos artículos puede dar un indicio de que estos artículos traten temas similares.

XML se usa para estructurar precisamente todos los elementos de los textos en los procesos de edición contemporánea de los artículos y otros tipos de documentos [33]. Es por ello que en los últimos años está ocurriendo un cambio en cuanto a la forma de publicación de los artículos científicos y XML comienza a jugar un papel fundamental en la comunidad científica internacional.

XML tiene muchas características para mejorar la presencia en la web, la distribución y el presupuesto. Es por ello, que las grandes editoriales de revistas (*Elsevier*[13], *DBLP*[14], *Scielo* [4], *PubMed Central*[15]) lo utilizan como base para el desarrollo de sus publicaciones.

Según [34], dos de las ventajas que ofrece este tipo de documentos para el trabajo con la información científica son:

- Facilitar la publicación anticipada en línea (AOP). Los artículos AOP son publicados en línea por delante de su fecha de publicación impresa, y estas versiones web son finales, es decir no presentan cambios en cuanto a la versión impresa. La publicación web anticipada permite una producción más rápida de los datos sensibles al tiempo, lo que significa que puede leerse y citarse antes, lo cual es especialmente importante para las revistas científicas.

[13] www.elsevier.com/journals/title/a

[14] dblp.uni-trie.de

[15] http://www.ncbi.nlm.nih.gov/pmc/

- Utilización de los metadatos en función de facilitar el proceso de recuperación de información para el usuario. Un ejemplo importante de cómo XML y los metadatos se pueden utilizar con este fin es *CrossRef*[16], que fue establecido por las editoriales académicas como "una infraestructura para la vinculación de las citas a través de los editores". En este sistema, un investigador a partir de la lectura de un artículo electrónico puede acceder directamente a otro artículo que sea referenciado en el artículo que está leyendo [34].

Gestionar el conocimiento a partir de la información encontrada es fundamental en el trabajo científico [35]. Sin embargo, la gestión de información científica se vuelve cada vez más compleja y desafiante. Superar estos desafíos es esencial para dar a los científicos mejores condiciones de administrar el tiempo necesario para procesar la información científica.

El conocimiento se puede gestionar de diversas formas y hacerlo requiere de la integración de varias áreas del saber: el descubrimiento de conocimiento en bases de datos, la minería de datos y de textos. Esta última integra la recuperación y extracción de información, el análisis de textos, el resumen, la categorización, la clasificación, el agrupamiento, la visualización, la tecnología de bases de datos, el aprendizaje automático y la minería de datos [8, 9].

En la actualidad los usuarios se enfrentan a grandes colecciones de información y tienen que ser muy pacientes para analizar la información que verdaderamente necesitan. La categorización, clasificación y agrupamiento se pueden utilizar para refinar resultados de la recuperación y extracción de información, y así contribuir al descubrimiento de conocimiento.

1.2 Agrupamiento de documentos en formato XML

Con el desarrollo de la Inteligencia Artificial y específicamente la Minería de Textos en la actualidad pueden ser procesados grandes volúmenes de información con el objetivo de extraer patrones que residan en los datos almacenados [36]. Las técnicas más utilizadas se dividen en tres grandes grupos: la clasificación, la categorización y el agrupamiento.

La *clasificación* comprende la distribución de los objetos de cualquier género de clases, sobre la base de rasgos diferenciales correspondientes. Al clasificar documentos se realiza un

[16] www.crossref.org/

análisis de su contenido y forma, y se sitúan en grupos mediante un sistema de clasificación desarrollado con estos fines [34].

La *categorización* de documentos es un tipo de problema perteneciente a la familia de problemas asociados a encontrar agrupamientos entre objetos de cualquier tipo. Si bien la categorización de documentos tiene características particulares que surgen de las propiedades de los documentos como objetos a agrupar, los principios generales coinciden con los que se aplican para categorizar cualquier otro tipo de elementos [37].

El *agrupamiento* es descrito como una herramienta para el descubrimiento porque tiene la potencialidad de revelar relaciones basadas en datos complejos no detectadas previamente [12, 38, 39].

El agrupamiento es fundamental para una eficiente organización y recuperación de los documentos relevantes. A continuación, se realiza una panorámica sobre el agrupamiento de documentos, enfatizando las técnicas especializadas en los XML.

El agrupamiento de artículos científicos es un tema que comienza a tomar auge en los últimos años [40-43] en [40] se muestran los resultados de un estudio realizado sobre el agrupamiento automático de texto aplicado a artículos científicos y texto periodísticos en portugués brasileño. También se reportan varios trabajos que hacen uso de las citas textuales en los documentos en función de lograr un mejor agrupamiento de los artículos científicos. Una de las obras pioneras en este tema fue la desarrollada por Garfield en [44] en la cual se analiza el vínculo entre las citas en los artículos académicos. Otro de los trabajos relacionados con este tema se desarrolla en [16] donde demuestran que las citas textuales proveen sinónimos relevantes y vocabulario relacionado, los cuales ayudan a incrementar la efectividad del modelo bolsa de palabras [45].

Realizar un proceso de agrupamiento es muy difícil, ya que en dependencia de un objetivo hay que tener en cuenta tanto la información sintáctica o la semántica. Por tales motivos, es de gran importancia la correcta selección de un modelo de representación de texto que tenga en cuenta ambos aspectos. El modelo espacio vectorial [46] es uno de los más utilizados [47-49], principalmente por su eficiencia para determinar similitud entre unidades textuales. En la Tabla 1.1 se muestran algunas técnicas para la representación.

Tabla 1.1 Diferentes modelos de representación textual

Modelo		Características
VSM	Ventaja	Eficiencia para determinar similitud entre unidades textuales. Gran flexibilidad, que consiste en la posibilidad de utilizar distintos esquemas de pesos y distintas funciones de similitud.
	Deficiencia	Se basa en una comparación estricta de los términos, la eficacia se ve afectada por palabras distintas que describen el mismo concepto (sinonimia) y por palabras con distintos significados (polisemia).
LSA	Ventaja	Supera dificultades semánticas, generadas por la sinonimia y la polisemia, utilizando una técnica del algebra lineal conocida como descomposición de valores singulares. Basado en "bolsa de palabras"; es decir, que el orden de las palabras es irrelevante.
	Deficiencia	Definición de un mecanismo para seleccionar la cantidad de dimensiones, en casos donde los valores propios son muy pequeños. Suficientemente grande como para reflejar la estructura real de los datos - el contenido conceptual de los documentos- Relativamente pequeño, para obtener los efectos deseados de la eliminación de ruido.
PLSA	Ventaja	Refleja el contenido semántico de los documentos basado en un modelo generativo conocido como "Modelo de Aspecto", realiza reducción de dimensiones. Basados en "bolsa de palabras"; es decir, que el orden de las palabras es irrelevante.
	Deficiencia	El costo computacional aumenta con el número de documentos que se utilizan en el entrenamiento de dicho modelo. Este problema es resuelto en otro de los modelos probabilísticos existentes: LDA
LDA	Ventaja	Partiendo de la "bolsa de palabras" toma en consideración la propiedad de intercambiabilidad tanto para las palabras como para los documentos.
	Deficiencia	Falla para modelar directamente la relación entre la aparición de temas

Para realizar las representaciones necesarias en este trabajo, se tuvo en cuenta el modelo Espacio Vectorial, ya que se realiza una comparación léxica. Este modelo no realiza una comparación semántica.

1.2.1 Clasificación de las técnicas de agrupamiento

Para realizar análisis de grupos se han propuesto una gran variedad de algoritmos de agrupamiento. Estos pueden clasificarse de diversas formas [12] atendiendo a: 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 la Figura 1.3 se muestra una taxonomía de algoritmos de agrupamiento donde se distinguen dos tipos: los que forman particiones y los jerárquicos.

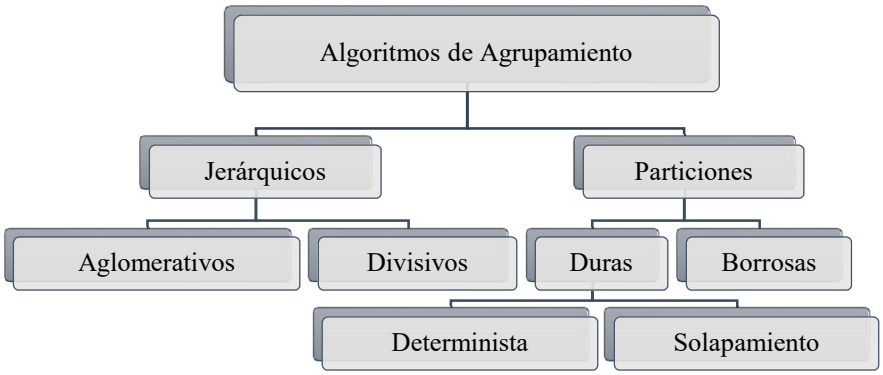


Figura 1.3 Taxonomía de algoritmos de agrupamientos

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. Uno de los algoritmos perteneciente a esta clasificación y que se usa ampliamente es el *k-medias* [50] y su extensión para dominios textuales *SKWIC* [51].

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 une, hasta que todos los objetos formen un único grupo, generalmente considerando una medida de distancia, en este grupo de algoritmos se encuentra el *K-Star* [52]. Mientras que los divisivos (top-down) consideran inicialmente que existe un único grupo al cual pertenecen todos los objetos y sucesivamente dividen 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 [53, 54] 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 [55]. 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, entre estos se encuentran los algoritmos *Estrellas* [56] con sus variantes *Estrella extendido* y *Estrella Generalizado* [57]. Las borrosas se subdividen en probabilísticas y posibilistas; una variante borrosa del algoritmo *SKWIC* se puede encontrar en Frigui (2001) [51].

Otros tipos de algoritmo se describen a continuación:

Algoritmos basados en árboles: Como su nombre lo indica, crea un árbol con los documentos de acuerdo a determinados criterios; de esta forma, los nodos del árbol resumen las características de los documentos que contienen. Ejemplos de este tipo de algoritmo son el STC [58] y el DC-tree [59]. El STC, presenta pasos extremadamente costosos respecto a la cantidad de términos de los documentos, por lo que solo ha sido probado con documentos de pequeña dimensionalidad como los snippets que no sobrepasan las 35 palabras. En el caso del algoritmo DC-tree, requiere prefijar valores para nueve parámetros.

Algoritmos basados en densidad: Los algoritmos de este tipo, dado un conjunto de datos *D*, definen un criterio de densidad para un grupo y tratan de encontrar grupos o divisiones de este conjunto de datos de forma tal que las densidades de estas divisiones sean las más cercanas posibles. En el mejor de los casos, son capaces de detectar automáticamente el número de grupos *k* en los que se deben dividir los datos, así como son capaces de descubrir grupos de diferentes formas y tamaños. Un ejemplo de estos algoritmos es el MajorClust [60]. Este algoritmo ha sido utilizado para agrupar colecciones de resúmenes [61] así como documentos de mayor tamaño [62].

Algoritmos basados en técnicas genéticas: Los algoritmos genéticos son heurísticas que ayudan a solucionar problemas que por vías analíticas serían difíciles de resolver, debido al

costo computacional involucrado. Este tipo de heurísticas trata de emular el comportamiento evolutivo de los seres vivos a través de procesos como cruzamiento, mutación o selección natural. Ejemplos de algoritmos de agrupamiento de documentos que utilizan estas técnicas se pueden apreciar en [63] y [64].

Atendiendo a la clasificación anterior y tomando como base facilidades de implementación, considerando eficacia y complejidad computacional; los métodos *SKWIC*, *F-SKWIC*, *K-Star* y *G-Star* resultan a criterios del autor adecuados para la evaluación del agrupamiento en el contexto de estudio. Por su parte, trabajos reportados en INEX[17] como Pinto (2009) [65] utilizan *K-Star*, y a los efectos de comparar la propuesta de metodología para el agrupamiento, constituye el referente a seguir.

Según Aggarwal [32], varias técnicas de agrupamiento se basan en funciones de similitud (o distancia). Una medida de similitud o función de similitud es una función real que cuantifica la similitud entre dos objetos. Toda función de similitud (o de distancia) calcula un valor que permite obtener una medida de proximidad o distancia entre dos documentos dados. Habitualmente es más fácil trabajar con atributos de dominio continuo, como puede ser un dominio numérico, que trabajar con dominios discretos como son los atributos con valores nominales [66].

Para que una medida de similitud pueda ser convertida en una medida de distancia debe cumplir con las propiedades de no negatividad, identidad simetría y desigualdad triangular [66], además, dada una similitud *S*:

$$S_{x,y} \leq S_{x,x} \quad \forall x, y \; con \; igualdad \; solo \; cuando \; x = y$$

$$S_{x,y} \in [0,1]$$

Algunas de las similitudes y distancias más utilizadas para comparar objetos son: la distancia *Euclidiana* [34], distancia *Minkowski* [36], Correlación de *Pearson* [67], entre otras. Entre las funciones más conocidas para el trabajo con colecciones textuales se encuentran: *Coseno* [68], *Jaccard* [37] y *Dice* [69]. En el Anexo 1 se muestra una selección de similitudes o distancias más usadas para comparar objetos. Existen distancias especializadas en la estructura de los documentos. En [1] se trabaja con la distancia de *edición de árboles* y sus

[17] Initiative for the Evaluation of XML Retrieval (http://inex.mmci.uni-saarland.de/)

modificaciones; una variante similar al coeficiente *Jaccard*, es propuesta en [70] para calcular distancia entre grafos.

La elección de una medida de similitud adecuada no es trivial y el desempeño de muchos algoritmos depende de la selección de una buena función que se acoja a los datos [34]. Para encontrar los agrupamientos naturales, la noción de similitud debe ser adaptada al problema particular; es por ello que actualmente se trabaja en la obtención de medidas que trabajen sobre tipos de datos específicos.

1.2.2 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 [13, 14], considerando los documentos como una bolsa de palabras; sin embargo, un buen proceso de agrupamiento no puede descartar el uso de la estructura [15, 71], algoritmos que combinan ambas componentes: estructura y contenido, constituyen un nuevo desafío, dada su gran complejidad [11].

1.2.2.1 Algoritmos que utilizan solo la estructura de los documentos

La estructura jerárquica de los documentos XML juega un papel importante en el agrupamiento [72]. Tener en cuenta solo esta, tiene aplicaciones interesantes para la extracción de información, la integración de datos heterogéneos, entre otras [3]. 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 de árbol correspondiente.

Una de las variantes para comparar árboles es utilizar la distancia de *edición de árboles* (TE), que intenta transformar un árbol A_1 en un árbol A_2, realizando una secuencia de operaciones; mientras menor sea la cantidad de operaciones necesarias en la transformación, mayor será la similitud entre los árboles correspondientes a los documentos comparados. Varios trabajos utilizan esta distancia [1, 73-78] 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[18], por lo que se han propuesto ideas como el cálculo de *Resúmenes Estructurales* [1] para reducir estos inconvenientes y luego se aplica el cálculo de la distancia TE. En la **Figura 1.4** se muestra el proceso de transformación de un árbol $T1$ a un árbol $T2$.

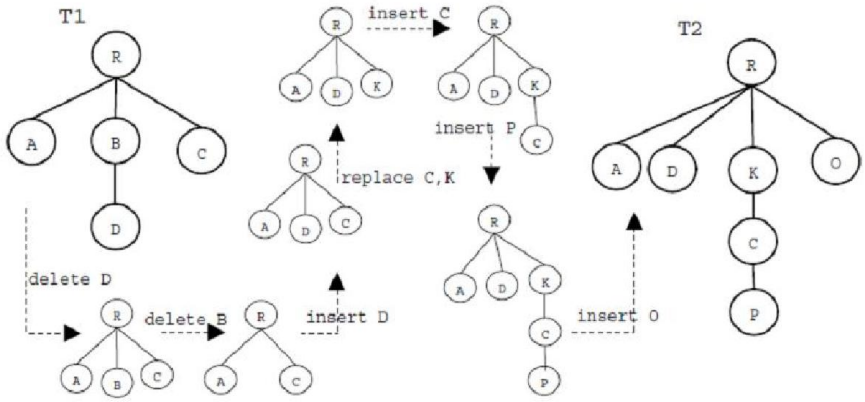


Figura 1.4: Uso de la distancia *edición de árboles*[19]

Otra variante es a través del cálculo de los *subárboles cercanos frecuentes* [79]. Basado en la estructura de los árboles correspondientes a los documentos y que no utiliza la distancia TE se presentó por [80], donde introducen el concepto de grafo estructurado (*s-graph*) de un documento *D*, el cual es muy similar a otra herramienta introducida por [81] para datos semiestructurados. Los trabajos [75, 76] se basan en técnicas de programación dinámica y el uso de *Grafos de edición* respectivamente, utilizan solo la estructura de los documentos para realizar agrupamientos. En [82] se utiliza la TE para comparar los DTD de los documentos. En [83] se aplica un razonamiento basado en casos para integrar conocimientos previos para calcular una mejor similitud en los documentos.

En [84], se propone un nuevo enfoque jerárquico que permite considerar múltiples formas de componentes estructurales en documentos XML. En cada nivel de la jerarquía resultante, los grupos son divididos considerando algún tipo de componentes estructurales que mantenga las diferencias estructurales de los documentos XML. En [85] se propone SOS, un método de búsqueda de similitud basado en estructuras y estilos de documentos *office*. Este método

[18] Mignet, L., et al., "The XML web: a first study".

[19] Tomado de Dalamagas, T., et al., "A Methodology for Clustering XML Documents by Structure".

calcula valores de similitud entre múltiples pares de archivos XML incluidos en los documentos *office*.

El método para el agrupamiento de documentos XML propuesto en [86], tiene como objetivo agrupar documentos que comparten estructuras similares, siguiendo un enfoque en dos etapas. En primer lugar, se extraen automáticamente la estructura de cada documento XML para ser clasificado. La estructura extraída es utilizada como un modelo de representación para clasificar el documento XML correspondiente.

Algunos algoritmos tratan la estructura de los documentos como grafos, estos se dividen en dos tipos fundamentalmente: basados en nodos y basados en grafos. Los métodos basados en nodos tratan de agrupar con el uso de distancias o similitudes basadas en los vértices de los grafos a comparar. Los algoritmos basados en grafos utilizan la estructura de estos como un todo y calculan similitudes entre grafos; esto es un mayor reto que el de los algoritmos basados en nodos pues necesitan igualar toda una estructura. Una descripción detallada de estos métodos se puede ver en Aggarwal (2010) [87].

Las redes neuronales artificiales también han sido utilizadas para realizar agrupamiento considerando solo la estructura de los documentos. Un ejemplo de estos métodos y que utiliza los grafos para agrupar XML es el *Graph Self Organizing Map* (GraphSOM) [88] que permite codificar la estructura en forma de grafo pero presenta pobre calidad en colecciones de la Wikipedia[20].

1.2.2.2 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 [79]. A continuación, se mencionan algunos trabajos existentes en la literatura.

[20] https://www.**wikipedia**.org/

Una primera variante muy sencilla es mezclar en una representación Espacio Vectorial (Vector Space Model*;* VSM) [46] el contenido y las etiquetas del documento y aplicar un algoritmo de agrupamiento conocido.

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.

Otros trabajos realizan extensiones a la representación VSM, llamadas C-VSM y SLVM [89-92]. 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 [93] se propone la similitud de *transportación proporcional*, donde se trabaja con comparaciones pesadas según la semejanza o no de los elementos a comparar en dos documentos.

Otro enfoque se muestra en [94], donde se realiza primeramente un agrupamiento teniendo en cuenta solo la estructura de los documentos, posteriormente proponen el uso del *Núcleo Semántico Latente* [95] para determinar la similitud entre el contenido de los documentos y se realiza un agrupamiento teniendo en cuenta el contenido.

En [72] desarrollaron el XCLSE que es una modificación al algoritmo de agrupamiento que utiliza solo la estructura XCLS [96], e incorpora una comparación a nivel semántico antes de realizar el agrupamiento. Esta propuesta no mejoró significativamente los resultados

alcanzados por XCLS, por el contrario realizó un importante esfuerzo de cálculo para la semántica de los datos.

En [97] se propone un marco de trabajo para tratar las similitudes estructurales y semánticas en documentos XML. Este marco de trabajo consta de cuatro módulos principales para descubrir similitudes estructurales entre los subárboles, para esto identifican el subárbol de semejanzas semánticas, calculan los costos de operación de edición basados en los árboles, y calculan la distancia TE. En [65] técnicas de clasificación no supervisadas son utilizadas con el fin de agrupar documentos de una gran dimensión. Los autores realizaron este enfoque mediante el uso del algoritmo de agrupamiento iterativo *K-Star* [52], en un proceso de agrupamiento recursivo sobre subconjuntos de la colección completa.

A modo de resumen de los métodos tratados, en la Tabla 1.2 se muestran algunos de los métodos propuestos.

1.3 Evaluación de los resultados del agrupamiento

Uno de los retos fundamentales cuando se realiza un agrupamiento es cómo evaluar los resultados obtenidos por el algoritmo utilizado [84]. Considerando las variaciones que ocurren en el resultado del agrupamiento a partir de características de los datos, las diferentes técnicas de análisis de grupos y la dependencia de algunas de estas técnicas de la definición de parámetros, es necesaria una evaluación de los resultados del agrupamiento, para de este modo poder medir la calidad del mismo. Una práctica común en tal sentido, es aplicar medidas de validación de grupos [85].

Una medida de validación de grupos es una función que hace corresponder a un agrupamiento un número real, de este modo se indica en qué grado el agrupamiento es correcto o no [55]. Existen varias medidas para la evaluación de los resultados del agrupamiento, cada medida existente no logra captar todas las propiedades estructurales deseadas al evaluar el agrupamiento (por ejemplo, densidad, cohesión, compactación, separación). Por tal motivo, los expertos pueden considerar los mejores resultados determinados por varios índices de validación y seleccionar aquel que mejor se ajuste a sus demandas.

Tabla 1.2 Resumen de algunos de los métodos existentes para agrupar documentos XML

Solo contenido	Kurgan, L. *Semantic mapping of xml tags using inductive machine learning*	Usan alguna variante de VSM.
	Shen, Y. *Clustering schemaless xml document*	
Solo estructura	Dalamagas, T. *A Methodology for Clustering XML Documents by Structure*	Utilizan la representación de árboles del XML para calcular alguna variante de la distancia de edición de árboles.
	Flesca, S. *Fast detection of XML structural similarities*	
	Lesniewska, A. *Clustering XML documents by structure*	
	Chawathe, S.S. *Comparing Hierarchical Data in External Memory*	Consideran la estructura de los XML basados en el uso de grafos de edición.
	Costa, G. *Hierarchical clustering of XML documents focused on structural components*	Proponen un nuevo enfoque jerárquico basado en la estructura.
	Aïtelhadj, A. *Using structural similarity for clustering XML documents*	Utilizan un enfoque en dos pasos para realizar el agrupamiento de los documentos.
Contenido y estructura	Kutty, S. *Combining the structure and content of XML documents for clustering using frequent subtrees*	Hacen uso de los subárboles frecuentes cercanos.
	Yang, W. *A semi-structured document model for text mining*	Analizan una variante de comparación de documentos XML basados en VSM.
	Tekli, J.M. *A Novel XML Document Structure Comparison Framework based-on Subtree Commonalities and Label Semantics*	Proponen un marco de trabajo para tratar con el contenido y la estructura, trabajando esta última con la distancia de edición de árboles.
	Pinto, D. BUAP: Performance of K-Star at the INEX'09 Clustering Task	Utilizan un algoritmo iterativo (*K-Star*) en un proceso de agrupamiento recursivo.

1.3.1 Clasificación de las medidas de validación

Una clasificación muy usada divide la validación del agrupamiento en: medidas internas y medidas externas. Las medidas internas evalúan considerando solamente los resultados del agrupamiento en términos de cantidades que involucran los vectores de datos, ejemplo de estas son: Similitud global (*Overall Similarity*) [98], Índices *Dunn*, Índice *Davies – Bouldin* [99], Modularidad (*Modularity*) [100], entre otras. Las medidas externas evalúan el resultado con respecto a una estructura pre-especificada [84, 101]. A continuación se muestran una selección de medidas externas, este tipo de medidas fue la escogida en este trabajo para evaluar los resultados del agrupamiento, pues se cuenta con la clasificación de referencia de los documentos seleccionados para el estudio.

1.3.2 Medidas externas

El uso de medidas externas necesita de la existencia de una clasificación de referencia, es decir, un criterio externo que es impuesto sobre los datos. Estas medidas juegan un rol importante en la validación de resultados de agrupamiento, sobre todo, proporcionan el criterio externo requerido para establecer comparaciones con los resultados de medidas internas.

Una medida externa es la entropía [102], la cual es una función de la distribución de las clases en los grupos resultantes. La entropía total para un conjunto de grupos es calculada como la suma de las entropías de cada grupo, ponderadas con el tamaño del grupo [103].

Dos medidas muy usadas en la recuperación de información [21, 86, 97] son precisión (*Pr*) y cubrimiento (*Re*). La primera indica qué parte de los documentos recuperados es correcta, y la segunda hace referencia a qué parte de los documentos que debían ser recuperados lo fueron. Estas medidas son adaptadas a la validación del agrupamiento. Precisión y cubrimiento se calculan teniendo en cuenta las expresiones: $Pr(i,j)=n_{ij}/n_j$ y $Re(i,j)=n_{ij}/n_i$ donde j e i son el grupo y la clase dados, respectivamente; 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 calcula como la media armónica entre precisión y cubrimiento, la influencia de precisión y cubrimiento en su cálculo depende de un umbral $\alpha(0\leq \alpha \leq 1)$ [104].

La medida-*F* global (*Overall F- measure*; *OFM*), se calcula usando el promedio ponderado de los valores máximos por clase de la medida-*F* sobre todos los grupos [105].

Variantes de precisión y cubrimiento, *micro-averaged precision* y *micro-averaged recall* [106], son utilizadas para evaluar el agrupamiento [107], las expresiones para su cálculo coinciden si cada objeto pertenece a sólo un grupo y la clasificación de referencia también tiene una única clasificación para cada objeto.

Otras medidas externas y utilizadas en INEX, se basan en el cálculo de *Purity*: *Micro-Purity* y *Macro-Purity* [108, 109]. El criterio *Purity* se utiliza para determinar la calidad de los grupos, 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. En general, mayores valores de *Purity*, reportan mejores resultados del agrupamiento [65].

Las expresiones de las medidas mencionadas en este subepígrafe se pueden observar en el Anexo 2. En este trabajo para realizar la evaluación se proponen las medidas *Overall F-Measure*, y el criterio *Purity* (*micro-Purity* y *macro-Purity*).

El agrupamiento de documentos semiestructurados que combinen estructura y contenido constituye una necesidad para contribuir a la organización de documentos debido al crecimiento continuo de información hace prácticamente imposible su gestión mediante la aplicación de métodos tradicionales [110]. Sin embargo, se requieren de herramientas, plataformas que faciliten dichos procesos.

1.4 Herramientas para la manipulación de documentos

Los investigadores de esta área del conocimiento se han dado a la tarea de confeccionar marcos de trabajos, APIs y herramientas para la manipulación de grandes colecciones textuales, entre estas se encuentran: Lucene[21], Tika[22] y SolR[23].

[21] https://lucene.apache.org/core/

[22] https://tika.apache.org/

[23] lucene.apache.org/solr/

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.

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 [111]. 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.

A modo de resumen *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 cuando se manipulan documentos XML 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 [112]. Adicionalmente, *Analyzer* obtiene las raíces de las palabras mediante heurísticas, y tratar la sinonimia y polisemia.

1.4.2 Tika

Apache Tika es un marco de trabajo de código abierto enfocado en el procesamiento automático de documentos. Específicamente está encaminado a la identificación de diferentes tipos de formatos, detección del idioma, así como extracción de información textual y metadatos en los documentos [111]. *Tika* posee una arquitectura extensible y modular, así como una gran flexibilidad para tratar los diferentes modelos de metadatos existentes.

Una de las principales ventajas de esta herramienta es su extensibilidad, pues su arquitectura permite adicionar tantos analizadores como nuevos tipos de documentos se quieran tratar, por defecto *Tika* contiene varios analizadores para tratar los documentos más utilizados actualmente, como son: *.doc*, *.pdf* y *.txt*. En caso de no especificarse el analizador para tratar un documento determinado lo detecta automáticamente siempre que lo tenga incorporado.

1.4.3 Solr

Solr es un motor de búsqueda de código abierto basado en la biblioteca *Java* del proyecto *Lucene*, con APIs en XML/HTTP y JSON. Es una herramienta diseñada para trabajar con grandes volúmenes de documentos, puede almacenar millones de ellos en sus índices. Debido a esto posee un gran nivel de optimización en la ejecución de las consultas y en el almacenamiento de los documentos que contienen texto, como son: correos electrónicos,

páginas web y documentos PDF [113]. Una de sus potencialidades más importantes es que los resultados de las consultas se devuelven ordenados de acuerdo a la relevancia. Otra de sus ventajas es su escalabilidad, pues puede correr sobre un *grupo* de computadoras con múltiples servidores. Por otra parte, *Solr* es fácil de instalar y contiene una configuración de ejemplo que facilita lograr una rápida familiarización con el mismo.

Solr cuenta con un archivo de configuración nombrado *schema.xml* donde deben estar definidos todos los posibles campos a indexar. Este archivo contiene por defecto un conjunto de campos predefinidos, pero si se quiere incorporar nuevos campos se deben añadir las declaraciones correspondientes. En la Figura 1.5 se muestra un fragmento donde aparecen declaraciones de algunos campos comunes en diferentes documentos.

```
---------------------------------------------------------------------------
<field name="id" type="string" indexed="true" stored="true"
required="true" multiValued="false" />
<field name="name" type="text_general" indexed="true" stored="true"/>
<field name="url" type="text_general" indexed="true" stored="true"/>
<field name="title" type="text_general" indexed="true" stored="true"
multiValued="true"/>
---------------------------------------------------------------------------
```

Figura 1.5 Fragmento de la declaración de algunos campos de un documento

En la fase de búsqueda al igual que en el indexado se necesita establecer previamente la conexión con el servidor *Solr*, para luego enviar la consulta y obtener los resultados como una lista de documentos ordenada de acuerdo a la relevancia. Esta potencialidad de *Solr* permite establecer posteriormente un ranking para los grupos de documentos basado en el ranking general.

1.5 Conclusiones parciales del capítulo

Para almacenar la información de documentos en forma semiestructurada el lenguaje XML resulta recomendado, ya que posibilita la fácil lectura y edición de los documentos. Su amplia utilización queda evidenciada por la gran cantidad de documentos semiestructurados en este formato.

El desarrollo de herramientas que permitan organizar la información existente o recuperada, entre las que se encuentra el agrupamiento de documentos que combinen estructura y contenido, constituye una temática de actualidad

La selección de la función de similitud resulta esencial para lograr un buen agrupamiento. Dependiendo del problema se determina la función de similitud. Para el agrupamiento de documentos textuales se proponen funciones de similitud que se enfocan más al análisis de la similitud según el texto, sin embargo existen muchas propuestas de funciones para datos estructurados. Resulta necesario analizar sobre la base de combinar estructura y contenido, cómo analizar la similitud de documentos semiestructurados.

Los métodos *SKWIC*, *F-SKWIC*, *K-Star* y *G-Star* son eficaces y de fácil implementación, pudiendo ser representativos para recomendar una evaluación relativa a un agrupamiento. Al contar con las clasificaciones de referencia para evaluar los resultados de los agrupamientos las medidas externas: *Overall F-measure*, *Micro-Purity* y *Macro-Purity* resultan adecuadas para estos fines.

En la actualidad se desarrollan múltiples técnicas para el procesamiento de la información. La reutilización de código resulta primordial en el desarrollo de herramientas que involucren el trabajo con colecciones textuales de forma eficiente.

2

METODOLOGÍA PARA EL AGRUPAMIENTO DE DOCUMENTOS CONSIDERANDO ESTRUCTURA Y CONTENIDO

2. METODOLOGÍA PARA EL AGRUPAMIENTO DE DOCUMENTOS CONSIDERANDO ESTRUCTURA Y CONTENIDO

La mayoría de los métodos propuestos para agrupar documentos basados en las relaciones estructura o contenido no utilizan las conexiones entre ambas dimensiones; limitando el análisis de la similitud existente entre estos. En este capítulo se presenta: (1) una metodología 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 la metodología y (4) la evaluación de los resultados obtenidos por la metodología.

2.1 Metodología para el agrupamiento

Reflexionando brevemente sobre el concepto de documento, se pueden encontrar múltiples tipos que resultan 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), *conclusiones*, *referencias bibliográficas*, entre otras. Consecuentemente cada documento perteneciente a un conjunto $D=\{D_1,\ldots, D_m\}$, consta de un conjunto $UE=\{UE_1,\ldots, UE_n\}$, que se denominará Unidades Estructurales (UE). Por ejemplo, para el caso de los artículos científicos una unidad estructural es el *título*, otra el *resumen* y así sucesivamente.

La metodología que se propone para la aplicación del agrupamiento de documentos XML persigue el criterio plasmado anteriormente [22]. Se inicia a partir de la colección de documentos y como resultado se obtienen grupos homogéneos de documentos afines. Una visión gráfica del esquema de la metodología presentada en este trabajo se muestra en la Figura 2.1.

En la metodología se persigue la construcción de una matriz de similitud global, para lo cual es necesario realizar tres pasos fundamentales: (1) Construir una primera representación

denominada *Representación I*, utilizando las UE de los documentos, una representación por cada UE; (2) Crear una segunda representación, *Representación II*, considerando la colección completa; (3) realizar un agrupamiento previo utilizando *Representación I*. En la Figura 2.2 se muestra los tres módulos principales que contiene la metodología propuesta.

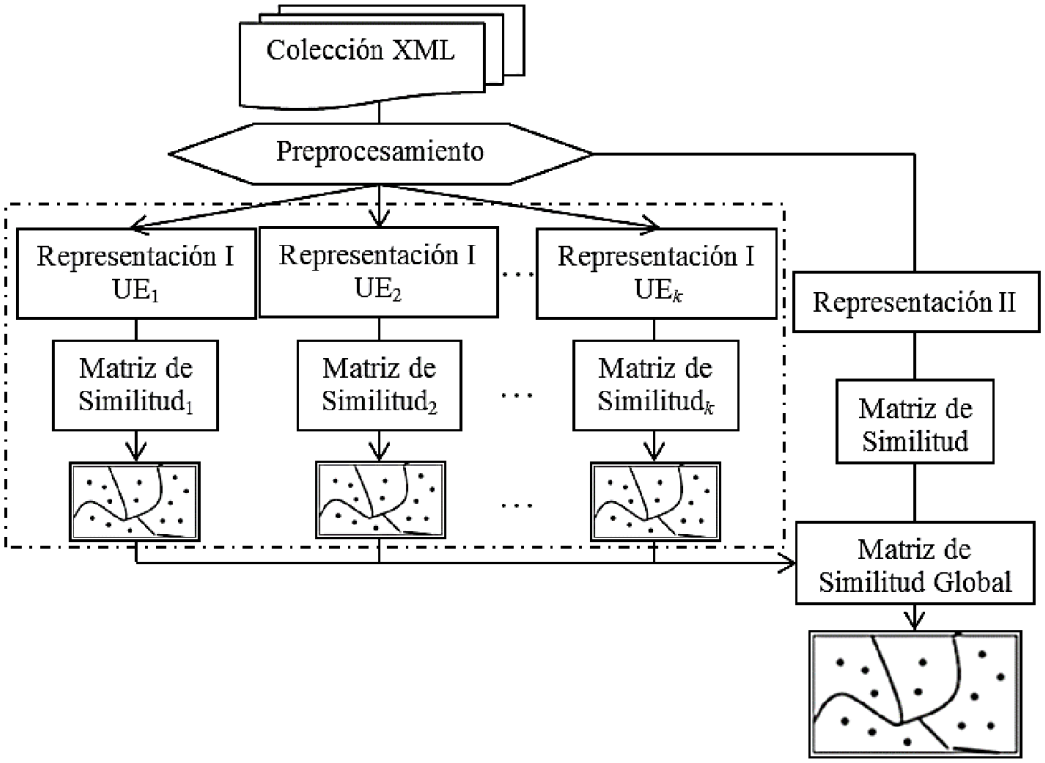


Figura 2.1 Esquema que muestra metodología propuesta[24]

2.1.1 Preprocesamiento del corpus textual

En la metodología propuesta se transforma el corpus convirtiendo los ficheros de entrada en una secuencia de tokens[25] de palabras. En el paso subsecuente a la extracción de términos, estos tokens se usan para generar rasgos significativos (índices de términos).

En esta etapa de la trasformación del corpus se determina en cuáles UE del documento se encuentran cada token, además se divide la colección original en *k* colecciones independientes, donde *k* es el número de UE en un documento.

[24] Tomado de Magdaleno Guevara, D.et al,. Clustering XML Documents using Structure and Content Based in a Proposal Similarity Function (OverallSimSUX).

[25] En este trabajo se tratarán indistintamente los vocablos términos, token o palabras.

Módulo 1. Preprocesamiento del corpus textual a procesar, identificando 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.

Figura 2.2 Módulos principales de la metodología propuesta

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 nuevos documentos DUE_k:

$$DUE_k = \{UE_k \in d, \forall d \in D\} \quad 2.1$$

donde *d* es un documento perteneciente a *D*, UE_k es la *k*-ésima *UE* de *d*.

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 que contienen caracteres alfa-numéricos, y sustituyendo las contracciones por sus expresiones completas [114]. La eliminación de palabras de parada o gramaticales [115-117], es una de las técnicas de selección utilizadas en este esquema de aplicación. 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.

2.1.2 Representación textual

Según Lanquillon [114], 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.

Para obtener las representaciones que utiliza la metodología, 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).

La representación se establece seleccionando los *m* mejores rasgos, considerando esencialmente las expresiones I y II de calidad de términos [118], mostradas en el Anexo 13 [119]. En esta etapa se genera un vector pesado para cada documento, basado en el vector de frecuencias de términos. En particular, se utilizó TF-IDF [118] (Ver Anexo 13), medida estadística de pesado utilizada frecuentemente en el procesamiento del lenguaje natural para determinar cuán importante es un término en un corpus. La importancia de cada término es proporcional al número de veces que aparece en el documento (TF), aunque es balanceada por la frecuencia del término en el corpus (IDF).

La representación VSM fue seleccionada por ser efectiva para representar documentos y ser ampliamente reconocida en la comunidad de minería de textos [120-122]. En un VSM cada documento se identifica como un vector de rasgos en un espacio en el cual cada dimensión corresponde a términos indexados distinto. Un vector documento, en cada componente tiene un valor numérico para indicar su importancia.

La metodología cuenta con dos tipos de representaciones (*Representación I*, una para cada UE y *Representación II*, tomando en consideración toda la colección).

En la *Representación I* los valores numéricos iniciales coinciden con la frecuencia de aparición absoluta de cada término en cada documento. En la Tabla 2.1 se muestra como para todo componente cada vector documento tiene un valor numérico que indica su importancia; este valor ($tf_{di}(t_j)$) no es más que la frecuencia de aparición del token t_i en el documento d_j.

Tabla 2.1 Matriz VSM, 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	$tf_{dn}(t_1)$	$tf_{dn}(t_2)$	…	$tf_{dn}(t_m)$

Por otra parte, se considera que un token tiene mayor o menor importancia para la comparación de dos documentos en dependencia del lugar que este ocupe dentro de los mismos [123]. 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.

En este trabajo se tiene en cuenta esta idea, logrando agregar 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 [123].

$$tf_{ij} = \sum_{k=1}^{n} (w_{kj} \times frecuencia_{ik}) \qquad 2.2$$

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

Donde n es la cantidad de UE presentes en el documento d_j, $frecuencia_{ik}$ es la frecuencia del término t_i en la UE_k y w_{kj} es el peso que se le calcula a UE_k en d_j. El cálculo del peso de la UE_k para cada 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 de d_j y pot es un valor suministrado.

En la **Figura 2.3** se muestra una gráfica con tres funciones que representan los valores de peso obtenidos por la ecuación 2.3, para valores de *pot* iguales a 2, 5 y 10 respectivamente. Como se puede observar, para valores muy pequeños de *pot*, la diferencia entre los pesos de las unidades estructurales pequeñas (*título*, *resumen*, etc) es imperceptible. Por otra parte, cuando los valores de *pot* son mayores, superior a 10, la diferencia entre los pesos de las unidades estructurales mayores (*Cuerpo*, *Referencias bibliográficas*, etc) comienza a ser menor. Los valores más discriminantes se obtuvieron para valores de *pot* 5 y 6. Por lo que en este trabajo se realizan todos los experimentos con *pot* igual 5.

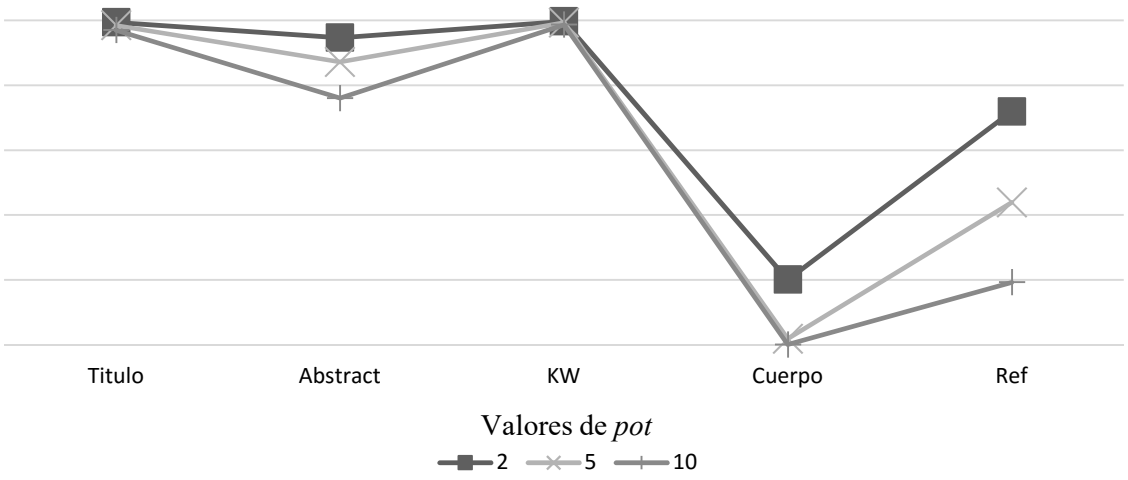


Figura 2.3 Análisis de la estimación del parámetro *pot* para el peso de las unidades estructurales

Finalmente, como resultado del proceso de extracción de términos en los submódulos 2.1 y 2.2, se obtienen *k* representaciones VSM clásicas (*Representación I*) y una representación VSM global que considera el contenido en función de la estructura (*Representación II*).

2.1.3 Función de similitud *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 [124].

Resolver este problema consiste en esencia hallar la estructura interna de los objetos en el espacio de representación inicial, 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 [124].

Definición 2.2 (β-semejantes) Dos descripciones[26] (objetos) $I(O_i)$, $I(O_j)$ se denominan β-semejantes si $\beta(O_i,O_j)>\beta_0$, considerando β_0 como el umbral de semejanza.

2.1.3.1 Agrupamiento de las k-colecciones

Un paso previo al cálculo de la función de similitud *OverallSimSUX* es el agrupamiento de las *k*-colecciones identificadas en la etapa de pre-procesamiento. Para cada *k*-colección identificada se realiza un agrupamiento independiente, utilizando la *Representación I*.

2.1.3.2 Descripción de la 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 estructurales. En este trabajo se propone una nueva medida de similitud que facilita capturar el grado de 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 que fueron representadas utilizando la *Representación I* y la matriz de similitud obtenida aplicando alguna medida de similitud (ver Anexo 1) a partir de la *Representación II*.

[26] Indistintamente se utiliza el término objeto y descriptores de objetos.

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

Definición 2.3 (λ-pertenencia) Dados los objetos *i*, *j* se define la λ-pertenencia como una relación de pertenencia booleana (toma el valor 1 si ambos objetos (*i*, *j*) pertenecen al mismo grupo *n*, en otro caso toma el valor 0) de ambos objetos a un mismo grupo, a partir de los resultados del agrupamiento de cada *Representación I*. Esta relación se formaliza en la ecuación 2.4.

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

Definición 2.4 (*OverallSimSUX*) Dado los objetos *i*, *j*, se define la medida de similitud normalizada *OverallSimSUX*, con valores en el intervalo [0,1], tal como se muestra en la ecuación 2.5.

$$S_{OSSUX}(i,j) = \frac{\sum_{k=1}^{n}\left(w_k \times \lambda_k(i,j)\right) + S_g(i,j)}{\sum_{k=1}^{n}(w_k) + 1} \qquad 2.5$$

Donde, $S_g(i,j)$ es un elemento de la matriz S_g, calculada a partir de la *Representación II* utilizando alguna medida de similitud; w_k es el peso de la UE_k; *n* cantidad de UE identificadas en los documentos. $\lambda_k(i,j)$ es el valor de *λ*-pertenencia de los documentos *i*, *j* resultante del agrupamiento realizado a la *Representación I* de la UE_k.

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

2.2 Un algoritmo de agrupamiento basado en la similitud OverallSimSUX

En esta sección se detallan el algoritmo para realizar un agrupamiento utilizando la matriz de similitud *OverallSimSUX*. La Figura 2.4 muestra los cuatro pasos del algoritmo.

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.

Figura 2.4. Algoritmo de agrupamiento utilizando *OverallSimSUX*

Construcción de la matriz de similitud *OverallSimSUX*

La función de similitud *OverallSimSUX* 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[27]. De este modo, es necesario determinar primeramente la matriz de similitud (aplicando alguna medida de similitud) 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.

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 (alguna medida de similitud afín reportada en la literatura para los agrupamientos simples o similitud *OverallSimSUX* para el agrupamiento general), se define una función booleana de semejanza de la siguiente manera:

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

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

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.

Determinación de los núcleos iniciales

La conformación de los núcleos iniciales del agrupamiento está determinada por el cálculo inicial del 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.

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 [119].

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 [124].

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

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

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

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

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

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

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

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; ecuación 2.10:

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

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

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

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

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

2.4 Procedimiento general para el agrupamiento de documentos XML

Como parte de la metodología de agrupamiento propuesta, 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 de la metodología. Cada módulo del procedimiento general se corresponde con los módulos mencionados en la metodología de agrupamiento, y se describen siguiendo el mismo orden.

Los módulos del procedimiento general son (observe el Anexo 4):

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 a la metodología. Para realizar el agrupamiento se utilizó el algoritmo *K-Star* clásico [52]; ya que no necesita conocer el número de grupos a crear pues los crea de forma totalmente no-supervisada. Es un algoritmo rápido y obtiene resultados buenos cuando se aplica en dominios textuales [125]. La función de similitud utilizada fue la *Coseno* [126], que ha sido utilizada para comparar documentos que son sometidos a un proceso de agrupamiento. No obstante, elegir otro algoritmo u otra función de similitud no

invalida la concepción de la metodología propuesta, más adelante se muestran los resultados de un estudio comparativo, analizando el desempeño de la metodología utilizando otras técnicas de agrupamiento y otras medidas de similitud.

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*[28] de *Java* destinada al trabajo con documentos XML, que permite identificar las UE que se incorporan al índice creado introduciéndose las facilidades de *Lucene*[29] [110].

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* [127], 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 [118] 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 [118].

[28] http://www.jdom.org/

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

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 y se utiliza como medida la similitud coseno, esta se muestra en la ecuación 2.13. Se genera un agrupamiento para cada UE a partir de la similitud asociada a la *Representación I*.

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

La matriz de similitud global se obtiene a partir del resultado de cada agrupamiento y la matriz de similitud asociada a la *Representación II*, usa como medida de similitud *OverallSimSUX*, ver ecuación 2.5. Finalmente el agrupamiento general es el resultado de aplicar el algoritmo de agrupamiento, ver Figura 2.4, que utiliza 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 implementó la medida externa *Overall F-measure* [98], basada en Precisión (Pr) y Cubrimiento[30] (Re) [104] y las propuestas por *INEX*[31] para la evaluación de técnicas de clasificación supervisadas y no supervisadas de documentos XML, basadas en *Purity*, *Micro-Purity* y *Macro-Purity* [65].

En la Figura 2.5 se muestra la combinación de los cuatro módulos que componen el procedimiento general descrito con anterioridad. En el Anexo 4 se muestra una gráfica con el modelo general para el agrupamiento.

[30] En este documento se utiliza cubrimiento como traducción de la medida recall.

[31] Initiative for the Evaluation of XML Retrieval.

```
Entrada: Colección D 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 */
   – Construir todas las k-colecciones (Colección D)
2. Para cada DSUk
   – Representación-I←Realizar_Representación-I(DSUk)
   – Matriz_Similitud←Similitud_Coseno(Representación-I)
   – Agrupamientos[ue]←Realizar_Agrupamientoue(Matriz_Similitud)
3. Fin_Para
4. Representación-II←Realizar_Representación-II, a toda la colección D, utilizando la ecuación 2.2
5. Matriz_Similitud-II←Similitud_Coseno(Representación-II)
6. Matriz_Sim_Global←OverallSimSUX(Agrupamientos,Matriz_Similitud-II)
7. Realizar Agrupamiento_General(Matriz_Similitud_Global)
8. Realizar Evaluación_Agrupamiento(Agrupamiento_General)
Fin
```

Figura 2.5 Procedimiento general para el agrupamiento usando *OverallSimSUX*

2.5 Complejidad computacional de la metodología propuesta

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 [128]. No obstante, nuevos enfoques que abordan el agrupamiento de documentos XML basados en estructura y contenido, como el referido por [94], tienen como desventaja su alto costo computacional, al proponer el uso del *Núcleo Semántico Latente* [95] para determinar la similitud entre el contenido de los documentos.

El algoritmo, 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 $O(kn^2)$ [52]. 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 Evaluación de los resultados de la metodología

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" [129]. En este epígrafe, se presentan los resultados de los experimentos diseñados para evaluar la metodología de agrupamiento propuesto en esta investigación [130].

Para chequear la validez de los resultados obtenidos se han diseñado tres experimentos, aplicados a tres casos de estudio; con el propósito de realizar un análisis estadístico que permita verificar: (1) que la metodología no influye en el comportamiento de los algoritmos de agrupamiento seleccionados para aplicar la metodología, (2) Los resultados obtenidos por los algoritmos de agrupamiento son en su mayoría mejores, cuando estos utilizan la matriz resultante de la función de similitud *OverallSimSUX*, que cuando utilizan funciones de similitud clásicas.

2.6.1 Sistema para la evaluación de la metodología

Para realizar la evaluación de la metodología propuesta, se implementó el sistema para el análisis de algoritmos agrupamientos documental utilizando *OverallSimSUX* (*XMLearning*) [131]. El sistema implementa el procedimiento general descrito en la sección 2.4.

El diseño del sistema *XMLearning* se dividió en tres capas fundamentales como se muestra en la Figura 2.6. 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 2.6 Diseño general del sistema *XMLearning*

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 a los algoritmos de agrupamiento que operan sobre estos datos. 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.

Entre las bondades que ofrece *XMLearning* se puede destacar que es posible ejecutar varios algoritmos de los que tiene implementados; de este modo es posible ver en una sola ventana los resultados obtenidos, así como los valores de las medidas de evaluación para cada uno de estos, logrando una comparación más cómoda.

2.6.2 Algoritmos seleccionados para realizar la evaluación

La evaluación se realizará a través de un estudio comparativo con cuatro algoritmos de agrupamiento de distintas características, (1) *K-Star*, aglomerativo; (2) *Generalized Star* (*G-Star*), basado en grafos; (3) *Simultaneous Keyword* Ident*ification and Grouping of text documents* (*SKWIC*), de particiones duras y determinísticas; (4) *Fuzzy SKWIC*, particiones borrosas. En [130] se muestra una breve descripción de cada algoritmo.

2.6.3 Definición de los casos de estudio para la aplicación de la metodología

- Caso de estudio 1: Documentos recuperados del sitio de información científico técnica del Centro de Estudios de Informática (CEI) de la Universidad Central "Marta Abreu" de las Villas (UCLV). http://ict.cei.uclv.edu.cu.

- Caso de estudio 2: Selección de documentos del repositorio IDE-Alliance. Suministrado por la Universidad de Granada, España.
- Caso de estudio 3: Selección de documentos de la colección de Wikipedia, publicados por "Iniciativa para la evaluación de XML recuperados", INEX'09. http://inex.mmci.uni-saarland.de/data/documentcollection.html

Se conformaron 16 colecciones de documentos XML. Las colecciones de la 1 a la 7 corresponden al primer caso de estudio, el promedio de cada colección es de 100 documentos; las colecciones de la 8 a la 11 con documentos correspondientes al caso de estudio 2, la cantidad de documentos promedio en estas colecciones es de 500 documentos; el resto de las colecciones son documentos escogidos del tercer caso de estudio, estas colecciones tienen como promedio 1 000 documentos. En el Anexo 55, se muestra un fragmento de un documento XML correspondiente al corpus 9. En el proceso de agrupamiento no se tiene en cuenta la etiqueta *class*, esta etiqueta es utilizada solamente en el proceso de evaluación.

2.7 Validación del agrupamiento

La validación del agrupamiento se conoce por el procedimiento de evaluar los resultados de algoritmos de agrupamiento [132, 133].

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).

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 [136]. 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.

2.7.1.1 Criterios para la selección del umbral

Las variaciones en el umbral de similitud permiten restringir o no el conjunto de objetos más representativos para caracterizar los grupos. Una de las variantes para la selección del umbral 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 2.7 se observan histogramas de las diferentes frecuencias, atendiendo al criterio para el cálculo del umbral.

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[32]. En el Anexo 66 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.

[32] Se utilizó SPSS 13.0 para Windows

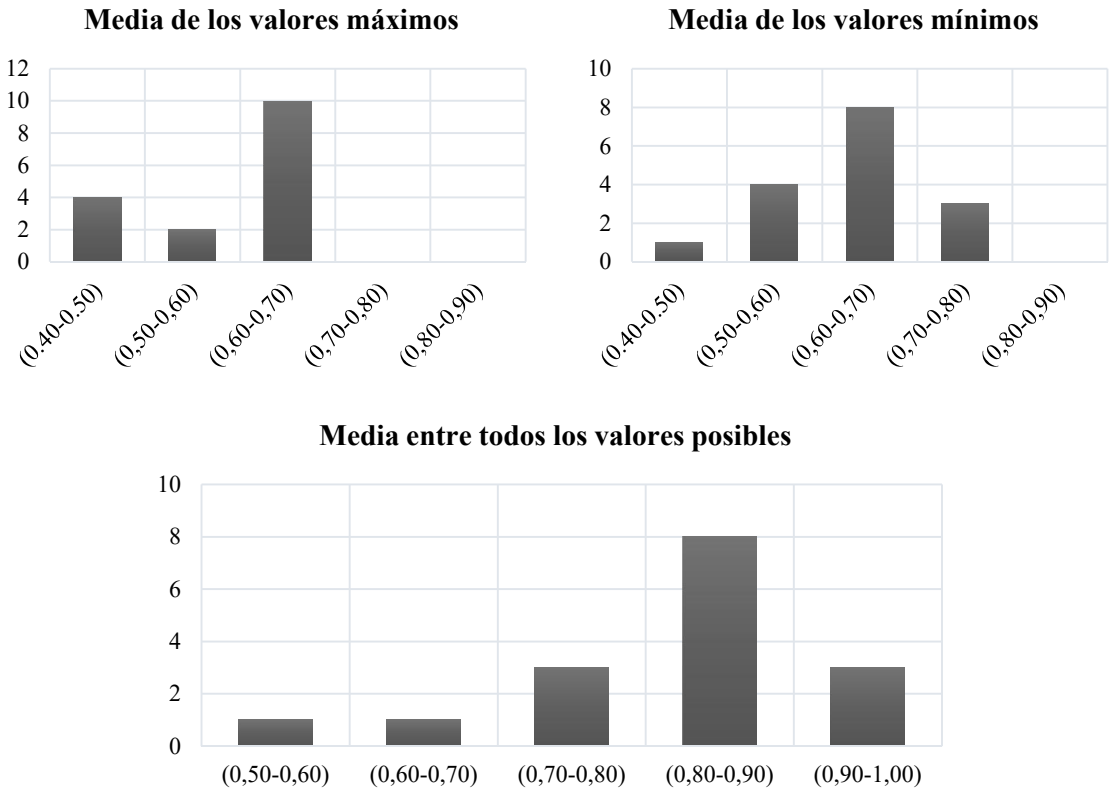


Figura 2.7 Histograma de frecuencias de calidad del agrupamiento basadas en OFM. Procedentes de los casos de estudio 1 y 2

2.7.1.2 Diseño de los experimentos y resultados

El primer experimento consiste en verificar cómo se comportan globalmente, sobre los tres juegos de datos descritos anteriormente, los cuatro algoritmos seleccionados, sin utilizar la metodología propuesta. La Tabla 2.2 muestra estos resultados.

El segundo experimento se realiza para verificar cómo se comporta globalmente, sobre las 16 colecciones, los cuatro algoritmos seleccionados, usando la metodología propuesta en este trabajo. Los resultados de este experimento se muestran en la Tabla 2.3.

En el primer y segundo experimento los resultados son muy similares. Los mejores resultados fueron obtenidos por el algoritmo *F-SKWIC*, que logra mejores o iguales resultados que los otros algoritmos en el 56,25% de los casos, para los dos experimentos. El segundo mejor algoritmo en ambos experimentos fue el *K-Star*, con resultados iguales o mejores que los otros algoritmos en el 31,25% de los casos. Siguiendo esta idea para el análisis, los resultados arrojados en ambos experimentos coinciden en 12 de las 16 colecciones. Por lo que se puede

concluir que el uso de la metodología no cambia el desempeño de los algoritmos, es decir, si un algoritmo obtuvo el mejor desempeño para un corpus sin utilizar la metodología, cuando esta se utiliza, el algoritmo tendrá el mejor desempeño también.

Tabla 2.2 Valores de OFM para los cuatro algoritmos cuando la metodología no es utilizada

C	K-Star	SKWIC	G-Star	F-SKWIC
1	0,6943	0,5300	0,6400	**0,7214**
2	0,6000	0,4955	**0,6650**	0,6138
3	**0,6825**	0,4634	0,6050	0,6670
4	0,4717	0,3524	0,4418	**0,4785**
5	0,6753	0,2858	0,4604	**0,8307**
6	**0,4745**	0,2248	0,3325	0,4739
7	**0,5813**	0,3325	0,4850	0,4624
8	**0,8990**	0,5530	0,7952	0,7453
9	0,6249	0,3307	0,5146	**0,8925**
10	0,6461	0,4892	**0,7650**	0,6787
11	0,7715	0,5028	0,7922	**0,8819**
12	**0,9239**	0,4393	0,7145	**0,9239**
13	0,9582	0,5810	0,6289	**0,9601**
14	0,9047	0,3453	0,5091	**0,9839**
15	0,8830	0,3207	0,5914	**0,9139**
16	0,6682	0,8883	**0,9364**	0,7388

Tabla 2.3 Valores de OFM para los cuatro algoritmos cuando la metodología es utilizada

C	K-Star	SKWIC	G-Star	F-SKWIC
1	0,8204	0,6221	0,7463	**0,8524**
2	0,7418	0,4746	0,5199	**0,7589**
3	**0,8564**	0,4825	0,5384	0,8369
4	0,7100	0,4337	0,4599	**0,7202**
5	0,6371	0,2792	0,4663	**0,7837**
6	**0,6861**	0,2904	0,3880	0,6852
7	**0,7322**	0,3181	0,4716	0,5824
8	0,8862	0,5821	**0,8981**	0,8815
9	0,8520	0,4003	0,5602	**0,8862**
10	0,8147	0,4884	**0,8772**	0,8558
11	**0,8952**	0,5731	0,8874	0,8741
12	**0,9471**	0,4464	0,7198	**0,9471**
13	0,9526	0,6148	0,7112	**0,9545**
14	0,8885	0,3443	0,5154	**0,9663**
15	0,7998	0,3105	0,6150	**0,8278**
16	0,7485	**0,9412**	**0,9412**	0,8276

Con el objetivo de probar lo anteriormente concluido, para los dos primero experimentos se realizó el test de *Friedman* [137]. Este test realiza un procedimiento de comparación múltiple capaz de detectar diferencias significativas entre los comportamientos de dos o más algoritmos; ya que descubre si al menos dos muestras representan poblaciones con valores diferentes a la media, en un conjunto de n ejemplos ($n \geq 2$). En la Figura 2.8 se muestran los valores de ranking medios y la significación (*p-value*) asociada a este test para cada algoritmo, sin usar la metodología y cuando esta es usada, respectivamente.

Usando un nivel de significación de 0,05, correspondiente a un intervalo de confianza del 95%, el test de *Friedman* (con *p-value*$<0,05$) sugiere rechazar la hipótesis nula, esto es, que existen diferencias significativas entre al menos dos algoritmos. Además es posible observar que los algoritmos *F-SKWIC* y *K-Star* son los mejores rankeados; sin embargo, esta información no puede ser usada para concluir la afirmación previa [131]; es por esto que se aplicó el test post-hoc *Nemenyi* [138] después de aplicar el test de *Friedman*.

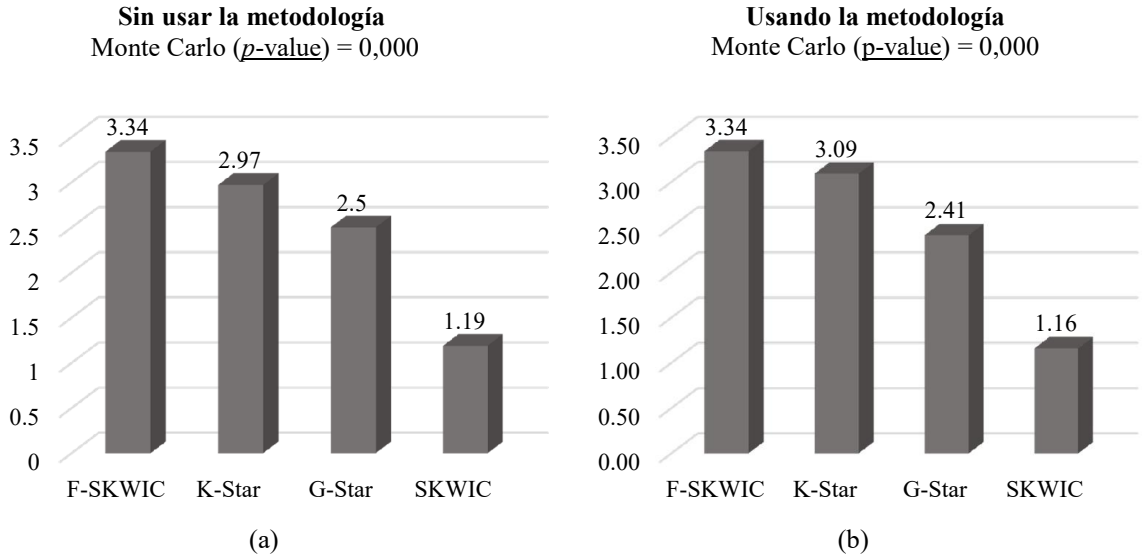
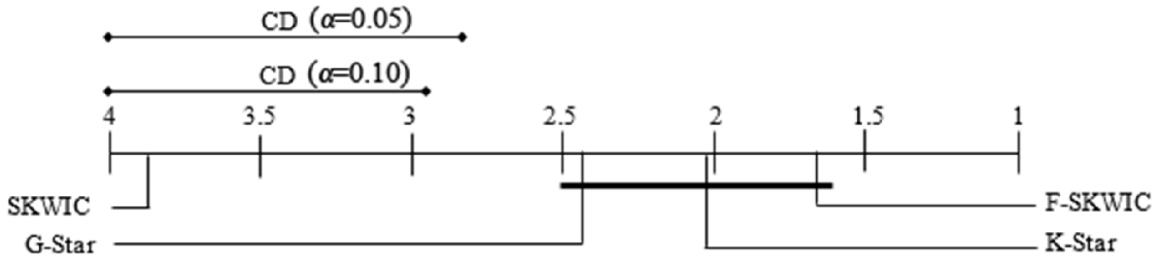


Figura 2.8 Resultados del test de Friedman. (a) Algoritmos sin usar la metodología, (b) Algoritmos usando la metodología.

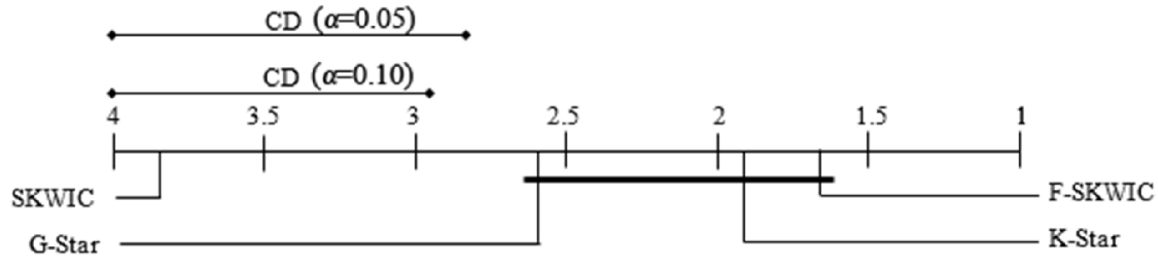
Con el test de *Nemenyi*, el desempeño de dos técnicas es significativamente diferente si el ranking promedio correspondiente difiere en al menos el valor de diferencia crítica (CD), donde los valores críticos q_α están basados en los rangos estadísticos estudentizados divididos por $\sqrt{2}$ [109]. Para 16 colecciones y cuatro algoritmos, usando un $\alpha = 0{,}05$ el valor de CD es 1,17, cuando $\alpha = 0{,}10$ el valor de CD es 1,04. Los resultados arrojados por el test de *Nemenyi* están representados visualmente en la Figura 2.9.

Las dos líneas que se encuentran en la parte superior de los diagramas corresponden a los valores de CD para α=0,05 y α=0,10 respectivamente. En la abscisa se graficó el ranking promedio de los métodos. Los valores de la recta se encuentran rotados, de forma que los rankings menores (los mejores, en el caso de *Nemenyi*) se encuentran a la derecha del diagrama, por tanto se puede decir que los métodos con mejor ranking se encuentran a la derecha [109].

Utilizando el sistema *Keel* [126], se calcularon los valores de significación arrojados por *Nemenyi* para $\alpha = 0.05$ y $\alpha = 0.10$. La Tabla 2.4 y Tabla 2.5 muestran los resultados obtenidos con el sistema.



(a) Algoritmos sin utilizar la metodología, resultado del análisis de los resultados de la Tabla 2.2.



(b) Algoritmos cuando utilizan la metodología, resultado del análisis de los resultados de la Tabla 2.3.

Figura 2.9 Visualización del test de *Nemenyi* utilizando $\alpha = 0.05$ y $\alpha = 0.10$. Algoritmos que no presentan diferencias significativas se encuentran conectados por la línea en negrita.

Tabla 2.4 Resultados arrojados por el test de *Nemenyi*, cuando no es utilizada la metodología

Parejas	(*p*-value)[a,b]
F-SKWIC vs. SKWIC	0,000001
F-SKWIC vs. K-Star	**0,411314**
F-SKWIC vs. G-Star	**0,086964**
K-Star vs. SKWIC	0,000054
K-Star vs. G-Star	**0,373439**
G-Star vs. SKWIC	0,001636

Tabla 2.5 Resultados arrojados por el test de *Nemenyi*, cuando se utiliza la metodología

Parejas	(*p*-value)[a,b]
F-SKWIC vs. SKWIC	0,000002
F-SKWIC vs. K-Star	**0,583882**
F-SKWIC vs. G-Star	**0,03998**
K-Star vs. SKWIC	0,000022
K-Star vs. G-Star	**0,132006**
G-Star vs. SKWIC	0,00617

[a] $\alpha = 0.05$, El test de *Nemenyi* rechaza la hipótesis nula a aquellos valores con un $p \leq 0.008333$
[b] $\alpha = 0.10$, El test de *Nemenyi* rechaza la hipótesis nula a aquellos valores con un $p \leq 0.016667$

Estos resultados confirman estadísticamente que al utilizar la metodología para el cálculo de la función *OverallSimSUX* los algoritmos presentan el mismo comportamiento que cuando esta no es utilizada. Por otra parte se puede afirmar que los mejores resultados fueron alcanzados por el algoritmo *F-SKWIC*, aunque este no presenta diferencias significativas con respecto a los algoritmos *K-Star* y *G-Star*, para los casos de estudios analizados.

El último experimento consiste en realizar una comparación por pares de los resultados obtenidos por el mismo algoritmo, utilizando la metodología y sin usar esta. La Figura 2.10

muestra estos resultados, una gráfica para los resultados de cada algoritmo. Como se puede apreciar, en los cuatro gráficos los resultados arrojados por los algoritmos son en su mayoría mejores cuando se utiliza la metodología.

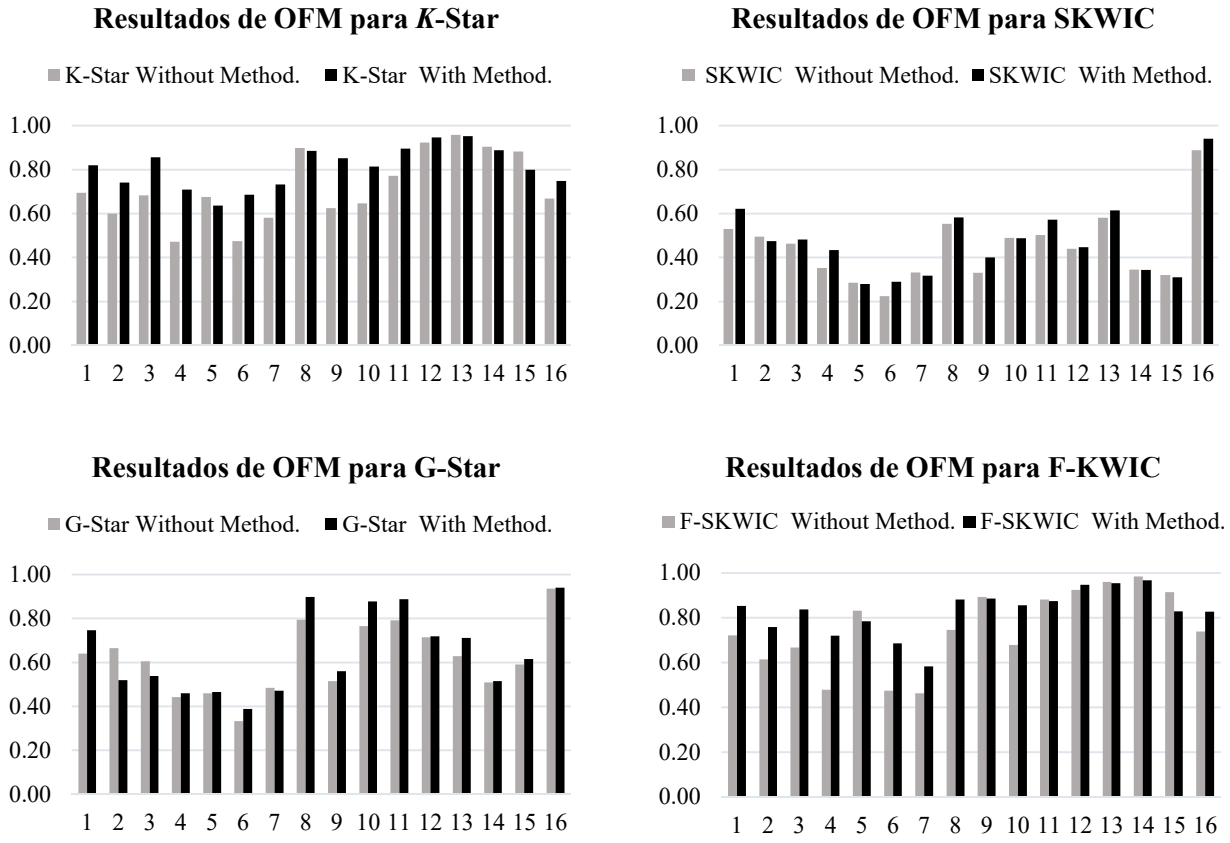


Figura 2.10. Comparación por pares, formados por los resultados del mismo algoritmo, usando la metodología y sin usarla.[33]

El test no paramétrico de *Wilcoxon* fue aplicado a los resultados del tercer experimento, para chequear si existen diferencias significativas entre las parejas de resultado (cuando es utilizada la metodología y cuando esta no es utilizada), de este modo se verifica estadísticamente si los resultados alcanzados por los algoritmos son comparables o mejores cuando se utiliza la metodología a cuando esta no es utilizada. La tabla Tabla **2.6** muestra los resultados del test de *Wilcoxon*.

[33] Tomado de Magdaleno, D., et al., "Comparative Study of Clustering Algorithms using OverallSimSUX Similarity Function for XML Documents"

Tabla 2.6 Resultado del test estadístico de *Wilcoxon* para las comparaciones en parejas, cuando la metodología es utilizada (*mG-Star*) y cuando no es utilizada (*G-Star*)

Parejas	Suma de ranking		(*p*-value)[a]
mF-SKWIC - FSKWIC	Neg.	23	0,009
	Pos.	113	
mK-Star - K-Star	Neg.	18	0,004
	Pos.	118	
mGStar - G-Star	Neg.	30	0,024
	Pos.	106	
mSKWIC - SKWIC	Neg.	25	0,011
	Pos.	111	

[a] Significación de Monte Carlo

Como se observa en la Tabla 2.6, el test de *Wilcoxon* sugiere rechazar la hipótesis nula (*p*-value < 0,05) para todas las parejas comparadas. Este resultado confirma estadísticamente que para los casos de estudio analizados, los cuatro algoritmos estudiados obtienen mejores resultados utilizando la metodología para el cálculo de la similitud *OverallSimSUX* que cuando no la utilizan.

2.8 Sistema para el agrupamiento de artículos científicos en formato XML usando Lucene (LucXML)

LucXML [23] es un sistema implementado en Java, que 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.

El sistema implementa el propuesto en la sección 2.2, que 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.

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 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. El sistema permite realizar el agrupamiento físico, permitiendo tener una carpeta por cada agrupamiento.

2.9 Conclusiones Parciales

La metodología propuesta para el agrupamiento de documentos científicos en formato XML facilita eficazmente el agrupamiento combinando las dimensiones estructura y contenido, con ello contribuye a la gestión de información y conocimiento.

La función *OverallSimSUX* captura de forma eficaz y eficiente el grado de similitud existente entre documentos XML, utilizando las unidades estructurales de los documentos.

De las tres variantes para el cálculo del umbral analizadas los mejores resultados se alcanzaron con la media de las similitudes.

El método de agrupamiento a emplear no influye en los resultados alcanzados con la metodología.

De los algoritmos de agrupamiento analizados, el *Fuzzy-SKWIC* presenta los mejores resultados, aunque no presenta diferencias significativas con respecto al *K-Star* y al *G-Star* acorde a las conclusiones arrojadas por el test de post-hoc de *Nemenyi*. Para cada algoritmo analizado y los casos de estudio seleccionados, cuando se utiliza la metodología, estos obtienen mejores resultados que cuando no se tiene en cuenta.

La complejidad computacional de la metodología es $O(mn^2)$, siendo m el número de documentos y n el número de rasgos existentes en la colección.

El sistema *XMLearning* facilita el procesamiento de colecciones de artículos científicos en formato XML mediante la indexación y agrupamiento, permitiendo además establecer comparaciones de varios resultados de agrupamientos, pues incorpora varias técnicas de agrupamiento. Por su parte *LucXML* es un sistema que permite la indexación y recuperación a través de consultas utilizando *Lucene*; el resultado de la recuperación de información se muestra organizado en grupos de documentos afines utilizando la metodología propuesta.

3

APLICACIONES DE LA METODOLOGÍA EN DIFERENTES CONTEXTOS

3. APLICACIONES DE LA METODOLOGÍA EN DIFERENTES CONTEXTOS

La cantidad de artículos científicos existentes en la Web en formato XML está creciendo exponencialmente, bases de datos de revistas internacionales almacenan su información en este formato (ejemplo: *DBLP*, *Elseiver*). Los artículos científicos presentan características particulares que los distinguen de otros documentos (palabras claves, referencias bibliográficas). Si estas características distintivas se usan en función de lograr un mejor agrupamiento de los artículos científicos pueden obtenerse resultados relevantes.

Por otra parte, a pesar del auge de los documentos XML, en un repositorio de información existen varios tipos de documentos que son no estructurados como los de formato: *doc*, *pdf*, y *txt*. La mayoría de estos documentos no presentan una estructura explícita que pueda ser explotada en un agrupamiento. Sin embargo es posible afirmar que estos documentos se encuentran estructurados implícitamente de acuerdo a la semántica de su contenido. Por lo que es posible considerar la posibilidad de realizar el agrupamiento de documentos no estructurados atendiendo no solo a su contenido sino también a la estructura que llevan implícita.

Por tal motivo en este capítulo, Partiendo de la metodología, se presentan varias aplicaciones: (1) trabajo con documentos científicos en formato XML, (2) una aplicación WEB que implementa la metodología incorporando documentos científicos en diferentes formatos y (3) una aplicación de la metodología en el área de la Salud, considerando las Historias Clínicas Electrónicas.

3.1 Aplicación de la metodología sobre documentos científicos en formato XML

Los artículos científicos por lo general presentan una estructura bien definida (*autor*, *título*, *resumen*, *palabras claves*, *contenido principal del artículo*, *notas* y *referencias bibliográficas*). En la Figura 3.1 se muestra una visión gráfica de la adaptación realizada a la

metodología realizando un análisis más profundo con las componentes existentes en las *Referencias Bibliográficas*. Nótese que cada elemento de esta unidad estructural es tratado de forma independiente, de este modo se pretende obtener información más precisa a la hora de comparar los documentos.

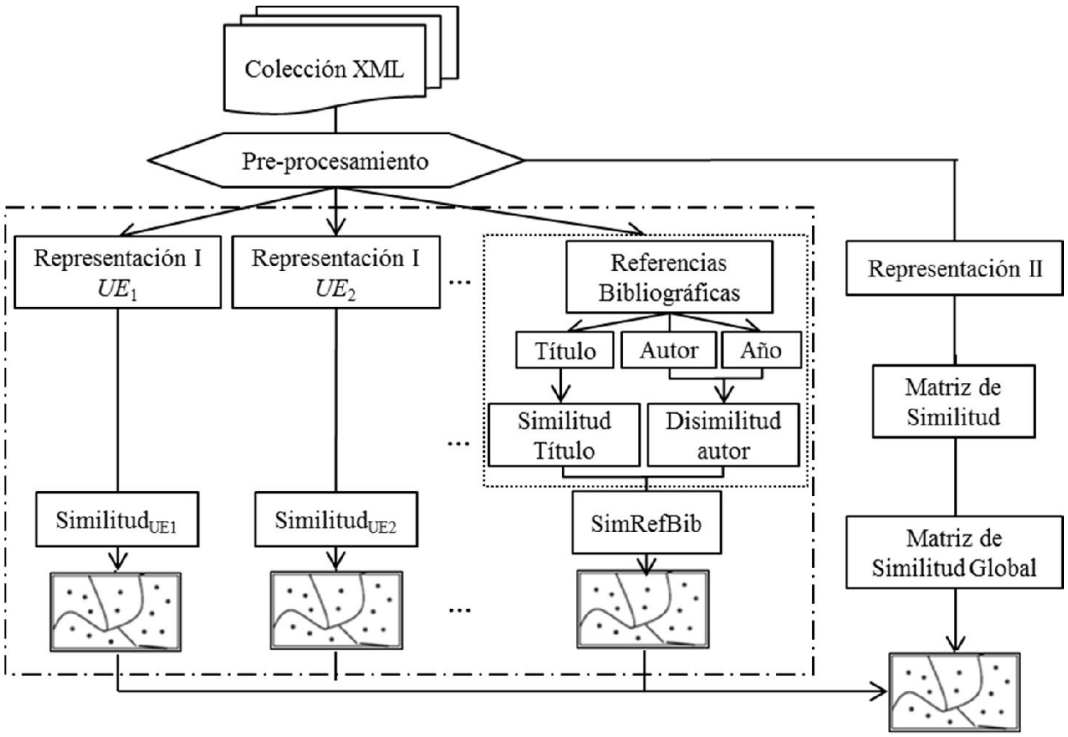


Figura 3.1 Esquema que representa la metodología analizando de forma más detallada las Referencias Bibliográficas

3.1.1 Representación de la información de las referencias bibliográficas

Se utilizó la representación VSM para la subunidad *título* y una modificación de la misma para la representación de la subunidad *autor*. Para realizar la representación de la subunidad *título* se realizan las transformaciones mencionadas en la sección 2.1.1. La transformación de la subunidad *autor* es diferente a la realizada para la subunidad *título*, ya que carece de sentido tratar cada palabra que compone el nombre de un autor como un término independiente, pues podría ocasionar ruido a la hora del análisis.

Ejemplo:

Supóngase que se tienen los documentos *di* y *dj*. En *di* se referencia al autor "Pedro García Pérez" y en *dj* al autor "Jorge Álvarez García", la palabra "García" como término independiente aportaría cierta similitud entre ambos documentos, sin embargo esta similitud

realmente no existe debido a que se están referenciando autores totalmente diferentes. Por tal motivo se decidió que el contenido de la subunidad autor será tratado como una cadena de caracteres, así se considerará el nombre completo de un autor como un solo término.

3.1.1.1 Extracción de términos de la subunidad título

Para obtener la representación de la subunidad título, se parte de una secuencia de tokens y se produce una secuencia de términos indexados basados en estos, seleccionando aquellos que se consideran relevantes. Se dice que una palabra es considerada candidata a relevante cuando su frecuencia de aparición supera el umbral *fap*; este umbral varía dependiendo de la cantidad de referencias bibliográficas ($CRB(i)$) existente en el documento analizado, así:

$$fap(i) = \begin{cases} 2 & si \quad CRB(i) \leq 10 \\ 3 & si \quad 10 < CRB(i) \leq 20 \\ 4 & si \quad 20 < CRB(i) \leq 25 \\ 5 & e.o.c. \end{cases} \qquad 3.1$$

En el caso de la subunidad *título* puede darse el caso de que al tratar los tokens como términos aislados se obtengan valores de similitud que no se correspondan con el grado de semejanza real que existe entre los documentos. Por tanto, después de realizar la selección se realiza un proceso de unión.

Ejemplo:

Supóngase que para un documento *di* se obtienen los tokens relevantes (fuzzy, set, logic, precision) y para un documento *dj* se obtienen (set, rough, generalization, decision); para este caso se tiene el término relevante set en ambos documentos, sin embargo estos documentos no guardan relación, *di* trata conjuntos difusos (fuzzy set) y *dj* trata conjuntos aproximados (rough set).

El proceso de unión de tokens, en su primera fase, consiste en buscar la frecuencia de aparición de los tokens relevantes (tomados dos a dos) en la subunidad título del documento. Luego de obtenidos los pares de tokens relevantes, los cuales serían aquellos que superen el umbral *fap*, se analiza si algunos de los pares formados pueden unirse, esto se hace solo para los pares que la subcadena inicial del primer par coincida con la subcadena final del segundo, o viceversa, tomando como subcadena inicial la primera palabra del par y como subcadena final la última palabra.

Al terminar el proceso de unión de tokens se obtienen las frases relevantes para el documento, así como la importancia de cada una de ellas. La Definición 3.1 denota la importancia de la palabra relevante *k* en el documento *i*.

Definición 3.1 (importancia de palabra relevante): Sea f_{ki} la frecuencia de aparición de la palabra relevante *k* en el documento *i* y *CRB*(*i*) la cantidad de referencias bibliográficas presentes en el documento *i*, se define la importancia de la palabra *k* en el documento *i* como:

$$Imp(k,i) = \begin{cases} f_{ki}/CRB & si\ f_{ki}/CRB \leq 1 \\ 1 & e.o.c. \end{cases} \quad 3.2$$

Posterior al proceso de unión aún quedan tokens aislados o independientes, estos no pasan directamente a formar parte de las palabras relevantes, solo se consideran relevantes aquellos que tienen una frecuencia de aparición mayor o igual que el 25% de la cantidad de referencias del documento, o que su frecuencia absoluta de aparición en las referencias bibliográficas es mayor que 10.

También puede ocurrir que algunos de los tokens que fueron unidos con otros superen como token independiente el umbral del 25% de las referencias o que su frecuencia de aparición sea mayor que 10, estos tokens también pasan a formar parte de las palabras relevantes.

Ejemplo:

Supóngase que se tienen los tokens (fuzzy, 20), (set, 8), (logic, 6) del documento *i* donde el segundo elemento del par indica la frecuencia de aparición del token; además se tiene que *CRB*(*i*)=24. Al aplicar la unión de tokens se obtiene que el par de tokens unidos fuzzy set presenta una frecuencia de aparición 7, por tanto el término fuzzy de manera independiente tiene una frecuencia de aparición 13. Como la frecuencia del término independiente fuzzy es mayor que el 25% (54% aproximadamente) el término pasa a formar parte también de las palabras relevantes, así se tienen como palabras relevantes del documento: fuzzy set, fuzzy, logic.

El proceso de unión de tokens no será aplicado en aquellos documentos en los que ningún término supere el umbral *fap*, ya que como términos independientes no logran superar el umbral, evidentemente no lo harán tampoco como términos unidos.

Palabras relevantes distintivas

Al obtener las palabras relevantes se puede apreciar que existen algunas que se repiten de manera significativa en la colección. Estas palabras serán denominadas palabras relevantes distintivas, ya que al estar presentes en una cantidad considerable de documentos, aportan un grado de similitud mayor y permiten determinar de una manera más precisa grupos de documentos afines. El proceso de selección de las palabras relevantes distintivas se describe en la Figura 3.2.

1. Calcular la cantidad de veces que aparece cada palabra relevante.
2. Ordenar las palabras descendentemente de acuerdo a la cantidad de veces que aparecen.
3. Si *k*=0 ir al paso 5, donde *k* es el número de palabras relevantes distintivas que el usuario decide seleccionar.
4. Seleccionar las *k* primeras palabras. Salir.
5. Seleccionar todas las palabras que su frecuencia de aparición en la colección sea mayor que 5. Salir.

Figura 3.2 Pasos para la selección de palabras claves distintivas

3.1.1.2 Extracción de términos de la subunidad autor

Para obtener la representación de la subunidad *autor* se utiliza una modificación de VSM, considerando el nombre completo del autor como un solo término; almacenando por cada autor: la cantidad de veces que es referenciado en el documento (importancia), el intervalo de años en los que este autor es referenciado en el documento y la cantidad de veces que no es referenciado como principal.

Las diversas variantes con las que un autor firma sus artículos dificulta el análisis de las citas recibidas, disminuyendo el impacto de su producción científica y su visibilidad [139]. Resulta imprescindible en el procesamiento de la subunidad *autor* normalizar el almacenamiento de los nombres de los autores.

Ejemplo: Supóngase que se tiene el autor Juan Pablo Pérez Rodríguez, el mismo puede aparecer referenciado de las siguientes formas (e incluso otras):

- J. P. Pérez
- J. Pérez
- Pérez J.
- Pérez J. P.
- Juan P. Pérez Rodríguez

Para resolver este problema se estandariza el nombre de autor de la siguiente forma: *XY Apellido*, donde *X*, *Y* representan las iniciales del nombre del autor y *Apellido* será el primer apellido del autor.

Es importante destacar algunos trabajos en torno a la desambiguación y normalización del nombre de los autores, como por ejemplo el realizado por [140] en el cual se hace un análisis crítico de las principales aproximaciones existentes en la literatura para solucionar el problema de la desambiguación de los autores en las publicaciones científicas. Sin embargo, la mayoría de los trabajos allí referenciados brindan soluciones que utilizan los metadatos de las revistas digitales o la web como fuente de información lo cual resultaría difícil adaptar a la propuesta.

Otro de los problemas que se pueden encontrar es el término *et.al.*, utilizado para referirse a un grupo de autores. Este término no aporta información relevante y puede causar ruido en el momento de establecer la similitud, por tanto no se tiene en cuenta en el análisis.

3.1.2 Cálculo de la similitud entre artículos científicos en formato XML

Las subunidades *título* y *autor* son tratadas de manera independiente en el cálculo de la similitud. La subunidad *año* se utiliza en función de la subunidad *autor*, pues por tener dos artículos años similares en las referencias no tienen porque realmente tratar de un mismo tema.

3.1.2.1 Cálculo de la Similitud Título

La similitud título entre un par de documentos i, j se define por el grado de semejanza que existe entre las palabras relevantes obtenidas de la subunidad *título* para cada documento, por tanto el primer paso en la obtención de esta similitud será calcular los valores de semejanza entre cada par de palabras.

Supóngase que se tiene el documento d_i con las palabras relevantes (PI_1, PI_2,…, PI_n) y el documento d_j con las palabras relevantes (PJ_1, PJ_2,…, PJ_m). Se forma la matriz *Sim_matrix*(nxm), donde n y m son la cantidad de palabras relevantes de *di* y *dj* respectivamente, Cada celda (i,j) de la matriz indica el valor de semejanza entre los términos i, j, este valor se calcula utilizando la distancia *JaroWinkler* [141], ver ecuación 3.3.

$$JW(i,j) = \begin{cases} d_{jaro} & si \quad d_{jaro} < b_t \\ d_{jaro} + \left(l \times p \times \left(1 - d_{jaro}\right)\right) & e.o.c. \end{cases} \quad 3.3$$

donde: l es la longitud de prefijo común de las cadenas hasta un máximo de cuatro caracteres; p es un factor de escala constante que no debe ser mayor que 0,25, de lo contrario la distancia puede dar valores mayores que 1 (0,1 es el valor estándar que se utiliza); b_t es el umbral que se define (se recomienda 0,7) [141]; d_{jaro} es la distancia de Jaro entre las cadenas i, j, la cual se define como:

$$d_{jaro} = \begin{cases} 0 & si \quad m = 0 \\ \frac{1}{3}\left(\frac{m}{|i|} + \frac{m}{|j|} + \frac{m-t}{|m|}\right) & e.o.c. \end{cases} \quad 3.4$$

donde n indica la cantidad de caracteres que igualan y t representa la mitad de la cantidad de trasposiciones de los caracteres que machean.

El cálculo de la *Similitud Título* se formaliza en el algoritmo que se muestra en la Figura 3.3.

De forma general este algoritmo busca en la matriz *Sim_matrix* el par de palabras (i, j) con máximo valor de similitud, si este valor supera el umbral establecido (para este caso 0,9) o ambas palabras contienen una misma palabra relevante distintiva, el valor de semejanza de estas palabras se considera en la función y además se multiplica por el promedio de los pesos de estas palabras, el peso de una palabra x en un documento i está dado por la importancia de esta palabra en el documento y es calculado mediante la expresión 3.2. Se buscarán como máximo q pares de palabras, siendo q el mínimo entre n y m. El valor general de semejanza obtenido será estandarizado con la suma de: la diferencia del máximo valor entre m y n con respecto a la cantidad de palabras que coinciden, y la sumatoria de los pesos máximos de cada par (i, j) que fue seleccionado.

3.1.2.2 Cálculo de la disimilitud autor

Para el cálculo de la disimilitud entre los documentos, teniendo en cuenta la subunidad *autor* se utilizó la medida *DisAut* la cual se precisa en la Definición 3.2.

Entrada: *DOC*1(2x*N*) y *DOC*2(2x*M*), *N* y *M* cantidad de palabras relevantes de *DOC*1 y *DOC*2 respectivamente, *DOC*1[*i*,1]: palabra relevante, *DOC*1[*i*,2]: importancia de la palabra. Vector *PRS* de longitud *k* donde *PRS*[i]=palabra relevante significativa.
Salida: *sim* = *sumaPesoN*/((max(*N*, *M*) − *PalabrasMachean* + *sumaPesoD*)), (Semejanza entre los documentos *di*, *dj*)
Inicio
1.Sim_matrix ← Calcular matrix similitud, según ecuación 3.3
2.**Para cada** *PRSi* **hacer**
si *DOC*1[s_1,1] **y** *DOC*2[s_2,1] contienen a *PRSi* **entonces**
ListaPRIC[*j*,1] ← s_1, *ListaPRIC*[j,2] ← s_2
Finalizar_Para
3.*minCPR* ← min(*N*,*M*)
4.**Repetir**
maximoV ← max(Sim_matrix[i,j])|((i ∉ *Ielegidas*) ∧ (j ∉ *Jelegidas*))
si máximoV ≥ 0.9 **o** *ListaPRIC* contiene (*i*,*j*) **entonces**
incrementar *PalabrasMachean*
incrementar *t*
Ielegidas ← *Ielegidas* ∪ *i*
Jelegidas ← *Jelegidas* ∪ *j*
sumaPesoN += $\sqrt{\text{maximoV}}\frac{(DOC[i,2]+DOC[j,2])}{2}$
sumaPesoD += *max*$(DOC[i,2], DOC[j,2])$
sino
Sim_matrix[i,j] ← 0
Hasta que (*t* ≥ *minCPR*) **o** (*maximoV*=0)
Fin

Figura 3.3 Cálculo de la similitud título

Definición 3.2 (DisAut): Dados los documentos *i*, *j* se define la medida de disimilitud *DisAut*(*i*,*j*), la cual indica que tan diferentes son este par teniendo en cuenta los autores que se referencian en los mismos; esta medida se formaliza matemáticamente en la ecuación 3.5.

$$DisAut(i,j) = DP(i,j)^{(1-SB(i,j))} \times DSP(i,j)^{SB(i,j)} \qquad 3.5$$

SB(*i*,*j*) indica la similitud binaria entre el par de documentos analizados, *DP*(*i*,*j*) es la disimilitud ponderada entre los mismos y *DSP*(*i*,*j*) es la disimilitud no ponderada.

La idea general de calcular la similitud binaria (*SB*(*i*,*j*)) parte de la siguiente hipótesis: si dos documentos referencian en un alto por ciento a los mismos autores estos documentos deben tratar temas similares y se aplicaría entonces la disimilitud sin peso.

La disimilitud no ponderada se refiere en este caso a no considerar el intervalo de años en que se referencia cada autor. Es probable que si en dos documentos *di* y *dj* un mismo autor

es referenciado pero la diferencia entre los intervalos de referencia de cada uno de los documentos es considerablemente grande, este autor pudiera tratar temas diferentes, o lo que es lo mismo, puede haber cambiado su línea de investigación, pero esta probabilidad se reduce en la medida en que aumenta la cantidad de autores iguales que se referencian en ambos artículos científicos, ya que sería muy casual que varios autores cambiaran al unísono su línea de investigación.

Definición 3.3 (Similitud binaria): Dados los documentos *i*, *j* se define la similitud binaria entre los mismos como:

$$SB(i,j) = \begin{cases} 1 & si\ SFT \geq \varepsilon \\ 0 & e.o.c. \end{cases} \quad 3.6$$

Siendo ε el umbral de similitud que se define (se recomienda ε=0.5). *STF*(*i*,*j*) se define a través de la ecuación 3.7.

$$STF(i,j) = \frac{2 \times \sum_{k=0}^{n} STFP(i_k, j_k)}{\sum_{k=0}^{n} NC(i_k, j_k)} \quad 3.7$$

En la ecuación anterior *n* indica la cantidad de autores referenciados en la colección. Los valores de *STFP*(*ik*,*jk*) y *NC*(*ik*,*jk*) se formalizan en las ecuaciones 3.8 y 3.9.

Cik y *Cjk* indican la cantidad de veces que es referenciado el autor *k* en los documentos *i*, *j* respectivamente, *Cnpik* y *Cnpjk* indican la cantidad de veces que el autor *k* es referenciado como no principal en los documentos mencionados.

$$STFP(i,j) = \begin{cases} 1 & si\ (Ci_k \neq 0) \wedge (Cj_k \neq 0) \\ 0 & e.o.c. \end{cases} \quad 3.8$$

$$SB(i,j) = \begin{cases} 1\ si\ \ (Ci_k - Cnp_{ik} \neq 0) \vee \left(Cj_k - Cnp_{jk} \neq 0\right) \vee (STFP(i,j) \neq 0) \\ 0 \qquad\qquad e.o.c. \end{cases} \quad 3.9$$

La disimilitud ponderada *DP* indica la diferencia que existe entre un par de documentos teniendo en cuenta: los autores que referencia, la cantidad de veces que lo hace y el intervalo de años en que es referenciado cada autor. La formalización matemática de *DP* aparece en la ecuación 3.10.

El peso está dado por el operador w_k el cual varía dependiendo del intervalo de años en que se referencia el autor k en los documentos i, j. De esta manera, si los intervalos de años se intersectan o son cercanos (se considera un par de intervalos cercanos como una vecindad de 5 años), se aplicaría el operador aritmético diferencia (-) a la cantidad de referencias del autor k. En caso contrario se aplica el operador aritmético adición (+) ya que a pesar de estar referenciando al mismo autor, los documentos no tienen por qué tratar temas similares debido a que la diferencia de años es considerable.

$$DP(i,j) = \frac{\sum_{k=0}^{n} \left| C_{ik}(w_k) C_{jk} \right| - \left(AR_i + AR_j \right)}{\left(CA_i - AR_i \right) + \left(CA_j - AR_j \right)} \qquad 3.10$$

Donde: n indica la cantidad de autores referenciados en la colección de documentos; Ci_k y Cj_k representan la cantidad de veces que es referenciado el autor k en los documentos i, j respectivamente; CAi y CAj representan la suma de las cantidades de referencias de todos los autores en i y en j respectivamente; ARi y ARj representan la suma de las veces que los autores en los documentos i, j proporcionan ruido.

Un autor k se considera que proporciona ruido al hallar la disimilitud entre un par de objetos i, j, (con $Ci_k \geq Cj_k$) si la cantidad de veces que se referencia como autor no principal en el documento i es mayor que cero y la diferencia entre las cantidades de referencias totales (Ci_k Cj_k) de este autor en el documento i y en el documento j es también mayor que cero, el valor de ruido ($VR(k,i)$) que aporta el autor se define como:

$$VR(k,i) = \begin{cases} Cnp_{ik} & si \left(C_{ik} - C_{jk} - Cnp_{ik} \right) \geq 0 \\ C_{ik} - C_{jk} & e.o.c. \end{cases} \qquad 3.11$$

Cnp_{ik} representa la cantidad de veces que el autor k es referenciado como autor no principal en el documento i.

La disimilitud no ponderada *DSP* solo varía con respecto a *DP* en que siempre se aplica el operador w_k como operador diferencia (-), por lo cual queda definida matemáticamente como ecuación 3.12:

$$DSP(i,j) = \frac{\sum_{k=0}^{n} \left| C_{ik} - C_{jk} \right| - \left(AR_i + AR_j \right)}{\left(CA_i - AR_i \right) + \left(CA_j - AR_j \right)} \qquad 3.12$$

3.1.2.3 Medida general de semejanza

Para el cálculo de la similitud general de la unidad estructural *referencias bibliográficas* de dos documentos, se hace necesario combinar de alguna manera el valor obtenido por la *Similitud Título* con el valor de la *Disimilitud Autor* de los mismos.

Un aspecto a tener en cuenta es que los valores obtenidos no tienen igual peso, debido a que se consideran más similares dos documentos que se parezcan más en cuanto a los títulos que referencian que en cuanto a los autores, pues no es posible relacionar de una manera unívoca un autor con un tema dado. Por tanto, se utiliza como valor de mayor peso para medir la similitud el obtenido por la *Similitud Título*, tratando el valor obtenido por la *Disimilitud Autor* como una influencia positiva. Finalmente se formaliza la similitud general entre dos documentos *i*, *j* considerando las *Referencias Bibliográficas* a través de la función (*SimRefBib*(*i,j*)), como se puede observar en la ecuación 3.13:

$$SimRefBib(i,j) = \begin{cases} ST(i,j)^{\,DistAut(i,j)} & si \quad ST(i,j) > 0 \\ 1 - DistAut(i,j) & si \quad ST(i,j) = 0 \end{cases} \qquad 3.13$$

3.1.3 Algoritmo de agrupamiento basado en Referencias Bibliográficas

En la Figura 3.4 se muestra un algoritmo de agrupamiento (*SemClustDML*) teniendo en cuenta la UE *Referencias Bibliográficas*, utilizando la matriz de similitud *SimRefBib*.

Los pasos de estimación del umbral de similitud y asignación de elementos a los grupos coinciden con los descritos en el Capítulo 2, cuando se enunció la metodología. En el caso específico del umbral de similitud se tiene en cuenta la función *SimRefBib*. A continuación se describen los pasos que son novedosos o sufren algún cambio.

3.1.3.1 Búsqueda de los centroides iniciales

Los centroides iniciales van a ser aquellos elementos a partir de los cuales se van a formar los grupos preliminares. Dado que dos elementos que tengan similitud menor que $\gamma/2$, siendo γ el umbral de similitud que se define, difícilmente pertenecerán a un mismo grupo. Por tanto,

se convierte en la búsqueda de un grupo de elementos que tengan similitud menor que $\gamma/2$ tomados dos a dos. Si no se encuentra al menos un par de elementos cuya similitud sea menor que $\gamma/2$ el algoritmo devolverá un solo grupo formado por el conjunto de documentos de la colección.

Entrada: Matriz de similitud *SimRefBib*, Conjunto de *n* documentos ($D = \{d_1, d_2, \ldots, d_n\}$), longitud mínima de cada grupo l, *v* cantidad de elementos aleatorios a seleccionar para comprobar si los grupos son agrupables.
Salida: *Lista de* grupos *formados* (*C*)
Inicio
1. Estimación del umbral de similitud.
2. Búsqueda de los centroides iniciales: C=$\{d_1, d_2, \ldots, d_n\}$, $k \leq n$, *donde cada* d_i *se considera un nuevo grupo c*.
3. Asignación de cada documento $d_i \notin C$ al c_j correspondiente.
4. **si** $cup \leftarrow \bigcup_{g,h=1}^{k} cap(c_g, c_h) = \emptyset$ donde $cap = c_g \cap c_h$, **entonces** ir al paso 6.
5. Determinar para cada $d_i \in cap(c_g, c_h) = \emptyset$ el cj correspondiente, donde $sim \leftarrow \frac{\sum_{r=1}^{m} SimRefBib(d_i, c_{jr})}{m}$ es máxima, *m* cantidad de elementos en c_j. Ir paso 4.
6. $\forall d_i \notin C, c_j \leftarrow (c_j \cup d_i)$, donde $sim \leftarrow \frac{\sum_{r=1}^{m} SimRefBib(d_i, c_{jr})}{m}$ es máxima.
Fin

Figura 3.4 Algoritmo de agrupamiento *SemClustDML*

3.1.3.2 Grupos solapados

Definición 3.4 (Grupos solapados): Dos grupos c_i, c_j se dicen son solapados si $c_i \cap c_j \neq \emptyset$.

Que un elemento supere el umbral de similitud al compararlo con más de un centroide no es común dadas las características de la matriz, pero ante la ocurrencia se hace necesario calcular para cada par de grupos solapados la pertenencia de los elementos que se encuentran en la intersección a cada uno de estos grupos. La α-pertenencia de un elemento *i* a un grupo c_j se define mediante la ecuación 3.14.

$$\alpha(i, C_j) = \frac{1}{n} \sum_{k=1}^{n} SimRefBib(i, C_{jk}) \qquad 3.14$$

En la ecuación anterior *n* indica la cantidad de elementos del grupo c_j.

3.1.3.3 Elementos sobrantes

Al calcular los centroides iniciales y asignar cada uno de los elementos restantes a estos centroides se tendrán algunos elementos que no superen el umbral con ninguno de los centroides, por lo cual no serán unidos a ningún grupo, estos son los llamados elementos sobrantes.

Una vez formados los grupos se calcula la α-pertenencia de cada uno de estos elementos a cada uno de los grupos, el elemento será añadido al grupo para el cual se obtenga el mayor valor de pertenencia.

3.1.4 Evaluación de los resultados del método de agrupamiento basado en las Referencias Bibliográficas

Para verificar la validez de los resultados obtenidos a través del método de agrupamiento y la función de similitud descritos, se diseñaron dos experimentos, aplicados a tres casos de estudio (Los dos primeros casos de estudio coinciden con los descritos en la sección 2.6.3, el tercer caso de estudio es formado por la combinación de documentos pertenecientes a los dos primeros); con el propósito de realizar un análisis estadístico, que permita comprobar si existen diferencias significativas entre el método de agrupamiento propuesto y otras variantes de algoritmos reportados en la literatura. La evaluación incluye la verificación y validación.

Para ser consecuente con la evaluación realizada a la metodología, se tuvieron en cuenta las mismas medidas en la presente evaluación: *Overall F-measure*, *Macro-Purity* y *Micro-Purity*.

3.1.5 Diseño de los experimentos

El primer experimento consistió en verificar si existen diferencias al aplicar la función de similitud propuesta a los tres casos de estudios mencionados anteriormente con respecto a otras funciones existentes en la literatura. Fueron seleccionadas las funciones *Jaccard*, *Coseno* y *Dice*.

En las Tablas A7.1 y A7.2 se muestran los resultados obtenidos por las medidas *Micro-Purity* y *Macro-Purity*, respectivamente. Al aplicar el test de *Friedman* se pueden observar resultados como los de la Figura 3.5, el valor de significación (p=0,001) indica que

existen diferencias significativas entre las poblaciones comparadas, teniendo en cuenta los resultados de los dos medidas.

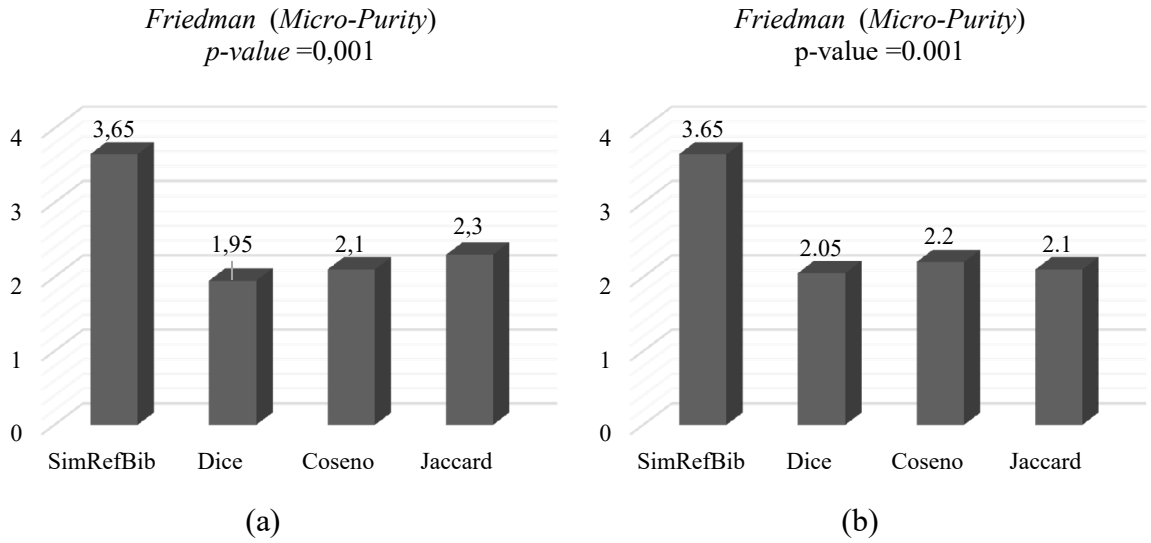


Figura 3.5 Valores de rango obtenidos al aplicar la prueba no paramétrica de *Friedman* a los valores obtenidos para las medidas (a) Micro-Purity y (b) Macro-Purity.

Por tanto se aplicó la prueba de *Nemenyi* a los resultados de las dos medidas. La Figura 3.6 muestra los resultados de este test. Se puede observar que existen diferencias significativas entre *SimRefBib* con respecto al resto de las funciones, para q=0,10. Estas diferencias significativas siempre arrojaron mejores resultados para la función *SimRefBib* que para las restantes.

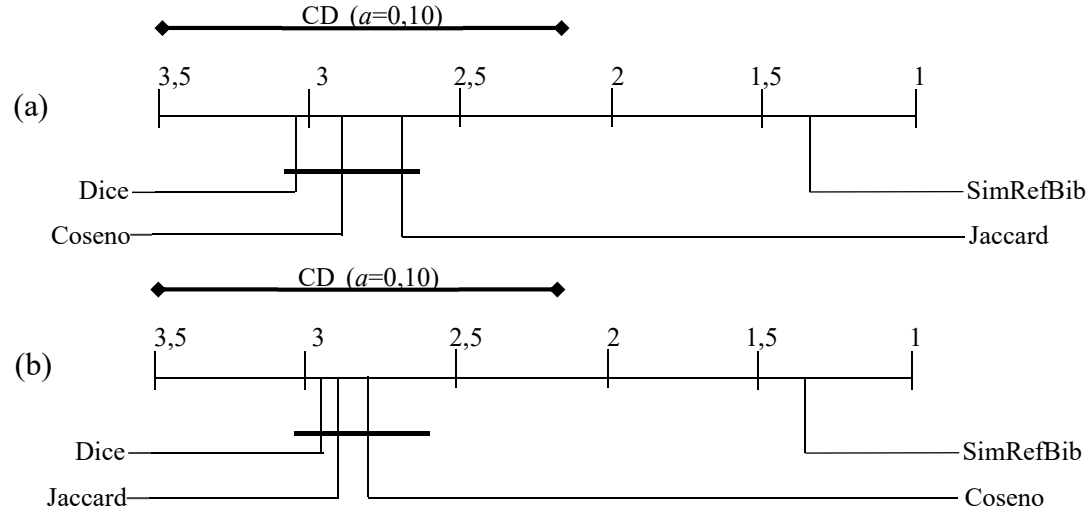


Figura 3.6 Test de *Nemenyi* para los valores de (a) *Micro-Purity* y (b) *Macro-Purity* obtenidos al aplicar el algoritmo *K-Star* a cada una de las funciones.

La prueba no paramétrica de *Friedman* arrojó que no existen diferencias significativas entre los valores OFM obtenidos para cada una de las funciones analizadas, ver Figura 3.7. Sin embargo se llegó a la conclusión que la función propuesta en esta investigación tiene un comportamiento más estable que el resto de las funciones analizadas como se muestra en la Figura 3.8.

N			10
Chi-Square			1.962
df			3
Asymp. Sig			0.58
Monte Carlo Sig	Sig		0.591
	95% Confidence Interval	Lower Bound	0.581
		Upper Bound	0.601

Figura 3.7 Resultados de la prueba estadística de *Friedman* con los valores de la medida OFM (Tabla A7.3)

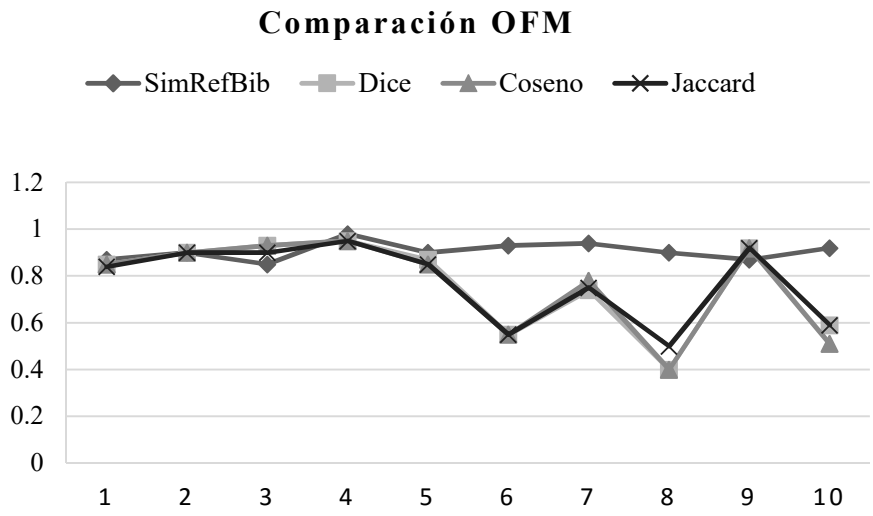


Figura 3.8 Comportamiento de la medida OFM para cada función una de las funciones analizadas

El segundo experimento realizado consistió en demostrar que los resultado del agrupamiento para el algoritmo *SemClustDML* propuesto en esta investigación mejora los resultados obtenidos por el algoritmo *K-Star* de INEX.

Para la comparación entre los algoritmos antes mencionados se utilizó la prueba no paramétrica de *Wilcoxon*. En la Tabla 3.1 se muestra el resultado de esta prueba aplicado a la medida OFM. Se demuestra que existen diferencias significativas entre las poblaciones

comparadas. La Figura 3.9 muestra el comportamiento de la medida OFM al aplicar el algoritmo *SemClustDML* y el algoritmo *K-Star* de INEX.

Tabla 3.1 Valores de significación de la prueba estadística de Wilcoxon con los valores de la Tabla A8.1

Parejas	**Suma de ranking**		**(*p*-value)[a]**
SemClustDML - K-Star	Neg.	5	0.021
	Pos.	50	

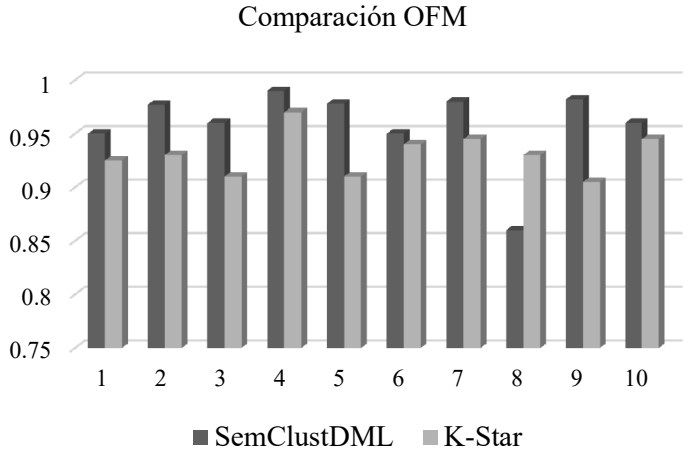


Figura 3.9 Comparación de los resultados obtenidos para la medida OFM al aplicar los algoritmos *SemClustDML* y *K-Star*.

3.2 Aplicación WEB que implementa la metodología incorporando documentos científicos en diferentes formatos

En esta sección se describe el proceso de implementación del sistema de recuperación de información que implementa la metodología *ScientificSolr*. El sistema permite recuperar los grupos de documentos más relevantes a una consulta realizada por el usuario, a partir de colecciones de documentos semiestructurados (XML) y no estructurados (PDF, DOC, TXT).

La metodología propuesta utiliza las UE de los documentos, bien delimitadas en los XML, para realizar el agrupamiento. Los documentos que se encuentran en un formato diferente a este no presentan una estructura explícita, por lo que para lograr una generalización del sistema se siguió la idea de segmentar el contenido de los documentos por tópicos, permitiendo realizar el agrupamiento bajo el supuesto de que estos constituyen las UE que componen el documento. En este variante del sistema *ScientificSolR* se seleccionó el método

TextLec propuesto en [142] como una posible solución. A continuación se mencionan las modificaciones necesarias para realizar esta adaptación.

3.2.1 Tópicos como unidades estructurales

La idea de considerar los tópicos como UE no solo requiere la implementación de las transformaciones mencionadas en la sección 2.1.1, sino que además plantea la necesidad de dar solución a una nueva problemática, que surge a la hora de realizar la *Representación I* de la metodología pues en esta se representa cada UE por separado, logrando analizar cada parte de un documento con sus homólogas de los restantes documentos. Por ejemplo, en el caso de una colección de artículos científicos de tipo XML, los resúmenes se agruparían entre sí, igualmente, las palabras claves, las referencias y así sucesivamente.

Desafortunadamente con los tópicos detectados en los documentos no estructurados, no se puede proceder del mismo modo, debido que estos no poseen un homólogo definido en los restantes documentos de la colección; por lo que se hace necesario realizar una adaptación a la *Representación I* para este tipo de documentos.

3.2.1.1 Modificación en la Representación I para el trabajo con los tópicos

Para realizar la *Representación I*, se evitó tener en cuenta el orden de aparición de los tópicos pues se tiene el inconveniente de que dos documentos pueden ser similares en cuanto a los tópicos que contienen, aunque estos no se desarrollen en el mismo orden y si se sigue este criterio no se tratarían los tópicos "homólogos".

Por tal motivo se propone considerar la similitud *coseno* entre los tópicos, en lugar del orden en que aparecen. De acuerdo a esta idea la similitud entre dos tópicos queda formalizada en la ecuación 3.15, donde w_{tr} es el peso del término *t* en el tópico *r* y se calcula dividiendo la frecuencia del término *t* en el tópico *r* entre el total de términos del tópico.

$$SimlTop(Top_i, Top_j) = \frac{\sum_{t=1}^{n} w_{ti} * w_{tj}}{\sqrt{\sum_{t=1}^{n} {w_{ti}}^2} * \sqrt{\sum_{t=1}^{n} {w_{tj}}^2}} \qquad w_{tr=\frac{f_{tr}}{T}} \qquad 3.15$$

En la Figura 3.10, se ilustra esta variante a través de un ejemplo tomando tres documentos. Supóngase que los tópicos que poseen mayor similitud están enlazados por las flechas. El

resto de los pasos de la metodología se mantiene inalterable, pues no es necesario realizar otras adaptaciones.

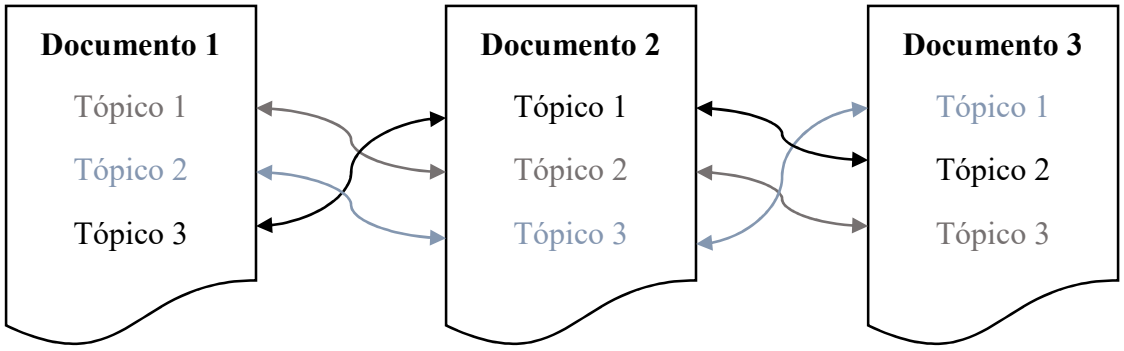


Figura 3.10 Comparación entre los tópicos

3.2.1.2 Método TextLec

Dada la necesidad de seleccionar un método para aplicar la metodología propuesta a documentos no estructurados, en esta sección se presenta el método *TextLec* propuesto en [142] como una posible solución. *TextLec* es un método de segmentación lineal, o sea está enfocado a textos con estructura lineal como es el caso de los artículos científicos. En los artículos científicos el vocabulario para desarrollar un tópico, pertenece a cualquier rama de la ciencia y la tecnología, por tanto, las palabras más relevantes, en relación a dicho tópico, usualmente se repinten con más frecuencia que el resto. Debido a esto el método considera la repetición de términos como un elemento confiable para identificar aquellas unidades textuales que están relacionadas con un mismo subtópico [142].

Otra característica importante de este método es que se basa en la similitud entre los párrafos para determinar los límites de los segmentos. *TextLec* toma como supuesto que el párrafo es la unidad básica del texto. Según algunos autores el párrafo se considera como una unidad con coherencia interna, por una parte, y con una conexión adecuada con el contexto lingüístico (que sigue y precede), por otra [143].

3.2.1.3 Un algoritmo de segmentación basado en TextLec

Dado que una condición necesaria para realizar la *Representación I* en la extensión que se propone, es detectar una misma cantidad de tópicos para todos los documentos, se hace necesario modificar el método seleccionado. *TextLec* posee esta única inconveniente para ser

aplicado en su forma original a la extensión que se propone, por tal motivo se realiza una variante del método donde la cantidad de tópicos a detectar constituye un parámetro del algoritmo. La variante consiste en adicionar un chequeo al final del algoritmo, si la cantidad de tópicos no coincide con la deseada se realizan las acciones que se describen a continuación, en dependencia de los dos casos que se pueden presentar: (1) que la cantidad sea mayor o (2) que la cantidad sea menor.

En caso de que la cantidad de tópicos obtenida sea mayor que la cantidad deseada, se aplica la misma idea que usa el algoritmo original para eliminar los segmentos espurios, o sea, se eliminan los segmentos de menor longitud uniéndolos a su vecino más similar como se muestra en el diagrama de la Figura 3.11. Nótese que la longitud de un segmento no es más que la cantidad de párrafos que lo componen.

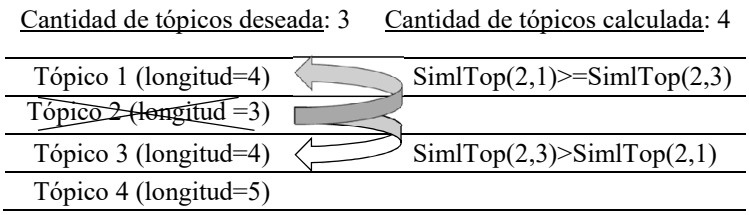


Figura 3.11 Acción para unir un tópico con su semejante.

El otro caso se produce cuando la cantidad obtenida es menor que la deseada, la solución que se propone es dividir el mayor tópico a la mitad obteniendo dos nuevos subtópicos, repitiendo este proceso hasta obtener la cantidad deseada. En la Figura 3.12 se muestra un ejemplo correspondiente a este caso.

Esta solución quizás no sea la más eficaz en cuanto a la unicidad de los tópicos que se obtienen, ya que se puede interrumpir un tópico al dividirlo, resulta una solución eficiente por su poca complejidad temporal.

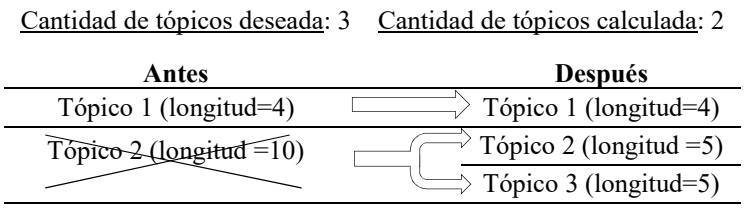


Figura 3.12 Acción para separar un tópico en dos

Es posible considerar otros criterios que tengan en cuenta las similitudes internas de los párrafos que contienen los segmentos, con el objetivo de poder decidir cuál será el segmento a particionar (no tienen que ser necesariamente el mayor) y el punto donde se particionará (no tiene que ser necesariamente el punto medio entre los párrafos que lo contienen). No obstante en el presente trabajo se adoptó la variante expuesta dando mayor importancia a la poca complejidad temporal que posee, lo cual es provechoso para un proceso tan extenso como el que se plantea para la extensión de la metodología.

De esta manera el nuevo algoritmo basado en *TextLec* queda constituido como expresa el pseudocódigo que se muestra en la Figura 3.13.

Dado que fijar la cantidad de tópicos a obtener influye en la calidad de los límites de tópicos que se obtienen como salida y posteriormente en la calidad del agrupamiento, se hace necesario establecer un criterio para estimar el mismo. Este tema queda abierto para futuras investigaciones, en el presente trabajo se asume que los documentos tienen entre 2 y 5 tópicos parciales o subtópicos.

3.2.2 Diseño del sistema ScientificSolr

El sistema permite recuperar de forma agrupada los documentos más relevantes a una consulta realizada por el usuario, a partir de colecciones de documentos semiestructurados (XML) y no estructurados (PDF, DOC, TXT). El diseño del mismo se basa en una arquitectura cliente-servidor y por tanto responde a dos tipos de usuarios, un administrador que debe encargarse de la indexación de las colecciones y un usuario que realiza las consultas. Ambas interfaces fueron diseñadas utilizando el marco de trabajo GWT, creado por Google bajo Licencia Apache v2.0 que permite crear aplicaciones AJAX en el lenguaje de programación Java.

3.2.2.1 Herramientas de RI

En el Anexo 99 se muestra el diagrama de componentes del sistema, donde se puede observar que se utiliza la biblioteca *jdom* para acceder a los documentos XML y también incorpora las herramientas de recuperación de información: *Lucene*, *Tika*, *SolR* para facilitar tanto el trabajo con los documentos no estructurados como con los índices.

```
Entrada: TxP: Matriz de términos por párrafos, N: Total de párrafos,
         CantTop: Cantidad de tópicos deseada
Salida: Lim: Vector con los límites de los segmentos
Inicio
1.Determinar Parf
2.Determinar los posibles límites de segmentos
3.Remover los segmentos de longitud menor a long_Ventana
4.Mientras k>CantTop hacer inicio /*Cantidad obtenida > CantTop */
   TopMin=Lim0-0
   Indice=0
   Para i=1 hasta k-1 hacer
    Si Lim(i+1)-Limi<TopMin entonces inicio
      TopMin= Lim(i+1)-Limi
      Indice=i+1
    Fin
   k=k-1
   Si simlTop(Indice-1, Indice)>simlTop(Indice, Indice+1) entonces
    Para j=Indice-1 hasta k hacer
       Limj=Lim(j+1)
   Sino Para j=Indice hasta k hacer
          Limj=Lim(j+1)
  Mientras k<CantTop hacer inicio/*Cantidad obtenida < CantTop */
   TopMax=Lim0-0
   Indice=0
   Para i= 1 hasta k-1 hacer
    Si Lim(i+1)-Limi>TopMax entonces inicio
       TopMax=Lim(i+1)-Limi
       Indice=i+1
    Fin
   Temp=LimIndice
   LimIndice= LimIndice-1+TopMax/2
   k=k+1
   Para j =indice+1 hasta k-1 hacer inicio
    Temp2=Limj
    Limj=Temp
    Temp=Temp2
   Fin
   Limk=Temp
Fin
Fin
```

Figura 3.13 Algoritmo de TextLec modificado para obtener la cantidad de tópicos deseada

3.2.2.2 GWT

El sistema cuenta con una interfaz para el administrador y otra para el cliente. Ambas interfaces fueron diseñadas utilizando el marco de trabajo GWT, creado por Google bajo Licencia Apache v2.0 que permite crear aplicaciones AJAX en el lenguaje de programación

Java. Estas aplicaciones son compiladas posteriormente en código JavaScript ejecutable optimizado que funciona automáticamente en los principales navegadores.

Este marco de trabajo posee estilos de organización muy convenientes para el desarrollo ya que utiliza solamente un lenguaje de programación.

Principales características de GWT:

- Componentes gráficos dinámicos y reusables: los programadores pueden usar clases prediseñadas para implementar.
- Soporte para depurado de Java.
- Internacionalización.
- Soporte para las *API's* de Google (inicialmente, soporte para Google Gears).
- Es de código abierto.
- Los desarrolladores pueden diseñar y desarrollar sus aplicaciones orientadas a objetos.
- Existen un numeroso conjunto de bibliotecas desarrolladas por Google y terceros que amplían las funcionalidades de GWT. [144]

3.2.3 Evaluación de los resultados del agrupamiento para documentos no estructurados

La evaluación persigue tres objetivos fundamentales: verificar que el sistema funciona correctamente y satisface las necesidades de los usuarios, encontrar si existen diferencias significativas entre el agrupamiento de los documentos no estructurados utilizando la metodología y otras existentes, así como determinar si la cantidad de tópicos influye en la calidad del agrupamiento correspondiente a la nueva variante. Con este propósito, primeramente, fue necesario definir los casos de estudio para la aplicación del agrupamiento de documentos no estructurados basado en la segmentación por tópicos, luego se exponen los estadígrafos seleccionados para evaluar la calidad del agrupamiento y finalmente se describen los experimentos que se diseñaron y se analizan los resultados obtenidos.

3.2.3.1 Casos de estudio para el agrupamiento de documentos no estructurados

En el presente trabajo los casos de estudio constituyen colecciones de artículos científicos en formatos no estructurados tomados del ICT que es un repositorio perteneciente al CEI en la UCLV. Dicho repositorio es visitado con frecuencia por los investigadores, profesores y estudiantes del centro ya que contiene un gran número de artículos científicos y documentos relacionados con diversos temas de investigación, disponibles para la red del Ministerio de Educación Superior. En el mismo la información se encuentra organizada y clasificada en diversos temas, lo que permite crear corpus de documentos de diferentes temas para aplicar agrupamientos sobre estos.

3.2.3.2 Diseño de los experimentos

Con el objetivo de evaluar la calidad de la extensión realizada se diseñaron dos experimentos. El primero está encaminado a determinar si la cantidad de tópicos detectada en los documentos no estructurados es un parámetro que influye significativamente en los resultados del agrupamiento. El segundo se enfoca en encontrar si existen diferencias significativas entre los resultados del agrupamiento de la mejor variante para cada corpus, basada en la segmentación por tópicos y la variante clásica que utiliza solamente la representación global.

Ambos experimentos se desarrollan sobre 16 corpus de documentos con formato PDF. A continuación se analizan los resultados de cada experimento. El sistema tiene incorporado como técnica de agrupamiento el algoritmo *K-Star*, con el cual fue realizada la evaluación.

Experimento 1

Este experimento consiste en comparar los resultados de las medidas seleccionadas para las variantes que detectan de 2 a 5 tópicos y se incluye también la variante global. Con el objetivo de determinar si existen diferencias significativas entre ellas se utilizó la prueba estadística de *Friedman*. A continuación se muestran los resultados obtenidos para cada una de las medidas de evaluación. Se emplea como notación *K_Star*(*Top*) para distinguir cada variante del método que usa tópicos, donde *Top* es la cantidad de tópicos detectados.

Como se puede apreciar en las Tablas A10.1-A10.3 la mayoría de las variantes que usan la segmentación por tópicos superan a la variante que utiliza la representación global para este

conjunto de documentos, según el valor de la medida de evaluación OFM; aunque el test de *Friedman* (ver Figura 3.14) muestra que no existen diferencias significativas entre ellas pues la significación es mayor que 0,05 (p=0,389). Los resultados tanto de *Micro-Purity* como de *Macro-Purity* muestran que del método que detecta tópicos solo la variante de donde se emplearon cinco tópicos es inferior al método global. Sin embargo las diferencias no son significativas.

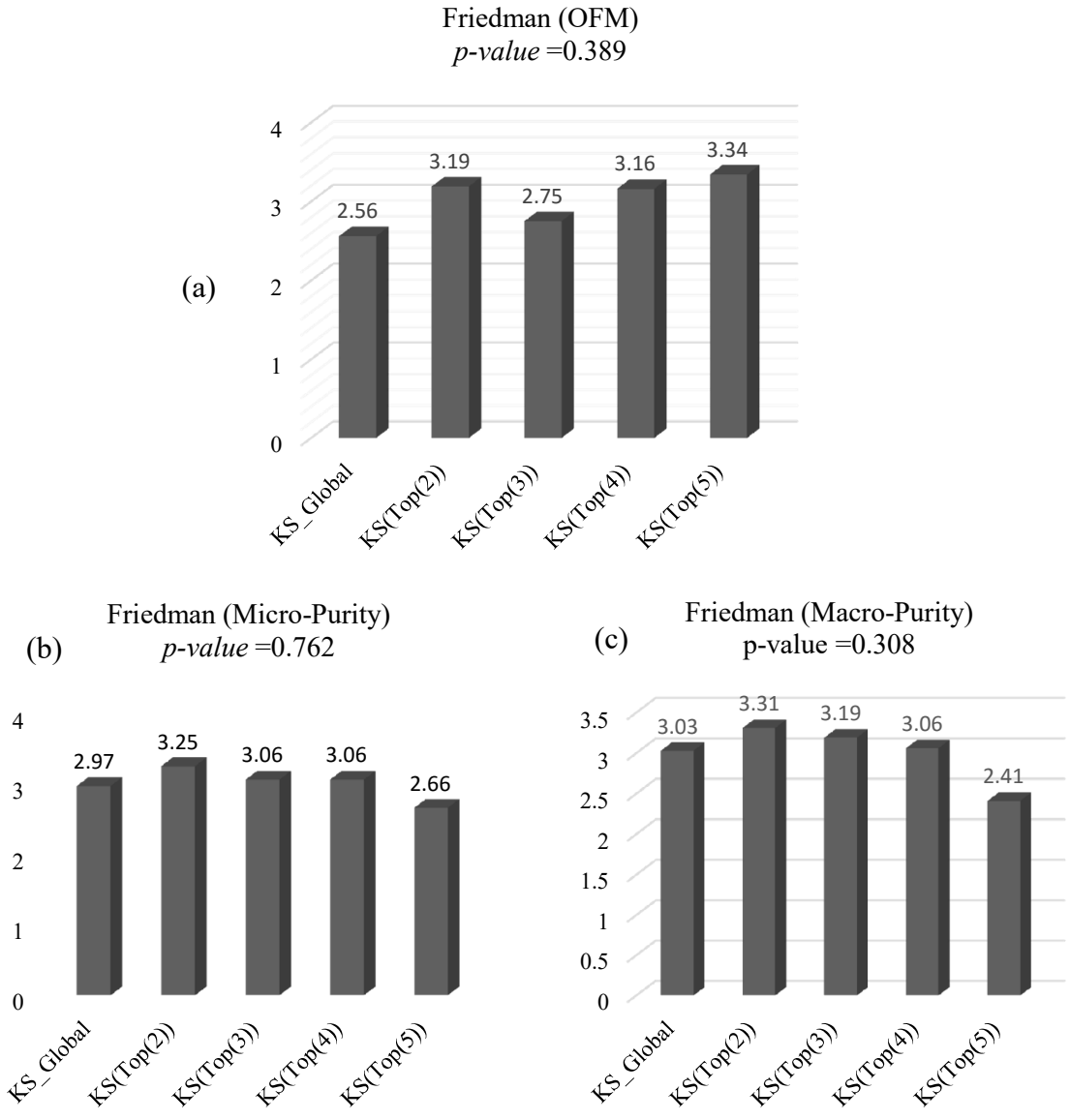


Figura 3.14 Resultados de la prueba de Friedman para las medidas (a) OFM, (b) *Micro-Purity*, (c) *Macro-Purity*

Experimento 2

El segundo experimento se dirige a encontrar si existen diferencias significativas entre la variante global y la variante con tópicos que mejor se comporte para cada corpus. Ahora por ejemplo se compara para un corpus C_i el valor de OFM de la variante global con el máximo de los valores de OFM de las variantes que utilizan la segmentación por tópicos como expresa la Tabla A11.1. De igual manera se procede en la comparación de los resultados de las medidas internas *Micro-Purity* y *Macro-Purity*. La **Tabla 3.2** muestra los resultados de aplicar la prueba de *Wilcoxon* a estos valores.

Tabla 3.2 Valores de significación de la prueba estadística de *Wilcoxon* con los valores de OFM, *Micro-Purity* y *Macro-Purity* obtenidos de la Tabla A11.1

Parejas	Suma de ranking		(*p*-value)[a]
OFM_UE_MAX - OFM_KSTAR	Neg.	0	0,002
	Pos.	55	
MicroP_UE_MAX - MicroP_KSTAR	Neg.	9	0,031
	Pos.	57	
MacroP_UE_MAX - MacroP_KSTAR	Neg.	13	1,63
	Pos.	42	

Los resultados del test de *Wilcoxon* no solo reafirman la conclusión de que las variantes que aplican la segmentación por tópicos se comportan mejor para este conjunto de documentos, sino que expresan además la existencia de diferencias significativas entre ambos métodos.

3.3 Aplicación de la metodología para Historias Clínicas Electrónicas

Administrar el flujo de documentos de una organización es esencial para asegurar el uso productivo de la información. Si bien los medios tecnológicos actuales, especialmente los informáticos, posibilitan la obtención y el almacenamiento de grandes cantidades de información, la llamada Sociedad de la Información está siendo superada por la necesidad de nuevos métodos capaces de procesar la información de forma eficiente y eficaz. Esto se hace lógicamente extensible a los centros hospitalarios, a partir del uso extendido de las historias clínicas en formato electrónico (HCE). Aunque varios sistemas han sido desarrollados con el fin de lograr una forma rápida y eficiente para compartir información, la heterogeneidad de

la misma determina que el extracto de conocimiento relevante se convierta en un proceso complejo y desafiante [145].

Disponer de información sistematizada, gestionarla de forma efectiva y segura es esencial para garantizar mejores prácticas en salud.

La importancia de la normalización y la codificación de los datos almacenados en una HCE es reconocida por varios investigadores [145, 146]. Como resultado, la recogida de información clínica debe migrar a la utilización controlada de textos estructurados. De hecho, la distribución de sus elementos permite concebir la HCE como un documento XML, debido a la estructura jerárquica y auto-descriptivo de la información implícita en cada uno de los factores que la componen. Health Level Seven (HL7) [145] es un conjunto de normas de salud con amplia cobertura internacional para dar soporte a las HCE. El formato HL7 permite compartir datos eletrónicos de información clínica a través de un modelo de modelado formal (UML) y el metalenguaje XML.

En este trabajo se implementó el sistema *HCDigital*, que permite gestionar repositorios de HCE a partir de la metodología propuesta, teniendo cuenta los diferentes factores de la Historia Clínica (HC) y los datos recogidos en cada uno de ellos a partir del interrogatorio y el examen físico, con el propósito de identificar automáticamente la relación de los pacientes a través de sus síntomas o signos.

Las unidades estructurales se definieron a partir de la recogida cronológica de los datos de una HCE basado en criterio de expertos, en la Figura 3.15 se muestran las unidades estructurales que se tuvieron en cuenta. Esta propuesta permite lograr una representación estandarizada del conocimiento y ofrece soporte a la toma de decisiones.

Al estar basado el agrupamiento de las HCE en los síntomas o signos existentes en estas y no sólo en el diagnóstico final, permite que el especialista pueda evaluar objetivamente el valor de diagnóstico de una prueba en particular sin interferir en los resultados de las demás pruebas.

Por otra parte, el beneficio de tener pacientes similares, con los mismos diagnósticos finales permite a los estudiantes de menos tiempo completar la HC de un paciente. Utilizando la metodología propuesta y la UE, donde se concentran las dudas, podría conseguir grupos de

casos similares que son recomendaciones sobre el uso correcto de un plan de tratamiento complementario, entre otras opciones.

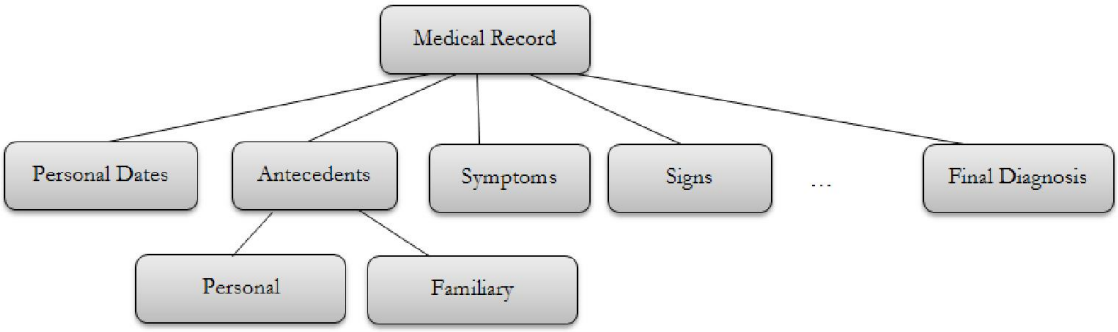


Figura 3.15 Unidades estructurales definidas para el sistema

Los resultados obtenidos por las colecciones de HCE agrupadas deben interpretarse teniendo en cuenta criterios de los expertos. El uso de reglas de asociación permitiría explicar las relaciones entre las HCE de los pacientes, que pertenecen al mismo grupo. Los centroides o HCE más relevante de cada grupo pueden permitir a los expertos estudiar los casos similares permitiendo pronósticos favorables.

3.4 Conclusiones Parciales

La medida de similitud propuesta, *SimRefBib*, facilita evaluar el grado de similitud de las referencias bibliográficas de dos artículos científicos en formato XML. Para la comparación de los documentos considera la relación existente entre las subunidades. Esta medida fue utilizada exitosamente en el algoritmo de agrupamiento *SemClustDML*, que se auxilia de las características especiales de la matriz *SimRefBib*, logrando buenos resultados al compararlos con otros algoritmos de agrupamiento.

La comparación de la función *SimRefBib* con otras funciones de similitud generales, arrojó que para las medidas Micro-Purity y Macro-Purity se obtienen diferencias significativas. En el caso de la medida OFM no se reportan diferencias significativas pero la función *SimRefBib* presenta un comportamiento más estable, por lo que se demuestra que resulta más conveniente tratar cada subunidad de las referencias bibliográficas de manera independiente que ver las referencias como una bolsa de palabras.

La implementación del sistema de recuperación de información *ScientificSolr* soporta la metodología. El sistema utiliza las facilidades de la herramienta *SolR* para realizar la recuperación de información, permitiendo la indexación de documentos remotos. *ScientificSolr* maneja documentos no estructurados; el agrupamiento de este tipo de documentos utilizando la metodología se logra a partir de una variante del método de segmentación por tópicos *TextLec*, considerando los tópicos como unidades estructurales.

La implementación de un sistema para el trabajo con Historias Clínicas, donde se utiliza el formato estándar HL7 y se definen unidades estructurales acordes a los datos; demuestra la facilidad de la metodología en dominios diferentes al trabajo con los artículos científicos.

CONCLUSIONES

Como resultado de esta investigación se diseñó una metodología para el agrupamiento automático de documentos semiestructurados en formato XML, combinando el contenido y la estructura existente en los mismos, para contribuir a la organización de la información recuperada.

- La metodología propuesta muestra que el agrupamiento combinando de forma apropiada las dimensiones estructura y contenido de los documentos XML obtiene buenos resultados al compararlos con otras variantes de agrupamiento existentes en la literatura. La metodología no es sensible a la utilización de otros algoritmos de agrupamiento.
- Se definió una nueva función de similitud denominada *OverallSimSUX* que permite capturar de forma eficiente y eficaz 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. Para los casos de estudio analizados al utilizar la metodología para el cálculo de la función *OverallSimSUX*, para cada algoritmo analizado y los casos de estudio seleccionados, cuando se utiliza la metodología, estos obtienen mejores resultados que cuando no se tiene en cuenta.
- Se comprobó que la metodología propuesta para documentos científicos en formato XML muestra resultados superiores o comparables cuando se realiza un tratamiento diferenciado a la Unidad Estructural Referencia Bibliográfica. La propuesta de función de similitud *SimRefBib*, facilita evaluar de forma eficiente el grado de similitud entre las Referencias Bibliográficas de dos artículos científicos en formato XML, pues captura el grado de semejanza existente entre las referencias bibliográficas, considerando la relación existente entre las subunidades. Los resultados al aplicar el algoritmo de agrupamiento *SemClustDML*, son superiores.
- El sistema *XMLearning*, es una aplicación implementada para evaluar la metodología que soporta la manipulación colecciones de documentos XML. Los resultados de la

evaluación demostraron que con los algoritmos seleccionados y los casos de estudio analizados, al utilizar la metodología se obtienen mejores resultados, que cuando esta no es utilizada.

- El sistema web *ScientificSolr* implementa la metodología para documentos en diferentes formatos. El sistema HCDigital para el trabajo con Historias Clínicas, trabaja con el formato estándar HL7, siguiendo la estandarización de estos documentos y se definen unidades estructurales acordes a los datos; obteniéndose una aplicación de la metodología diferente al trabajo con los artículos científicos.

RECOMENDACIONES

Derivadas del estudio realizado, así como de las conclusiones generales emanadas del mismo, se recomienda:

- Evaluar el efecto de la aplicación de la metodología a otros formatos de documentos semiestructurados (AIML, WSDL).
- Evaluar otros aspectos referentes a Unidades Estructurales o Subunidades que puedan ser determinantes en el agrupamiento de documentos
- Dotar a la herramienta *ScienticSolR* de nuevos requerimientos referidos a: la cantidad optima de tópicos, criterios para particionar los tópicos cuando la cantidad obtenida es menor que la deseada.

REFERENCIAS BIBLIOGRÁFICAS

1. Dalamagas, T., et al., *A Methodology for Clustering XML Documents by Structure.* Information Systems, 2006.
2. Abiteboul, S., *Querying semi-structured data.* Proceedings of the ICDT Conference, Delphi, Greece, 1997.
3. Guerrini, G., M. Mesiti, and I. Sanz, *An Overview of Similarity Measures for Clustering XML Documents.* 2006.
4. Packer, A.L., et al. *¿Porqué XML?* SciELO en Perspectiva 2014 [cited 2015 29/09/2015]; Available from: http://blog.scielo.org/es/2014/04/04/porque-xml/
5. Bueno, E. *Estado del arte y tendencias en creación y gestión del conocimiento*. in *Congreso Iberoamericano de Gestión del Conocimiento y la Tecnología (IBERGECYT 2001*. 2001. La Habana, Cuba.
6. Dalkir, K., *Knowledge Management in Theory and Practice*. 2005, Burlington, USA: Elsevier.
7. Canals, A., M. Boisot, and A. Cornella, *Gestión del conocimiento*. 2003, Gestión: 2000: Barcelona, España.
8. Dixon, M., *An overview of document mining technology*. 1997: http://www.geocities.com/ResearchTriangle/Thinktank/1997/mark/writings/dixm97_dm.ps.
9. Tan, A. *Text Mining: The state of the art and the challenges*. in *Proceedings of the Conference Knowledge Discovery and Data Mining (PAKDD'99): Workshop Knowledge Discovery from Advanced Databases*. 1999. Pacific Asia.
10. C., M., R. P., and S. H. *Introduction to Information Retrieval*. 2008. Cambridge University Press.
11. Tien T., R.N., *Evaluating the Performance of XML Document Clustering by Structure only.* 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, 2007.
12. Kruse, R., C. Döring, and M.-J. Lesor, *Fundamentals of Fuzzy Clustering*, in *Advances in Fuzzy Clustering and its Applications*, J.V.d. Oliveira and W. Pedrycz, Editors. 2007, John Wiley and Sons: Est Sussex, England. p. 3-27.
13. Kurgan, L., W. Swiercz, and K.J. Cios. *Semantic mapping of xml tags using inductive machine learning*. in *11th International Conference on Information and Knowledge Management.* 2002. Virginia, USA.
14. Shen, Y. and B. Wang. *Clustering schemaless xml document.* in *11th international conference on Cooperative Information System.* 2003.

15. Tran, T., R. Nayak, and P. Bruza. *Combining Structure and Content Similarities for XML Document Clustering*. in *Seventh Australasian Data Mining Conference*. 2008. Glenelg, Australia.
16. Aljaber, B., et al., *Document clustering of scientific texts using citation contexts Springer Science+Business Media.* 2009.
17. Hunter, L. and K. Cohen, *Biomedical language processing: What's beyond pubmed?* 2006.
18. Artiles, M., L. Arco, and D. Magdaleno, *Conjunto de Herramientas que Auxilian el Desarrollo de Sistemas Gestores de Información en Dominios Textuales.* Revista Ciencias Matemáticas, 2012. **26**(2): p. 135-145.
19. WANG, C. and D.M. BLEI. *Collaborative topic modeling for recommending scientific articles*. in *17th ACM SIGKDD international conference on Knowledge discovery and data mining.* 2011. San Diego, California, USA.
20. Nguyen, D.T., L. Chen, and C.K. Chan, *Clustering with multiviewpoint-based similarity measure.* Knowledge and Data Engineering, IEEE Transactions on, 2012. **24**(6): p. 988-1001.
21. Grossman, D.A. and O. Frieder, *Information retrieval: Algorithms and heuristics*. Vol. 15. 2012: Springer Science & Business Media.
22. Magdaleno Guevara, D., I.E. Fuentes Herrera, and M.M. García Lorenzo, *Clustering XML Documents using Structure and Content Based in a Proposal Similarity Function (OverallSimSUX).* Computación y Sistemas, 2015. **19**(1).
23. Magdaleno, D., I.E. Fuentes, and M.M. García, *Sistema para el Agrupamiento de Artículos Científicos en formato XML usando Lucene (LucXML)*, N.R. 3267-2013, Editor. 2013.
24. Henríquez Miranda, C. and J. Guzmán, *Extracción de información desde la web para identificar acciones de un modelo de dominio en planificación automática.* Ingeniare. Revista chilena de ingeniería, 2015. **23**(3): p. 439-448.
25. Díaz, A., et al., *AutoIndexer: Investigación y Desarrollo de Metodologías y Recursos Terminológicos de Apoyo para los Procesos de Indexación Automática de Documentos Clínicos.* Procesamiento del lenguaje natural, 2011. **47**: p. 343-344.
26. Sleiman, H.A., *Enterprise information integration unsupervised proposals for web information extraction*. 2012, Universidad de Sevilla.
27. Data, C.B., *Conociendo Big Data.* Revista Facultad de Ingeniería (Fac. Ing.), 2015. **24**(38): p. 63-77.
28. Martín, C., *Aprendizaje Automático y Minería de Datos con Modelos Gráficos Probababilísticos*, in *DEPARTAMENTO DE CIENCIAS DE LA COMPUTACION E INTELIGENCIA ARTIFICIAL*. 2007, Universidad de Granada: Granada. p. 43.
29. Piernik, M., et al., *XML clustering: a review of structural approaches.* The Knowledge Engineering Review, 2015. **30**(03): p. 297-323.
30. Deshmukh, M.V. and D.G. Bamnote, *An Empirical Study: XML Parsing using Various Data Structures.* International Journal of Computer Science and Applications, 2013. **6**(2).

31. Geetha, V., *Increasing Concurrency in XML Documents Using Semantic Locks.* International Journal of Database Theory and Application, 2014. **7**(6): p. 95-104.
32. Aggarwal, C.C. and C.K. Reddy, *Data Clustering: Algorithms and Applications*. Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, ed. C. Hall. 2013.
33. Ganesan, P., et al. *Comparative study of performance of distance measures in fuzzy C means clustering for CIELUV color images*. in *Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on*. 2014. IEEE.
34. Patidar, A.K., J. Agrawal, and N. Mishra, *Analysis of different similarity measure functions and their impacts on shared nearest neighbor clustering approach.* International Journal of Computer Applications (0975–8887) Vol, 2012. **40**.
35. Passoni, L., *Gestión del conocimiento: una aplicación en departamentos académicos.* Gestión y Política Pública, 2005. **XIV**(1): p. 57-74.
36. El-Sayed, M.A., *NEW SIMILARITY MEASURE BASED ON MINKOWSKI DISTANCE AND ITS APPLICATIONS IN FACE RECOGNITION.* Advances in Computer Science and Engineering, 2013. **11**(1): p. 1.
37. Taghva, K. and R. Veni, *Effects of Similarity Metrics on Document Clustering*, in *Seventh International Conference on Information Technology*. 2010
38. Anderberg, M.R., *Clustering Analysis for Applications*. 1973: New York: Academic.
39. SHANKAR, R., *Evolutionary Document Clustering and Summarization of Scientific Articles using Frequent Itemsets.* 2012.
40. Afonso, A.R. and C.G. Duque, *Automated text clustering of newspaper and scientific texts in brazilian portuguese: Analysis and comparison of methods.* 2014.
41. Amoli, P.V. and O.S. Sh, *Scientific Documents ClusteringBased on Text Summarization.* International Journal of Electricaland Computer Engineering (IJECE), 2015. **5**(4): p. 782~787.
42. Yau, C.-K., et al., *Clustering scientific documents with topic modeling.* Scientometrics, 2014. **100**(3): p. 767-786.
43. Guan, R., et al., *Full Text Clustering and Relationship Network Analysis of Biomedical Publications.* 2014.
44. GARFIELD, E., *Science citation index, a new dimension in indexing.* 1964.
45. Ferragina, P., *Beyond the bag-of-words paradigm to enhance information retrieval applications*, in *Proceedings of the Fourth International Conference on SImilarity Search and APplications*. 2011, ACM: Lipari, Italy. p. 3-4.
46. Salton, G., A. Wong, and C.S. Yang, *A vector space model for automatic text retrieval.* Communications of the ACM, 1975. **18**(11): p. 613-620.
47. Chen, Q., Y. Chen, and M. Jiang, *Cluster Analysis Based on Contextual Features Extraction for Conversational Corpus.* Journal of Computer and Communications, 2015. **3**(05): p. 33.
48. Erk, K., *Vector space models of word meaning and phrase meaning: A survey.* Language and Linguistics Compass, 2012. **6**(10): p. 635-653.

49. Huang, E.H., et al. *Improving word representations via global context and multiple word prototypes*. in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. 2012. Association for Computational Linguistics.
50. Xiong, H., J. Wu, and J. Chen. *K-means clustering versus validation measures: a data distribution perspective*. in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD 2006)*. 2006. Philadelphia, PA, USA: ACM Press.
51. Frigui, H. and O. Nasraoui, *Simultaneous clustering and dynamic keyword weighting for text documents.* 2001.
52. Shin, K. and S.Y. Han, *Fast clustering algorithm for information organization.*, in *In:Proc. of the CICLing Conference*. 2003, Lecture Notes in Computer Science.Springer-Verlag (2003). p. 619–622.
53. Cheng, D., et al., *A divide-and-merge methodology for clustering.* ACM Transaction on Database Systems (TODS), 2006. **31**(4): p. 1499-1525.
54. Cheng, D., et al. *A divide-and-merge methodology for clustering*. in *Proceedings of the 24th ACM SIGMOD SIGACT SIGART Symposium on Principles of Database Systems (PODS 2005)*. 2005. Baltimore, Maryland: ACM Press.
55. Höppner, F., et al., *Fuzzy cluster analysis: methods for classification, data analysis and image recognition.* 1999, West Sussex, England: John Wiley & Sons Ltd.
56. ASLAM, J., E. PELEKHOV, and R. D., *The star clustering algorithm for static and dynamic information organization.* Journal of Graph Algorithms and Applications,, 2004. **8**.
57. Pérez, A. and J.E. Medina. *A clustering algorithm based on generalized stars*. in *Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM 2007)*. 2007. Leipzig, Germany: Springer Verlag.
58. Dasgupta, S., S. Bhat, and Y. Lee, *STC: Semantic Taxonomical Clustering for Service Category Learning.* arXiv preprint arXiv:1303.5926, 2013.
59. Tang, J., et al. *DC-Tree: Density-Based Clustering Index for Objects in Skewed Distribution*. in *Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2012 IEEE 21st International Workshop on*. 2012. IEEE.
60. Tetali, R., J. Bose, and T. Arif. *Browser with Clustering of Web Documents*. in *Advanced Computing, Networking and Security (ADCONS), 2013 2nd International Conference on*. 2013. IEEE.
61. Alexandrov, M., A. Gelbukh, and P. Rosso, *An approach to clustering abstracts*, in *Natural Language Processing and Information Systems*. 2005, Springer. p. 275-285.
62. Pinto, D. and P. Rosso. *On the relative hardness of clustering corpora*. in *Text, Speech and Dialogue*. 2007. Springer.
63. Agustı, L., et al., *A new grouping genetic algorithm for clustering problems.* Expert Systems with Applications, 2012. **39**(10): p. 9695-9703.

64. Kuo, R., et al., *Integration of particle swarm optimization and genetic algorithm for dynamic clustering.* Information Sciences, 2012. **195**: p. 124-140.
65. Pinto, D., M. Tovar, and D. Vilariño. *BUAP: Performance of K-Star at the INEX'09 Clustering Task*. in *INEX 2009 Workshop Pre-proceedings*. 2009. Woodlands of Marburg, Ipswich, Queensland,Australia.
66. Camps, J., et al., *Diseño de Funciones de Similitud para el Razonamiento Basado en Casos usando Programación Genética: estudio con problemas sintéticos.* 2009.
67. Wilson, D.R. and T.R. Martínez, *Improved heterogeneous distance functions.* Journal of Artificial Intelligence Research, 1997. **6**: p. 1-34.
68. Huang, A., *Similarity Measures for Text Document Clustering*, in *NZCSRSC* 2008: Christchurch, New Zealand.
69. Strehl, A., J. Ghosh, and R. Mooney. *Impact of similarity measures on Web-page clustering*. in *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI 2000): Workshop of Artificial Intelligence for Web Search*. 2000. Austin, Texas.
70. Lian, W., et al., *An efficient and scalable algorithm for clustering XML documents by structure.* Knowledge and Data Engineering, IEEE Transactions on, 2004. **16**(1): p. 82-96.
71. Brzezinski, D., et al., *XCleaner: a new method for clustering XML documents by structure.* Control Cybern 2011. **40**(3): p. 877-891.
72. Nayak, R. *Investigating Semantic Measures in XML Clustering*. in *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*. 2006. IEEE.
73. Flesca, S., et al., *Fast detection of XML structural similarities.* IEEE Trans. Knowl. Data Engin., 2005. **7**(2): p. 160-175.
74. Nierman, A. and H.V. Jagadish, *Evaluating structural similarity in XML documents*, in *5th Int. Conf. Computational Science (ICCS'05)*. 2002.
75. Chawathe, S.S., et al. *Change Detection in Hierarchically Structured Information*. in *In Proceedings of the ACM International Conference on Management of Data*. 1996.
76. Chawathe, S.S. *Comparing Hierarchical Data in External Memory*. in *In Proceedings of International Conference on Very Large Databases*. 1999.
77. Zhang, K. and D. Shasha, *Simple Fast Algorithms for the Editing Distance Between Trees and Related Problems.* . SIAM Journal of Computing, 1989. **18**(6): p. 1245-1262.
78. Selkov, S.M., *The Tree-to-Tree Editing Problem.* Information Processing Letters, 1977. **6**: p. 184-186.
79. Kutty, S., et al., *Combining the structure and content of XML documents for clustering using frequent subtrees.* INEX, 2008: p. 391-401.
80. Lian, W., et al., *An Efficient and Scalable Algorithm for Clustering XML Documents by Structure.*, in *TKDEE* 2004. p. 82-96.
81. Goldman, R. and J. Widom. *DataGuides: Enabling Query Formulation and Optimization in Semistructured Databases.* in *In Proceedings of International Conference on Very Large Databases.* 1997.

82. Xing, G., Z. Xia, and J. Guo, eds. *Clustering XML Documents Based on Structural Similarity, LNCS 4443, pp. 905–911*. ed. L.N.i.C. Science. 2007, Springer.
83. Kirsten, M. and S. Wrobel. *Extending k-means clustering to first-order representations.* in *Proceedings of the 10th International Conference on Inductive Logic Programming.* 2000.
84. Rendón, E., et al., *Internal versus external cluster validation indexes.* International Journal of computers and communications, 2011. **5**(1): p. 27-34.
85. Stein, B., S.M. zu Eissen, and F. Wißbrock. *On cluster validity and the information need of users*. in *Proc. Artificial Intelligence and Applications*. 2003.
86. Manning, C.D., P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Vol. 1. 2008: Cambridge university press Cambridge.
87. Aggarwal, C.C. and H. Wang, *Graph data management and mining: A survey of algorithms and applications*, in *Managing and Mining Graph Data*. 2010, Springer. p. 13-68.
88. Hagenbuchner, M., et al., *Efficient clustering of structured documents using graph self-organizing maps*, in *Focused Access to XML Documents*. 2008, Springer. p. 207-221.
89. Doucet, A. and H. AhonenMyka, *Naive clustering of a large XML document collection.* INEX, 2002: p. 84-89.
90. Giannopoulos, P. and R.C. Veltkamp. *A Pseudo-Metric for Weighted Point Sets.* in *In Proceedings of the 7th European Conference on Computer Vision (ECCV)*. 2002.
91. Karmarkar, N., *A new polynomial-time algorithm for linear programming*, in *Proceedings of the 16th Annual ACM Symposium on the Theory of Computing*. 1984.
92. Yang, W. and X.O. Chen, *A semi-structured document model for text mining.* Journal of Computer Science and Technology, 2002. **17**(5): p. 603-610.
93. Wan, X. and J. Yang, *Using Proportional Transportation Similarity with Learned Element Semantics for XML Document Clustering.* International World Wide Web Conference Committee, 2006.
94. Tran, T., S. Kutty, and R. Nayak, *Utilizing the Structure and Data Information for XML Document Clustering.* INEX, 2008: p. 402-410.
95. Cristianini, N., J. Shawe-Taylor, and H. Lodhi, *Latent semantic kernels.* JJIS'2002, 2002. **18**(2).
96. Nayak, R. and S. Xu. *XCLS: A Fast and Effective Clustering Algorithm for Heterogenous XML Documents.* in *10th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD).* . 2006. Singapore: LNCS, p 292-302.
97. Powers, D.M., *Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.* 2011.
98. Steinbach, M., G. Karypis, and V. Kumar. *A comparison of document clustering techniques*. in *Proceedings of 6th ACM SIGKDD World Text Mining Conference*. 2000. Boston: ACM Press.
99. Davies, D.L. and D.W. Bouldin, *A cluster separation measure.* IEEE Transactions on Pattern Analysis and Machine Learning, 1979. **1**(4): p. 224-227.

100. Newman, M.E.J. and M. Girvan, *Finding and evaluating community structure in networks.* Physical Review E, 2004. **69**(2): p. 026113.
101. Yanchi, L., et al., *Understanding and Enhancement of Internal Clustering Validation Measures.* Cybernetics, IEEE Transactions on, 2013. **43**(3): p. 982-994.
102. Shannon, C.E., *A mathematical theory of communications.* The Bell System Technical Journal, 1948. **27**(3): p. 379-423.
103. Rosell, M., V. Kann, and J.E. Litton. *Comparing comparisons: document clustering evaluation using two manual classifications*. in *Proceedings of International Conference on Natural Language processings ICON*. 2004. Hyderabad, India.
104. Frakes, W.B. and R. Baeza-Yates, *Information Retrieval. Data Structure & Algorithms*. 1992, New York: Prentice Hall.
105. Steinbach, M., G. Karypis, and V. Kumar. *A comparison of document clustering techniques*. in *Proceedings of 6th ACM SIGKDD World Text Mining Conference*. 2000. Boston: ACM Press.
106. Pestian, J.P., et al., *Sentiment analysis of suicide notes: A shared task.* Biomedical informatics insights, 2012. **5**(Suppl 1): p. 3.
107. Lamirel, J.-C., et al., *A new efficient and unbiased approach for clustering quality evaluation*, in *New Frontiers in Applied Data Mining*. 2012, Springer. p. 209-220.
108. De Vries, C.M., et al., *Overview of the INEX 2010 XML mining track: Clustering and classification of XML documents*, in *Comparative evaluation of focused retrieval*. 2011, Springer. p. 363-376.
109. De Vries, C.M., S. Geva, and A. Trotman, *Document clustering evaluation: Divergence from a random baseline.* arXiv preprint arXiv:1208.5654, 2012.
110. Hatcher, E., O. Gospodnetic, and M. McCandless, *Lucene in Action*. Second ed. 2009.
111. MATTMANN, C.A. and J.L. ZITTING, *Tika in Action*. 2012, 20 Baldwin Road PO Box 261 Shelter Island, NY 11964: Manning Publications Co.
112. Lewis, D.D. and M. Ringuette. *A comparison of two learning algorithms for text classification*. in *Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval*. 1994. University of Nevada, Las Vegas.
113. Grainger, T. and T. Potter, *Solr in action*. 2014.
114. Lanquillon, C., *Enhancing Text Classification to Improve Information Filtering*, in *Research Group Neural Networks and Fuzzy Systems*. 2001, University of Magdeburg "Otto von Guericke": Magdeburg. p. 231.
115. Singh, S. and T.J. Siddiqui. *Evaluating effect of context window size, stemming and stop word removal on Hindi word sense disambiguation*. in *Information Retrieval & Knowledge Management (CAMP), 2012 International Conference on*. 2012. IEEE.
116. Zaman, A., P. Matsakis, and C. Brown. *Evaluation of stop word lists in text retrieval using Latent Semantic Indexing*. in *Digital Information Management (ICDIM), 2011 Sixth International Conference on*. 2011. IEEE.
117. Amarasinghe, K., M. Manic, and R. Hruska. *Optimal stop word selection for text mining in critical infrastructure domain*. in *Resilience Week (RWS), 2015*. 2015. IEEE.

118. Berry, M.W., *Survey of Text mining: Clustering, Classification, and Retrieval*. 2004, New York, USA: Springer Verlag.
119. Arco, L., *Agrupamiento basado en la intermediación diferencial y su valoración utilizando la teoría de los conjuntos aproximados*, in *Ciencias de la Computación*. 2009, Universidad Central "Marta Abreu" de Las Villas: Santa Clara, Villa Clara. p. 187.
120. Lee, C.-H. and S.-H. Wang, *An information fusion approach to integrate image annotation and text mining methods for geographic knowledge discovery.* Expert Systems with Applications, 2012. **39**(10): p. 8954-8967.
121. Xia, T. and Y. Du. *Improve VSM text classification by title vector based document representation method*. in *Computer Science & Education (ICCSE), 2011 6th International Conference on*. 2011. IEEE.
122. Huang, C., et al., *Emergency Case Retrieval Based on Fuzzy Sets and Text Mining*, in *Foundations of Intelligent Systems*. 2014, Springer. p. 911-919.
123. Magdaleno, D., et al., *New Textual Representation using Structure and Contents.* Research in Computing Science, 2011. **54**(Advances in Softcomputing algorithms): p. 117-130.
124. Ruiz-Shulcloper, J., *Introducción al reconocimiento de patrones. Enfoque lógico combinatorio*. 1995, México: CINVESTAV IPN.
125. Perez-Tellez, F., et al., *Improving the Clustering of Blogosphere with a Self-term Enriching Technique. Text, Speech and Dialogue.* Springer, 2009: p. 40-47.
126. Garcia, E., *Cosine similarity and term weight tutorial.* Information retrieval intelligence, 2006.
127. Wiese, A., V. Ho, and E. Hill. *A comparison of stemmers on source code identifiers for software search*. in *Software Maintenance (ICSM), 2011 27th IEEE International Conference on*. 2011. IEEE.
128. Chen, K. and L. Liu. *ClusterMap: labeling clusters in large datasets via visualization*. in *Proceedings of the ACM IEEE 13th Conference on Information and Knowledge Management (CIKM 2004)*. 2004. Washington, D.C.
129. Jain, A.K., M.N. Murty, and P.J. Flynn, *Data clustering: a review.* ACM Computing Surveys, 1999. **31**(3): p. 264-323.
130. Magdaleno, D., et al., *Comparative Study of Clustering Algorithms using OverallSimSUX Similarity Function for XML Documents.* Inteligencia Artificial, 2015. **18**(55): p. 69-80.
131. Magdaleno, D., et al., *Sistema para el procesamiento y agrupamiento de Artículos Científicos en formato XML (XMLearning)*, in *3726-11-2014*. 2014.
132. Theodoridis, S. and K. Koutroubas, *Pattern Recognition*. 1999: Academic Press.
133. Halkidi, M., Y. Batistakis, and M. Vazirgiannis, *Clustering validity checking methods: Part II.* ACM SIGMOD Record, 2002. **31**(3): p. 19-27.
134. Silberschatz, A. and A. Tuzhilin, *What makes patterns interesting in knowledge discovery systems.* IEEE Transactions on Knowledge and Data Engineering, 1996. **8**(6): p. 940-974.

135. Kaufman, L. and P.J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. Wiley Series in probability and mathematical statistics. 1990: John Wiley and Sons.
136. Rosell, M., V. Kann, and J.-E. Litton. *Comparing comparisons: document clustering evaluation using two manual classifications*. in *Proceedings of the International Conference on Natural Language Processing (ICON 2004)*. 2004. Hyderabad, India: Allied Publishers.
137. Sheskin, D.J., *Handbook of Parametric and Nonparametric statistical procedures*, ed. C. Hall/CRC. 2004, New York. 1184.
138. Demšar, J., *Statistical comparisons of classifiers over multiple data sets.* The Journal of Machine Learning Research, 2006. **7**: p. 1-30.
139. COSTAS, R. and M. BORDONS, *Algoritmo para solventar la falta de normalización de nombres de autores en los estudios bibliométricos.* 2006.
140. Alonso, L.E., Y. Hidalgo, and A.A. Leiva, *Desambiguación del nombre de los autores.* Revista Cubana de Ciencias Informáticas, 2014. **8**: p. 149-169.
141. Winkler, W.E., *Using the EM algorithm for weight computation in the Felligi -Sunter Model of Record Linkage.* 2000.
142. Rojas, L.H., *Método para la Segmentación de Textos por Tópicos usando una Ventana de Párrafos Inferiores para medir la Cohesión Léxica*, in *Ciencias de la Computación*. 2007, UCLV: Villa Clara.
143. Caro, E.M., *El párrafo como unidad discursiva: consideraciones de forma y contenido relativas a su demarcación y estructuración.* 2014.
144. Google. *Code Google*. 2010; Available from: http://code.google.com/intl/es/webtoolkit/overview.html.
145. Zwaanswijk, M., et al., *Benefits and problems of electronic information exchange as perceived by health care professionals: an interview study.* BMC health services research, 2011. **11**(1): p. 256.
146. J., G., *La historia clínica electrónica: muchas promesas y pocos hechos*, in *XXVIII CONGRESO DE MEDICINA DE FAMILIA Y COMUNITARIA*, A. Primaria, Editor. 2008. p. 13.
147. Batchelor, B., *Pattern Recognition: Ideas in Practice*. 1978, New York: Plenum Press.
148. Hand, D.J., *Discrimination and classification*. Wiley Series in Probability and Statistics. 1981: John Wiley and Sons.
149. Michalski, R.S., R.E. Stepp, and E. Diday, *A recent advance in data analysis: clustering objects into classes characterized by conjuntive concepts.* Progress in Pattern Recognition, 1981. **1**: p. 33-56.
150. Duch, W., *Similarity-based methods: a general framework for classification.* Control and Cybernetics, 2002. **29**(4): p. 937-968.
151. Niu, Z.-Y., D.-H. Ji, and C.-L. Tan. *Document clustering based on cluster validation*. in *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management (CIKM 2004)*. 2004. Washington, D.C., USA: ACM Press.

152. Sahami, M., *Using machine learning to improve informatio access*, in *Department of Computer Science*. 1998, Stanford University: Standford, USA.

153. Zipf, G.K., *Human Behaviour and the Principle of Least Effort*. 1949: Addison-Wesley.

154. Rijsberguen, C.J., *Information Retrieval*. 1979, London: Butterworths.

155. Yang, Y. and J.O. Pedersen. *A comparative study on feature selection in text categorization*. in *Proceedings of the Fourteenth International Conference on Machine Learning*. 1997. San Francisco, US: Morgan Kaufmann Publishers.

156. Salton, G. and C. Buckley, *Term weighting approaches in automatic text retrieval.* Information Processing and Management, 1988. **24**(5): p. 513-523.

157. Salton, G. and M. McGill, *Introduction to Modern Information Retrieval*. 1983, New York, USA: McGraw-Hill.

158. Nürnberger, A., A. Klose, and R. Kruse. *Clustering of document collection to support interactive text exploration*. in *Proceedings of the 25th Annuals Conference of the Gesellschaft für Klassification. Studies in Classification, Data Analysis and Knowledge Organization. Exploratory Data Analysis in Empirical Research*. 2001. Germany.

159. Fukuhara, T., H. Takeda, and T. Nishida. *Multiple-text summarization for collective knowledge formation*. in *Proceedings of the IEEE International Conference om Systems, Man and Cybernetics*. 1999. Tokyo: IEEE Press.

PRODUCCIÓN CIENTÍFICA DEL AUTOR

Publicaciones

- **Damny Magdaleno**, Ivett E. Fuentes, María M. García. "Clustering XML Documents Using Structure and Content based on a New Similarity Function OverallSimSUX". Computación y Sistemas Vol.19 N^{ro}.1, 2015. Indexado por Scopus, Thomson Reuters Web of Science. ISSN 2007-9737.
- **Damny Magdaleno**, Yadriel Miranda, Ivett E. Fuentes, María M. García. "Comparative Study of Clustering Algorithms using OverallSimSUX Similarity Function for XML Documents". Iberoamerican Journal of Artificial Intelligence. Indexado por Scopus. ISSN 1988-3064. Vol.18 N^{ro}.55, 2015. pp 69-80. doi: 10.4114/ia.v18i55.1098
- Michel Artiles, Leticia Arco, **Damny Magdaleno**. "Conjunto de Herramientas que Auxilian el Desarrollo de Sistemas Gestores de Información en Dominios Textuales". Revista Ciencias Matemáticas, Vol 26, No. 2, pp 135-145. ISSN 0256-5374. 2012. Indexado Latindex-Catálogo, Latindex-Directorio, Periódica.
- **Damny Magdaleno**, et al. "New Textual Representation using Structure and Contents". Research in Computing Science, Vol. 54, No. Advances in Softcomputing algorithms, pp. 117-130. ISSN 1870-4069. Indexada en: DBLP, LatIndex, Periódica.

Eventos

- Ivett E. Fuentes, **Damny Magdaleno**, María M. García. Fifth international workshop on Knowledge Discovery, Knowledge Management and Decision Support, EUREKA 2015. "Methodology for discovery of implicit knowledge in Medical Records".
- Ivett E. Fuentes, **Damny Magdaleno**, María M. García. UCIENCIA 2014. "LUCXML: Sistema para el agrupamiento de artículos científicos en formato XML utilizando estructura y contenido".
- **Damny Magdaleno**, Leticia Arco, Michel Artiles, Juan M. Fernández, Juan Huete. 10th Mexican International Conference on Artificial Intelligence. TESM (Mexico), Instituto Politécnico Nacional (Mexico), UAEH (Pachuca, Mexico), UNAM (Mexico), Instituto Mexicano de Petroleo (Mexico), WIQ-EI (Web Information Quality Evaluation Initiative, European project 269180). Puebla, México. Noviembre 26 - Diciembre 4, 2011 "New Textual Representation using Structure and Contents".
- **Damny Magdaleno**, Leticia Arco, Michel Artiles, Juan M. Fernández, Juan Huete. XII Congreso Nacional de Matemática y Computación (COMPUMAT 2011). Santa Clara,

Cuba. 23 al 25 de Noviembre del 2011. "Nueva Representación Textual Combinando Contenido y Estructura".

- Michel Artiles, **Damny Magdaleno**, Leticia Arco. XII Congreso Nacional de Matemática y Computación (COMPUMAT 2011). Santa Clara, Cuba. 23 al 25 de Noviembre del 2011. "Automatización de la Gestión Documental en el Centro de Estudios de Informática de la UCLV: Una Experiencia Generalizable.
- Michel Artiles, Leticia Arco, **Damny Magdaleno**. XII Congreso Nacional de Matemática y Computación (COMPUMAT 2011). Santa Clara, Cuba. 23 al 25 de Noviembre del 2011. "Conjunto de Herramientas que Auxilian el Desarrollo de Sistemas Gestores de Información en Dominios Textuales".
- **Damny Magdaleno**, Leticia Arco, Michel Artiles, Juan M. Fernández, Juan Huete. Cuba-Flanders WorkShop on Machine Learning and Knowledge Discovery. Santa Clara, Cuba. Fecha: 23 al 25 de Noviembre del 2011. "Textual Representation using Structure and Content for XML Documents Clustering".
- **Damny Magdaleno**, Leticia Arco, Michel Artiles. Tercer taller internacional de descubrimiento de conocimiento, gestión del conocimiento y toma de decisiones (Taller EUREKA). Universidad de Cantabria. Santander, España. 26 al 29 de Octubre de 2011. "Textual Representation for Knowledge Discovery".
- Michel Artiles, Leticia Arco, **Damny Magdaleno**. Tercer taller internacional de descubrimiento de conocimiento, gestión del conocimiento y toma de decisiones (Taller EUREKA). Universidad de Cantabria. Santander, España. 26 al 29 de Octubre de 2011. "Aplicaciones para el Desarrollo de Sistemas Gestores de Información en Dominios Textuales".
- **Damny Magdaleno**, Leticia Arco, Michel Artiles, Juan M. Fernández, Juan Huete. Conferencia Internacional de Ciencias Computacionales e Informáticas, INFORMATICA 2011.Habana, Cuba. 7 al 11 de Febrero del 2011. "Enfoque para Agrupar Documentos XML Contribuyendo a la Gestión del Conocimiento".

Monografías

- **Damny Magdaleno**, Leticia Arco. "Representación de Documentos XML, un Enfoque para el Agrupamiento, Aplicaciones en la Gestión del Conocimiento". Centro de Estudios de Informática. Editorial Feijó / 2011. ISBN: 978-959-250-738-8.

Registros de Software

- **Damny Magdaleno**, Ivett E. Fuentes, Yadriel Miranda, María M. García: "Sistema para el procesamiento y agrupamiento de Artículos Científicos en formato XML (XMLearning)". 2014. No Reg: 3726-11-2014
- **Damny Magdaleno**, Ivett E. Fuentes, María M. García. "Sistema para el Agrupamiento

de Artículos Científicos en formato XML usando Lucene (LucXML)". 2013. No Reg: 3267-2013

Premios

- Leticia Arco, **Damny Magdaleno**, Michel Artiles. "Automatización de la gestión de documentos combinando contenido y estructura". Premio CITMA Provincial. 2012.

ANEXOS

Anexo 1. Similitudes, distancias más usadas para comparar objetos

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 [34]

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

Distancia Minkowski [147] Minkowski [36]

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

La distancia Minkowsky es equivalente a la distancia Manhattan o city-block City-Block [33], y a la distancia Euclidiana cuando γ es 1 y 2, respectivamente[147]. Para los valores de $\gamma \geq 2$, la distancia Minkowsky equivale a Supermum [148].

Distancia Euclidiana heterogénea (HeterogenousEuclidean – OverlapMetric; HEOM)

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

$$d_{local}\left(o_{ik}, o_{jk}\right)=\begin{cases} d_{Overlap}\left(o_{ik}, o_{jk}\right) & si\ k\ simbólico \\ d_{NormEuclidean}\left(o_{ik}, o_{jk}\right) & si\ k\ numéric \end{cases} \tag{A1.3}$$

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

Distancia Camberra [149]

$$D_{Camberra}(O_i, O_j) = \sum_{k=1}^{m} \frac{|o_{ik} - o_{jk}|}{|o_{ik} + o_{jk}|} \quad \text{(A1.4)}$$

Correlación de Pearson[67]

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

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 [67]. En[150] 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[104]. Una valoración del impacto de la distancia Euclidiana y los coeficientes Dice, Jaccard y Coseno en dominios textuales se presenta en[69].

Coeficiente Dice

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

Coeficiente de Jaccard

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

Coeficiente Coseno

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

Anexo 2. Algunas medidas externas para la validación del agrupamiento

Medida-*F* Global (Overall *F*-Measure; OFM)[105]

$$Overall\,F-Measure=\sum_{i=1}^{k}\frac{n_i}{n}\max\{F-Measure(i,j)\} \qquad \text{(A12.1)}$$

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 $F\text{-}Measure(i,j)$ se calcula según la expresión siguiente. Si $\alpha = 1$, entonces OFM se nombra Purity[103].

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

$$F-Measure(i,j)=\frac{1}{\alpha\left(1/\Pr(i,j)\right)+\left(1-\alpha\right)\left(1/\mathrm{Re}(i,j)\right)} \qquad \text{(A12.2)}$$

Si $\alpha = 1$ entonces $F\text{-}Measure(i,j)$coincide con precision, si $\alpha = 0$ entonces $F\text{-}Measure(i,j)$coincide con cubrimiento. $\alpha = 0.5$ significa igual peso para precisión y cubrimiento.

Micro-averaged precision y micro-averaged recall [151]

$$\text{MA - Pr}=\frac{\sum_{i=1}^{k}\alpha_i}{\sum_{i=1}^{k}\left(\alpha_i+\beta_i\right)} \quad \text{y} \quad \text{MA - Re}=\frac{\sum_{i=1}^{k}\alpha_i}{\sum_{i=1}^{k}\left(\alpha_i+\gamma_i\right)} \qquad \text{(A12.3)}$$

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 propuestas por INEX

$$Purity(k) = \frac{NDMLC_k}{NDC_k} \tag{A12.1}$$

$$Micro - Purity(k) = \frac{\sum_{k=0}^{n} Purity(k) * TotalFoundByClass(k)}{\sum_{k=0}^{n} TotalFoundByClass(k)} \tag{A12.2}$$

$$Macro - Purity(k) = \frac{\sum_{k=0}^{n} Purity(k)}{TotalofCategories} \tag{A12.3}$$

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

Anexo 3. Algunas medidas de calidad de términos

Umbral de frecuencia de términos y Ley de Zpif. Eliminar los términos que tienen o muy alta o muy baja frecuencia de aparición [152], a partir de un cálculo adecuado del umbral y del estudio de la Ley de Zipf [153]. Términos que raramente aparecen en una colección de documentos tendrán poco poder discriminante y pueden ser eliminados [154]. En contraste, términos con frecuencia de aparición alta se asumen que son comunes y que tampoco tienen poder discriminante[34].

Umbral de frecuencia de documentos. Teniendo en cuenta que $n(t)$ es el número de documentos en los cuales el término t aparece al menos una vez, una heurística simple de selección es excluir todos los términos desde el vocabulario cuya frecuencia de documentos es menor que algún umbral, ya que términos que ocurren en sólo muy pocos documentos improbablemente llevan información que permita distinguir los grupos textuales y tienden a ser ruidosos [155]. Además, usar la ocurrencia de términos infrecuentes no es confiable estadísticamente. Al eliminar estos términos se mantiene el poder discriminante y se mejora la efectividad del agrupamiento y clasificación textual.

Frecuencia inversa de documento y TFIDF. La importancia de los términos se asume inversamente proporcional al número de documentos en los cuales el término particular aparece. Después de eliminar las palabras de parada, la importancia de un término se incrementa con su frecuencia de uso. Combinando estas ideas se formuló la medida frecuencia del término / frecuencia inversa de documentos (*tfidf*).

Dado un corpus D y un documento d_j ($d_j \in D$), el valor TF-IDF para un término t_i en d_j es calculado como el producto entre la frecuencia normalizada del término t_i en el documento d_j y la frecuencia inversa del término en el corpus [65].

$$tf_{ij} = \frac{frecuencia(t_i, d_j)}{\sum_{s=1}^{|d_j|} frecuencia(t_s, d_j)} \qquad \text{A1.1}$$

[34] Términos con alta frecuencia de aparición pueden formar parte de la lista de palabras de parada automáticamente construida desde la colección de documentos. En esta tesis se considera que la lista es suministrada.

$$idf(i) = log\left(\frac{|D|}{|d: t_i \in d, d \in D|}\right) \quad \text{A1.2}$$

$$tf_{ij} - idf_i = tf_{ij} \times idf(i) \quad \text{A1.3}$$

Una combinación similar de frecuencia de términos y frecuencia inversa de documentos es se utiliza usualmente para asignar pesos a los términos [156].

Razón de señal a ruido. Medir el poder discriminante que transmite cada término, basado en el ruido R(*t*), como la entropía de la distribución de la probabilidad del término *t* entre los documentos [157]:

$$\text{SNR}(t)=\log tf(t) - \text{R}(t),\ \text{R}(t) = -\sum_{j=1}^{n} \text{P}(d_j,t)\log \text{P}(d_j,t) \text{ y } \text{P}(d_j,t) = \frac{tf_{d_j}(t)}{tf(t)} \quad \text{(A1.1)}$$

Entropía. Calcular la entropía como una medida de importancia, según Lochbaum y Streeter en 1989 [158]:

$$\text{Entropía}(t) = 1 + \frac{1}{\ln(n)} \sum_{i=1}^{n} p_i(t) \cdot \ln(p_i(t)) \quad \text{donde } p_i(t) = \frac{tf_{d_i}(t)}{\sum_{j=1}^{n} tf_{d_j}(t)} \quad \text{(A3.2)}$$

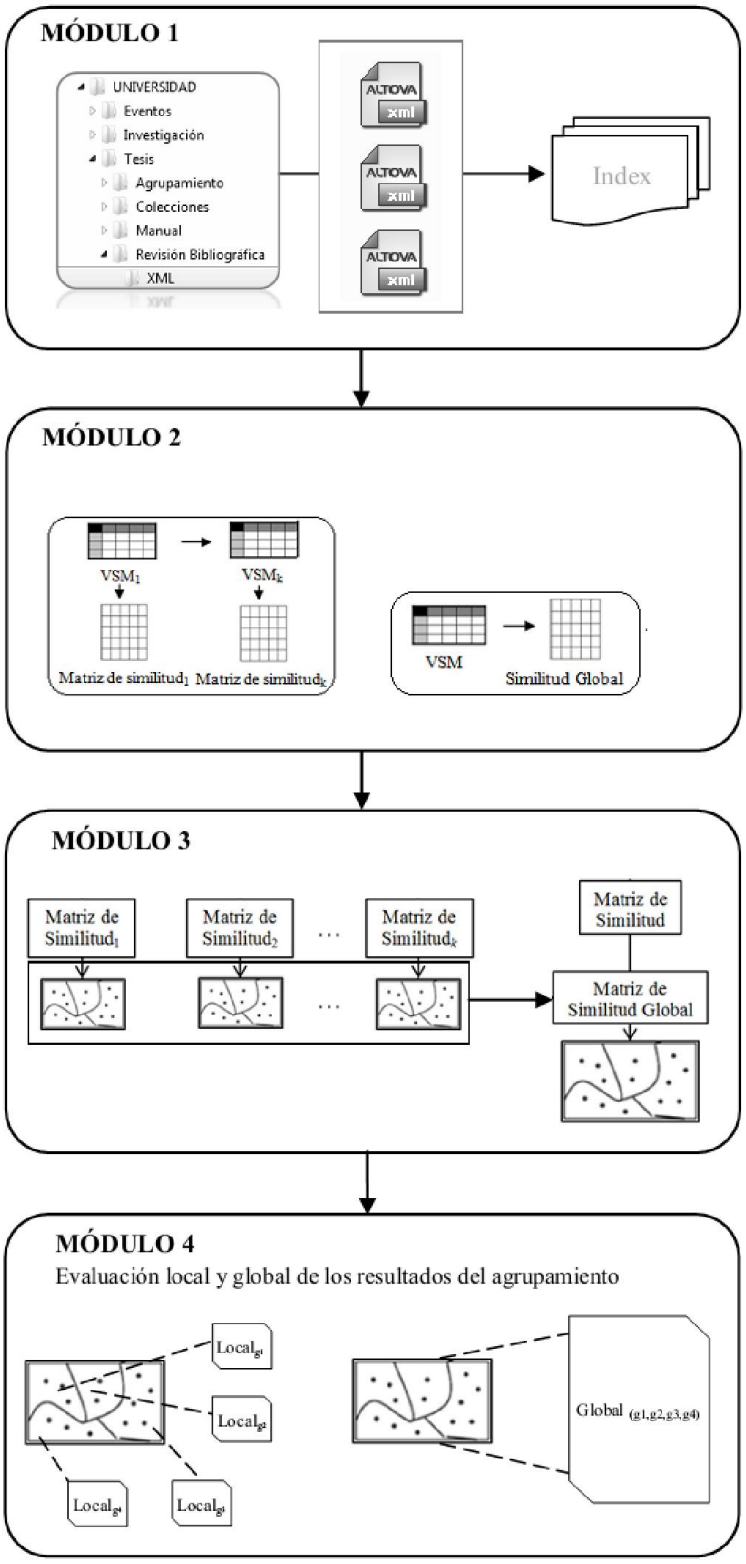
Calidad de términos. Medir la calidad de los términos según las expresiones q_0 y q_1, la segunda constituye una variante de la primera donde n_1 es el número de documentos en los cuales *t* ocurre al menos una vez [118].

$$q_0(t) = \sum_{j=1}^{n} \left(tf_{d\,j}(t)\right)^2 - \frac{1}{n}\left[\sum_{j=1}^{n} tf_{d\,j}(t)\right]^2 \text{ y } q_1(t) = \sum_{j=1}^{n_1} \left(tf_{d\,j}(t)\right)^2 - \frac{1}{n_1}\left[\sum_{j=1}^{n_1} tf_{d\,j}(t)\right]^2 \quad \text{(A3.3)}$$

Skewness y Kurtosis. Calcular la parcialidad de los términos mediante la combinación de Skewness y Kurtosis según P(*t*)=w_1·Skewness(*t*)+w_2·Kurtosis(*t*), donde w_1 y w_2 son pesos positivos y *s* es la desviación estándar de la ocurrencia del término *t* en la colección de documentos [159].

$$\text{Skewness}(t) = \frac{1}{n}\sum_{i=1}^{n}\frac{\left(tf_{d_i}(t) - \frac{tf(t)}{n}\right)^3}{s^3} \text{ y } \text{Kurtosis}(t) = \frac{1}{n}\sum_{i=1}^{n}\frac{\left(tf_{d_i}(t) - \frac{tf(t)}{n}\right)^4}{s^4} - 3 \qquad \text{(A3.4)}$$

Anexo 4. Modelo general para el agrupamiento



Anexo 5. Fragmento de uno de los documentos perteneciente al Corpus 9

```
<?xml version="1.0" encoding="utf-8"?>
   <paper>
   <class>0</class>
      <front>
        <title>Maximum Weight Matching via Max-Product Belief Propagation
        </title>
         <conference/>
         <authors>
            <author>
               <fname>Bayati</fname>
               <surname>Mohsen<surname/>
                        …
            </author>
                 …
         </authors>
         <keywords>
            <keyword>belief propagation <keyword/>
                       …
         </keywords>
         <abstract>
           <para>The max-product "belief propagation" algorithm is an iterative,
           local, message passing algorithm finding the … </para>
         </abstract>
       </front>
       <body>
          <section>
             <title>Introduction</title>
            <para>I. INTRODUCTION Graphical models (GM) are a powerful method
            representing and manipulating joint probability distributions. …
                                  …
</paper>
```

Anexo 6. Comparación de la calidad del agrupamiento para el cálculo del umbral

Tabla A6.1 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
16	0.623	0.677	0.892

Tabla A6.2 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 A6.3 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**	14[a]	2[b]	0[c]
Media - Min	**.001**	**16[d]**	0[e]	0[f]
Max - Media	**.001**	**16[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 7. Resultados del experimento 1 para el trabajo con las referencias bibliográficas

Tabla A7.1 Valores de la medida Micro-Purity al aplicar el algoritmo K-Star para las funciones SimRefBib, Dice, Coseno y Jaccard.

C	SimRefBib	Dice	Coseno	Jaccard	C	SimRefBib	Dice	Coseno	Jaccard
1	0,9812	0,972	0,972	0,976	6	0,741	0,576	0,576	0,576
2	0,973	0,961	0,961	0,961	7	0,983	0,862	0,869	0,862
3	1	0,75	0,75	1	8	0,852	0,754	0,754	0,544
4	1	1	1	1	9	1	1	1	1
5	0,961	0,942	0,942	0,942	10	0,701	0,623	0,623	0,641

Tabla A7.2 Valores de la medida Macro-Purity al aplicar el algoritmo K-Star para las funciones SimRefBib, Dice, Coseno y Jaccard.

C	SimRefBib	Dice	Coseno	Jaccard	C	SimRefBib	Dice	Coseno	Jaccard
1	0,991	0,981	0,981	0,985	6	0,752	0,64	0,64	0,64
2	0,983	0,974	0,974	0,974	7	0,915	0,885	0,891	0,885
3	1	0,833	0,833	1	8	0,966	0,78	0,78	0,571
4	1	1	1	1	9	1	1	1	1
5	0,974	0,964	0,964	0,964	10	0,758	0,738	0,738	0,679

Tabla A7.3 Valores de la medida Overall F-Measure al aplicar el algoritmo K-Star para las funciones SimRefBib, Dice, Coseno y Jaccard

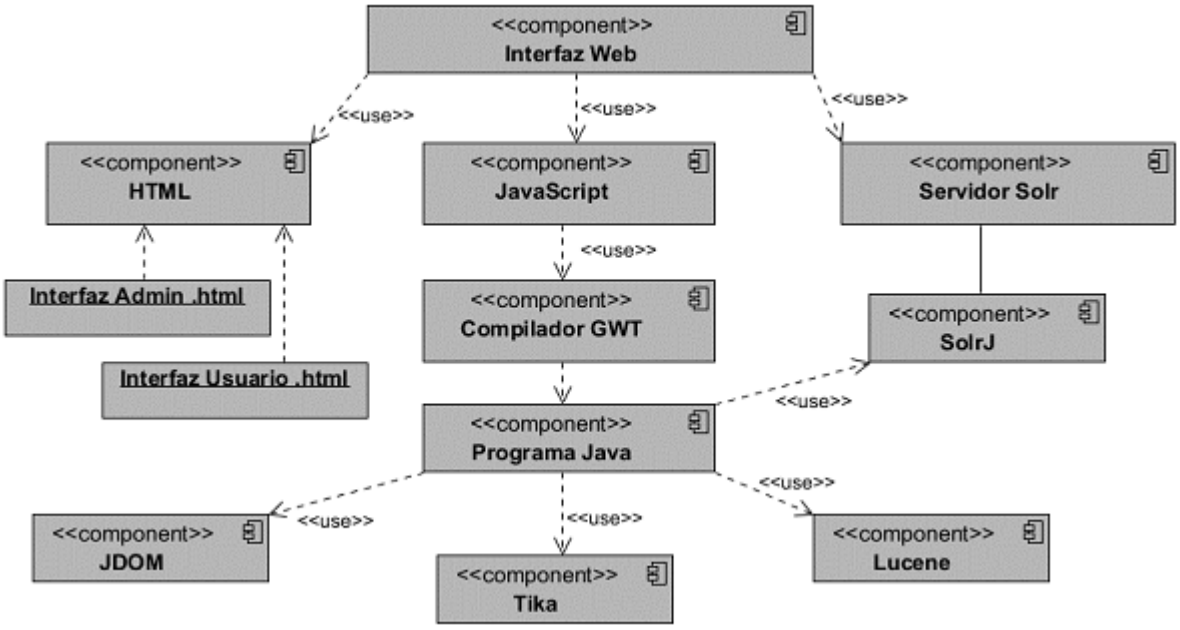
C	SimRefBib	Dice	Coseno	Jaccard	C	SimRefBib	Dice	Coseno	Jaccard
1	0,932	0,935	0,935	0,917	6	0,94	0,533	0,533	0,533
2	0,935	0,939	0,939	0,939	7	0,944	0,77	0,808	0,77
3	0,913	0,967	0,967	0,95	8	0,933	0,412	0,412	0,501
4	0,981	0,962	0,962	0,962	9	0,904	0,963	0,963	0,963
5	0,91	0,903	0,885	0,885	10	0,943	0,501	0,501	0,58

Anexo 8. Resultados del experimento 2 para el trabajo con las referencias bibliográficas

Tabla A8.1 Valores de la medida Overall F-Measure al aplicar el algoritmo INEXK-Star y el algoritmo SemClustDML

C	SemClustDML	K-Star
1	0,95	0,932
2	0,977	0,935
3	0,96	0,913
4	0,99	0,981
5	0,978	0,91
6	0,95	0,94
7	0,98	0,944
8	0,86	0,933
9	0,982	0,904
10	0,96	0,943

Anexo 9. Diagrama de componentes correspondiente al sistema ScientificSolR



Anexo 10. Resultados del experimento 1 para el trabajo con documentos no estructurados

Tabla A10.1 Valores de OFM para cada variante

C	K_StarGlobal	K_Star(2)	K_Star(3)	K_Star(4)	K_Star(5)
1	0,518	0,624	0,518	0,518	0,518
2	0,785	0,785	0,758	0,729	0,689
3	0,806	0,806	0,748	0,75	0,778
4	0,591	0,591	0,59	0,591	0,591
5	0,473	0,479	0,48	0,515	0,608
6	0,966	0,966	0,945	0,966	0,978
7	0,81	0,825	0,825	0,825	0,825
8	0,697	0,697	0,697	0,697	0,692
9	0,665	0,68	0,698	0,698	0,68
10	0,75	0,731	0,731	0,744	0,765
11	0,754	0,754	0,754	0,754	0,784
12	0,968	0,968	0,968	0,968	0,968
13	0,763	0,839	0,839	0,839	0,594
14	0,614	0,66	0,66	0,664	0,823
15	0,518	0,624	0,518	0,518	0,518
16	0,785	0,785	0,758	0,729	0,689

Tabla A10.2 Valores de Micro-Purity para cada variante

C	K_StarGlobal	K_Star(2)	K_Star(3)	K_Star(4)	K_Star(5)
1	0,667	0,75	0,667	0,667	0,667
2	0,8	0,8	0,84	0,818	0,752
3	0,8	0,8	0,556	0,727	0,667
4	0,464	0,464	0,718	0,464	0,464
5	0,603	0,467	0,552	0,532	0,578
6	1	1	0,957	1	0,75
7	0,916	0,92	0,92	0,92	0,92
8	0,792	0,792	0,792	0,792	0,611
9	0,754	0,696	0,683	0,683	0,696
10	0,604	0,633	0,633	0,633	0,571
11	0,639	0,639	0,639	0,559	0,659
12	1	1	1	1	1
13	0,938	1	1	1	0,835
14	0,773	0,756	0,756	0,897	0,933
15	1	1	1	1	1
16	0,81	1	1	1	1

Tabla A10.3 Valores de Macro-Purity para cada variante

C	K_StarGlobal	K_Star(2)	K_Star(3)	K_Star(4)	K_Star(5)
1	0,833	0,883	0,833	0,833	0,833
2	0,866	0,866	0,9	0,886	0,814
3	0,875	0,875	0,666	0,863	0,75
4	0,584	0,584	0,868	0,584	0,584
5	0,724	0,533	0,664	0,63	0,64
6	1	1	0,971	1	0,833
7	0,947	0,95	0,95	0,95	0,95
8	0,902	0,902	0,902	0,902	0,7
9	0,847	0,733	0,722	0,722	0,733
10	0,669	0,694	0,694	0,694	0,625
11	0,747	0,747	0,747	0,647	0,744
12	1	1	1	1	1
13	0,963	1	1	1	0,903
14	0,841	0,829	0,829	0,938	0,958
15	1	1	1	1	1
16	0,857	1	1	1	1

Anexo 11. Resultados del experimento 2 para el trabajo con documentos no estructurados

Tabla11.1 Mejores resultados para cada medida

C	K Star Global			K Star Top Max		
	OFM	**Micro-Purity**	**Macro-Purity**	**OFM**	**Micro-Purity**	**Macro-Purity**
1	0,518	0,667	0,833	0,624	0,75	0,883
2	0,785	0,8	0,866	0,785	0,84	0,9
3	0,806	0,8	0,875	0,806	0,8	0,875
4	0,591	0,464	0,584	0,591	0,718	0,868
5	0,473	0,603	0,724	0,608	0,578	0,664
6	0,966	1	1	0,978	1	1
7	0,81	0,916	0,947	0,825	0,92	0,95
8	0,697	0,792	0,902	0,697	0,792	0,902
9	0,665	0,754	0,847	0,698	0,696	0,733
10	0,75	0,604	0,669	0,765	0,633	0,694
11	0,754	0,639	0,747	0,784	0,659	0,747
12	0,968	1	1	0,968	1	1
13	0,763	0,938	0,963	0,839	1	1
14	0,614	0,773	0,841	0,823	0,933	0,958
15	0,853	1	1	0,853	1	1
16	0,792	0,81	0,857	0,944	1	1