UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA

"Clasificación Automática de Resúmenes de Tesis basada en Algoritmos de Agrupamiento Jerárquicos"

T E S I S

que para obtener el título de

Ingeniero en Computación

presenta

Adriana Gabriela Ramírez de la Rosa

Directora de tesis:

M. en C. María Auxilio Medina Nieto

Huajuapan de León, Oaxaca, a 1 de febrero de 2008

Para los que siempre estuvieron, están y estarán:
Mamá, Papá, Yubia, Brissa y Omar... mi familia
Gracias.

Resumen

Esta tesis presenta un análisis de los resultados de pruebas realizadas a diferentes algoritmos de agrupamiento jerárquicos donde se hace uso de una colección de 20 tesis digitales de la UTM, escritas utilizando el formato propuesto en la Iniciativa de Archivos Abiertos (OAI), *Dublin Core*. Se describe la implementación del sistema de clasificación, exploración y búsqueda de tesis digitales denominado Luna, el cual utiliza la herramienta CLUTO[1] para clasificar las tesis digitales de la UTM.

Entre las características que presenta el sistema desarrollado, resalta la exploración de la colección generada para encontrar de forma rápida y sencilla documentos potencialmente relevantes. Utiliza la herramienta Hermes [MNSBY03] para recuperar tesis de la colección e incorpora un método para la futura evaluación de la precisión de la recuperación.

Palabras clave: Algoritmos de agrupamiento jerárquicos, clasificación automática de documentos, recuperación de información, CLUTO.

[1] http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download

Índice

Capítulo 1

Introducción

En instituciones educativas, la difusión de los documentos de tesis es fundamental para evitar duplicidad de esfuerzos y para fomentar el seguimiento a los trabajos de investigación. Compartir estos documentos entre diversas universidades u organizaciones similares incrementa estos beneficios. La Universidad Tecnológica de la Mixteca (UTM[1]) cuenta con un conjunto de tesis digitalizadas al cual se le denominará *colección*.

Actualmente la clasificación de las tesis en la UTM se realiza por el personal que labora en la biblioteca, quienes emplean el tesauro de la biblioteca del Congreso de Washington. El proceso de clasificación requiere que una persona lea la tesis (regularmente sólo se lee el resumen y/o la introducción), posteriormente esta persona la asigna a una clase apoyado en el tesauro. Una de las ventajas de este proceso es que en él interviene la experiencia de la persona que clasifica. Si no existe mucho material por clasificar, la persona asignada puede dedicar más tiempo a la lectura de la tesis y de esta forma tener una idea más general de la clase a la cual debe pertenecer; sin embargo, esto conduce a tener una ubicación diferente, dependiendo de la persona que realice tal actividad y del tiempo que dedique a esta tarea. En general, la clasificación resultante de este proceso se limita a la referencia del tesauro mencionado. Al utilizar un sistema que realice la clasificación de tesis automáticamente, estos problemas quedarían resueltos, pues siempre que se clasifique una colección con los mismos parámetros, se obtendrán grupos iguales, y la clasificación no se basaría en otro documento, sino en el contenido de los documentos mismos.

En el ámbito de bibliotecas digitales, la Iniciativa de Archivos Abiertos (OAI) propone un protocolo de interoperabilidad de bajo nivel que puede utilizarse para hacer factible el intercambio de documentos. El protocolo se conoce como OAI-PMH [VdSL07], de las siglas en inglés de *Open Archives Initiative Protocol for Metadata Harvesting* [VdSL02]. La Iniciativa define dos componentes principales: proveedores de datos (organismos que comparten colecciones de información) y proveedores de servicios (sistemas que emplean los datos para ofrecer servicios como recuperación y clasificación de documentos).

Actualmente, la UTM está en vías de ser un proveedor de datos; pondrá a

[1] http://www.utm.mx/

disposición una colección de resúmenes que describirán algunos de los trabajos de tesis de esta institución. El tamaño de esta colección aumentará gradualmente, de tal manera que sin el empleo de mecanismos computacionales, será difícil encontrar de forma precisa, rápida y sencilla, documentos que respondan a los intereses de los usuarios.

Por ello, la justificación de esta tesis es la implementación de un sistema de clasificación automática de documentos que permita a los usuarios realizar las siguientes tareas: a) recuperar tesis relevantes y b) explorar la colección.

La clasificación se basa en algoritmos jerárquicos de agrupamiento, los cuales producen árboles de grupos de documentos que sirven para explorar y visualizar de forma rápida y ordenada una colección de documentos [ZK05].

1.1. Hipótesis

Es posible encontrar tesis relevantes a través de mecanismos de recuperación de información o al explorar la organización de los documentos en la colección de tesis digitales de la UTM.

1.2. Objetivo general

Implementar un mecanismo automático de clasificación y exploración para consultar la colección de tesis digitales de la UTM con base en el análisis de resúmenes.

1.3. Objetivos específicos

1. Comparar las características de los algoritmos jerárquicos aglomerados y de partición para agrupar documentos.

2. Evaluar la eficiencia de estos algoritmos para clasificar una colección de tesis mediante el empleo del software CLUTO.

3. Proponer un mecanismo para explorar la jerarquía generada por el algoritmo de mayor precisión aplicado a la colección de tesis digitales de la UTM.

4. Buscar tesis por palabras clave, título, autor, carrera y asesor.

5. Recuperar tesis por contenido mediante el modelo de espacios vectoriales.

6. Construir una base de datos que permita evaluar la precisión del sistema de recuperación por contenido a través del almacenamiento de las consultas y respuestas.

1.4. Alcances y limitaciones

1. Se asume que existe una colección de tesis de la UTM con al menos veinte resúmenes.

2. La exploración de la jerarquía se limitará a la región establecida dentro de la ventana cuando el número de documentos exceda el límite definido por el administrador del sistema.

3. La implementación del modelo de espacios vectoriales no forma parte del trabajo de tesis, pero si su incorporación.

4. El almacenamiento de preguntas y respuestas se implementará como un módulo independiente del diseño del resto del sistema. En esta tesis, una respuesta corresponde al conjunto de tesis que el sistema propone como relevantes. Es el resultado a una consulta.

1.5. Organización de la tesis

El Capítulo 2 presenta el marco teórico. El Capítulo 3 describe el análisis de algunos algoritmos de agrupamiento. El diseño del sistema de clasificación, exploración y búsqueda se presenta en el Capítulo 4, mientras que su implementación se explica en el Capítulo 5. Por último, el Capítulo 6 contiene las conclusiones y el trabajo a futuro.

Capítulo 2

Marco Teórico

El agrupamiento es el proceso de formar automáticamente grupos de objetos de manera que objetos de un mismo grupo tienen una similitud alta y objetos de diferentes grupos una similitud baja. A diferencia de la clasificación, en el agrupamiento se desconoce el nombre o número de los grupos. El objetivo del agrupamiento es encontrar una estructura de objetos que no están etiquetados [LLB03]. Esta sección describe tipos comunes de algoritmos de agrupamiento que pueden aplicarse para apoyar la administración de colecciones de documentos.

2.1. Algoritmos de agrupamiento

Los algoritmos de agrupamiento proveen un mecanismo para organizar y visualizar grandes cantidades de información en pequeños grupos que contienen similitud en su interpretación o contenido. En particular, las soluciones de agrupamiento jerárquico ofrecen una vista de los documentos a diferentes niveles de granularidad, que las hace ideales para visualizar y explorar grandes colecciones de documentos [LLB03].

Además de las Bibliotecas Digitales, los algoritmos de agrupamiento se aplican en otras áreas como las siguientes [Zai06]:

- Reconocimiento de patrones
- Análisis de datos espaciales
- Procesamiento de imágenes
- Ciencias económicas, especialmente mercadotecnia
- Clasificación automática de documentos
- Aplicaciones web

Exiten varios tipos de algoritmos de agrupamiento. La Figura 2.1 muestra algunas de las características principales que los definen.

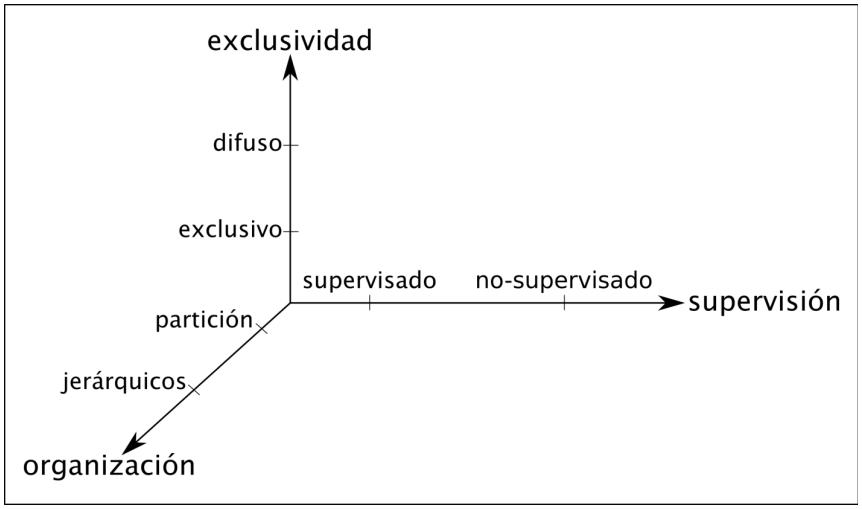


Figura 2.1. Algunas características de algoritmos de agrupamiento.

De la Figura 2.1, la supervisión involucra la participación humana durante la construcción de los grupos. Los algoritmos capaces de descubrir automáticamente similitudes o asociaciones en los objetos son no supervisados. La exclusividad se relaciona con la membresía de un objeto a un grupo. Cuando los objetos pertenecen a un sólo grupo, el algoritmo es exclusivo. En caso contrario, se denomina difuso. La organización se refiere a la estructura que el algoritmo construye para explorar los objetos. Por ejemplo, los algoritmos de partición producen estructuras planas de grupos, mientras que los algoritmos jerárquicos producen árboles.

Existen otras características muy importantes para seleccionar el algoritmo apropiado como la dimensionalidad, la precisión y la cobertura, o la velocidad del agrupamiento. En esta tesis el interés se centra en los algoritmos jerárquicos porque éstos forman grupos o *clusters* organizados en forma de árbol que facilita la recuperación de los documentos y permite la exploración de una colección. Implícitamente, las medidas utilizadas en la tesis calculan la precisión y la cobertura. No se considera la velocidad ni la dimensionalidad de acuerdo al criterio propuesto en [SKK00], porque el dominio de aplicación es menor a 5000 documentos.

En la literatura, el análisis de los algoritmos jerárquicos de agrupamiento emplea la terminología de los árboles en estructuras de datos, o de herencia en la programación orientada a objetos, de forma que se emplean conceptos como nivel, nodo padre o superclase, nodo hijo o subclase, entre otros.

De acuerdo al enfoque para construir el árbol de grupos, [MS03] divide a los algoritmos de agrupamiento jerárquicos en *aglomerados* y *divisibles*.

Los algoritmos de agrupamiento jerárquicos aglomerados, llamados también algoritmos ascendentes (*bottom-up*), inician creando un grupo por cada elemento. En los pasos subsecuentes se combinan los grupos más similares en un mismo grupo,

Función	Definición
Enlace Simple	Similitud de los dos miembros más similares
Enlace Completo	Similitud de los dos miembros menos similares
Promedio del grupo	Promedio de similitud entre miembros

Tabla 2.1. Funciones de Similitud.

hasta llegar a agrupar en la parte más alta a todos los elementos dentro de un solo grupo.

Los algoritmos de agrupamiento jerárquicos divisibles o descendentes (*top-down*), inicialmente agregan todos los elementos a un grupo. En cada iteracción, determinan qué grupo es el menos coherente y lo divide. El objetivo es que los grupos con objetos similares tengan mayor cohesión que los grupos con objetos no similares. En el agrupamiento, la cohesión es una medida que sirve como un umbral para indicar qué tan similares son los objetos de un grupo.

Tanto para los algoritmos aglomerados como para los divisibles, se debe determinar la similitud entre los objetos y los grupos. Para ello se utilizan funciones de similitud. Algunas de las más comunes se definen en la Tabla 2.1 [MS03, Gle01].

Las imágenes que aparecen en la Figura 2.2 muestran dos grupos A (izquierda) y B (derecha); en la figura del inciso a) se muestra un ejemplo del enlace simple entre los grupos A y B, pues están ligados o enlazados a través de sus miembros (puntos) más cercanos; mientras que el inciso b) muestra un ejemplo del enlace completo donde están enlazados por sus miembros más alejados.

2.2. Funciones de criterio para agrupamiento

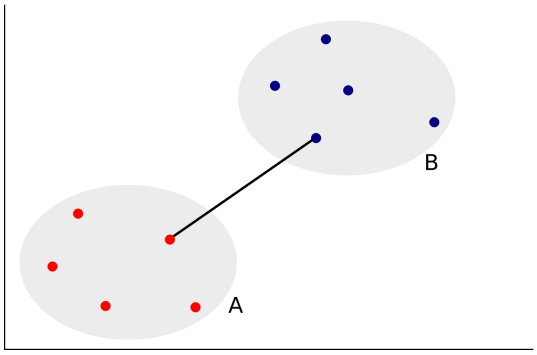
Para esta sección se describirán las funciones de criterio que usan los algoritmos de agrupamiento que se estudian. Para ello se necesitan definir dos vectores, *compuesto* (D_A) y *centroide* (C_A) como sigue:

$$D_A = \sum_{d \in A} d \tag{2.1}$$

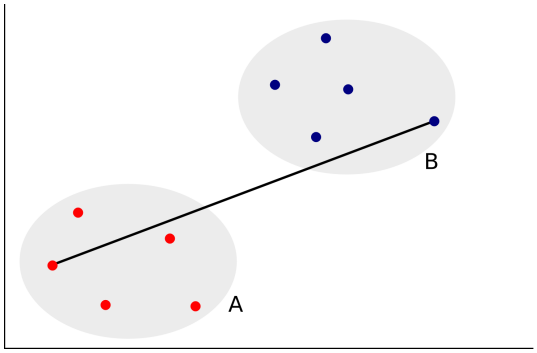
$$C_A = \frac{D_A}{|A|} \tag{2.2}$$

donde A es un conjunto de documentos y d es un documento de este conjunto.

En las secciones posteriores se usará la simbología siguiente: n denota el número de documentos, m el número de términos, y k el número de grupo. S se usará para denotar un conjunto de n documentos que se desean agrupar, así como $S_1, S_2, ..., S_k$ denotan cada uno de los k grupos y $n_1, n_2, ..., n_k$ denotan el tamaño de los correspondientes grupos.



(a) Enlace simple



(b) Enlace completo

Figura 2.2. Ejemplo de las funciones de similitud.

2.2.1. Funciones de criterio internas

Este tipo de funciones se enfocan en producir una solución de agrupamiento que optimiza una función de criterio particular definida sobre los documentos que son parte de cada grupo y que no forman parte de los documentos asignados a grupos diferentes.

La Ecuación 2.3 es el máximo de la suma del promedio de similitud entre los documentos asignados a cada grupo. En una variante del algoritmo de agrupamiento *K-means* (denominado también K-partes) [MS03], cada grupo se representa por un vector centroide, (el promedio de los valores que definen a los documentos dentro de la colección [Gle01]), por lo que en la Ecuación 2.4, el objetivo es encontrar una solución de agrupamiento que maximice la similitud entre cada documento y el centroide de cada grupo al que es asignado. La siguiente función de criterio, expresada en la Ecuación 2.5, es usada en el algoritmo *K-means* tradicional, éste usa la distancia euclidiana para determinar cuáles documentos deberían agruparse [ZK01].

$$\text{máx}\, I_1 = \sum_{r=1}^{k} n_r (\frac{1}{n_r^2} \sum_{d_i, d_j \epsilon S_r} \cos(d_i, d_j)) \tag{2.3}$$

$$\text{máx}\, I_2 = \sum_{r=1}^{k} \sum_{d_i \epsilon d_j} \cos(d_i, C_r) \tag{2.4}$$

$$\text{máx}\, I_3 = \sum_{r=1}^{k} \sum_{d_i \epsilon d_j} \|d_i - C_r\|^2 \tag{2.5}$$

2.2.2. Funciones de criterio externas

Esta clase de funciones se centra en la elaboración de una solución de agrupamiento que optimiza una función particular, la cual se define por la identificación de los documentos que forman parte de cada grupo, y no toma en cuenta los documentos asignados a otros grupos. Las funciones que se mencionan tratan de separar los documentos de cada grupo de la colección completa.

La Ecuación 2.6 trata de minimizar el coseno entre el vector centroide de cada grupo y el vector centroide de la colección entera. Actualmente esta función es una función de criterio híbrida que combina características de grupos externas e internas. La Ecuación 2.7 se define en función de la distancia euclidiana y el cuadrado de los errores del vector centroide.

$$\text{mín}\, E_1 = \sum_{r=1}^{k} n_r \cos(Cr, C) \tag{2.6}$$

$$\text{máx}\, E_2 = \sum_{r=1}^{k} n_r ||C_r - C||^2 \tag{2.7}$$

2.2.3. Funciones de criterio híbridas

Las funciones de criterio internas tratan de maximizar varias medidas de similitud sobre los documentos dentro de cada grupo; las funciones de criterio externas tratan de minimizar la similitud entre los documentos de los grupos y la colección. Sin embargo, varias funciones de criterio pueden combinarse para definir un conjunto de funciones híbridas que al mismo tiempo optimicen múltiples funciones de criterio individuales.

La función de criterio H_1 (Ecuación 2.8), combina las funciones de criterio interna I_1 y externa E_1, mientras que la función de criterio H_2 (Ecuación 2.9, combina las funciones interna I_2 y externa E_1 [ZK01].

$$\text{máx}\, H_1 = \frac{I_1}{E_1} \tag{2.8}$$

$$\text{máx}\, H_2 = \frac{I_2}{E_1} \tag{2.9}$$

2.2.4. Funciones de criterio basadas en grafos

Hasta el momento, las funciones de criterio tratan los documentos de la colección como un vector multidimensional. Una alternativa es describir los documentos usando grafos. En la función de criterio definida en la Ecuación 2.10, se usa un grafo que se obtiene con el cómputo de los pares similares entre los documentos; mientras que la función de criterio definida en la Ecuación 2.11, usa un grafo que se obtiene tratando al documento y a los términos como un grafo bipartito.

$$\text{mín}\, G_1 = \sum_{r=1}^{k} \frac{D_r^t D}{||D_r||^2} \tag{2.10}$$

$$\text{mín}\, G_2 = \sum_{r=1}^{k} \frac{cut(V_r, V - V_r)}{W(V_r)} \tag{2.11}$$

De la Ecuación 2.11, V_r es el conjunto de vértices asignado al r-ésimo grupo y $W(V_r)$ es la suma de los pesos de las listas adyacentes de los vértices asignados al r-ésimo grupo [ZK01].

2.3. Medidas para determinar la calidad del agrupamiento

En esta sección se describen las medidas que son utilizadas para determinar qué clasificación es mejor en relación a otra. En los siguientes capítulos se usan estas medidas para comparar los algoritmos de agrupamiento evaluados en esta tesis.

De acuerdo con [SKK00], existen dos tipos de medidas de calidad: *internas* y *externas*. Las medidas internas comparan diferentes conjuntos de grupos sin referencias a conocimiento externo. Las medidas externas evalúan el desempeño del agrupamiento al comparar los grupos producidos por técnicas conocidas.

En esta tesis se utilizarán las medidas externas pues es necesario comparar los grupos producidos, haciendo referencia a conocimiento externo, con la finalidad de determinar la calidad de estos grupos.

Dentro de las medidas externas, se pueden listar las siguientes: entropía, cobertura, precisión, medida F, medida E y medidas orientadas al usuario.

2.3.1. Entropía

La *entropía* (E_j) mide la calidad de los grupos que una consulta arroja. Sea CS un conjunto solución. Para cada grupo, se calcula la distribución de clase de cada dato. Por ejemplo, para el grupo j, se procesa p_{ij}, es decir, la probabilidad de que un miembro del grupo i pertenezca al grupo j. Usando esta distribución de clase, la entropía de cada grupo j se calcula como:

$$E_j = -\sum_i p_{ij} \log(p_{ij}) \tag{2.12}$$

donde log es el logaritmo de base 10 de p_{ij} y la suma se hace sobre todas las clases.

La *entropía total* (E_{CS}) de un conjunto de grupos se calcula como la suma de las entropías de cada grupo por el tamaño del grupo:

$$E_{CS} = \sum_{j=1}^{m} \frac{n_j * E_j}{n} \tag{2.13}$$

donde n_j es el tamaño del grupo j, m es el número de grupos y n es el número total de datos [SKK00]. En general, entre más bajo sea el valor de la entropía, mejor es la solución de agrupamiento [ZK05].

2.3.2. Cobertura y precisión

[BYRN99] define estas medidas como sigue:

Cobertura (*recall*) es la fracción de los documentos relevantes que se recuperaron.

$$Cobertura = \frac{|Ra|}{|R|} \tag{2.14}$$

Precisión es la fracción de los documentos recuperados que son relevantes.

$$Precisión = \frac{|Ra|}{|A|} \tag{2.15}$$

Ra, R y A se muestran en el diagrama de la Figura 2.3; donde mediante un algoritmo de recuperación de información se accede a la colección y se obtiene el conjunto respuesta A (documentos recuperados), mientras que el conjunto R (documentos relevantes) es la porción de documentos dentro de la colección que son relevantes independiente de los algoritmos de recuperación de información empleados; por lo tanto, la intersección de estos dos conjuntos da como resultado el conjunto Ra, los documentos relevantes recuperados por algún algoritmo.

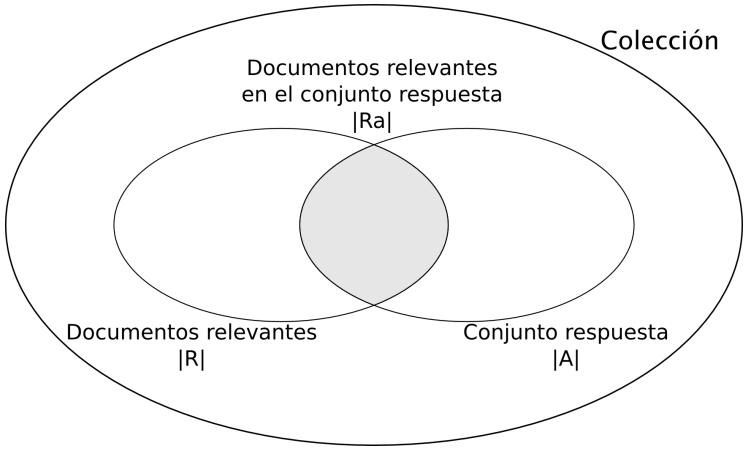


Figura 2.3. Definición de cobertura y precisión [BYRN99].

2.3.3. Medida F

La *medida F* combina la cobertura y la precisión [BYRN99, SKK00] como se muestra en la fórmula siguiente.

$$F(j) = \frac{2}{\frac{1}{r(j)} + \frac{1}{P(j)}} \tag{2.16}$$

donde $r(j)$ es la medida cobertura del j-ésimo documento de la lista recuperada, $P(j)$ es la medida de precisión para el j-ésimo documento en la lista recuperada, y $F(j)$ es la medida F de $r(j)$ y $P(j)$ relativo al j-ésimo documento de la lista recuperada.

La *medida F* asume valores en el intervalo [0, 1]. Es 0 si no se recupera ningún documento relevante y 1 si todos los documentos recuperados son relevantes [BYRN99].

2.3.4. Medida E

Al igual que la medida F, la *medida E* combina las medidas cobertura y precisión. Su objetivo es permitir al usuario especificar en cuál de estas medidas está más interesado. Se define como sigue:

$$E(j) = 1 - \frac{1 + b^2}{\frac{b^2}{r(j)} + \frac{1}{P(j)}} \tag{2.17}$$

donde $r(j)$ es la medida cobertura del *j-ésimo* documento de la lista recuperada, $P(j)$ es la medida de precisión para el j-ésimo documento en la misma lista, $E(j)$ es la *medida E* relativa a $r(j)$ y $P(j)$, y b es un parámetro especificado por el usuario, el cual refleja su interés relativo en la cobertura o precisión.

Para $b = 1$, la *medida E(j)* funciona como el complemento de la *medida F*; valores más grandes que 1 indican que el usuario está más interesado en la precisión que en la cobertura, mientras que valores menores a 1, indican que el usuario está más interesado en cobertura que en la precisión [BYRN99].

2.3.5. Medidas orientadas al usuario

Las medidas cobertura y precisión asumen que el conjunto de documentos relevantes para una consulta es el mismo, independientemente del usuario. Sin embargo, cada usuario puede tener su propia opinión. Las medidas orientadas al usuario, tales como *cociente de cobertura*, *cociente de novedad*, *cobertura relativa*, *esfuerzo de cobertura* tienen como objetivo resolver este problema [BYRN99].

Cociente de cobertura

El *cociente de cobertura* (*coverage ratio*) se define como la fracción de los documentos relevantes conocidos (por el usuario) (Rk), sobre los documentos recuperados (U).

$$cociente\_de\_cobertura = \frac{|Rk|}{|U|} \tag{2.18}$$

Cociente de novedad

El *cociente de novedad* (*novelty ratio*) calcula los documentos relevantes recuperados desconocidos por el usuario. Se calcula mediante la fórmula siguiente:

$$cociente\_de\_novedad = \frac{|Ru|}{|Ru| + |Rk|} \tag{2.19}$$

donde Ru son los documentos relevantes que se desconocían, los cuáles fueron recuperados; Rk los documentos relevantes recuperados esperados por el usuario; y U los documentos relevantes conocidos por el usuario. Ver el diagrama de la Figura 2.4.

Un *cociente de cobertura* alto, indica que el sistema encuentra más de los documentos relevantes que el usuario espera ver, es decir, el sistema encuentra documentos relevantes desconocidos por el usuario. Un *cociente de novedad* alto, indica que el sistema muestra muchos documentos relevantes nuevos los cuales eran desconocidos.

La *cobertura relativa* se calcula como el cociente entre el número de documentos relevantes encontrados (por el sistema) y el número de documentos relevantes que el usuario espera encontrar. Por otro lado, el *esfuerzo de cobertura* divide el número de documentos relevantes que el usuario espera encontrar entre el número de documentos examinados, en un intento por encontrar los documentos relevantes esperados.

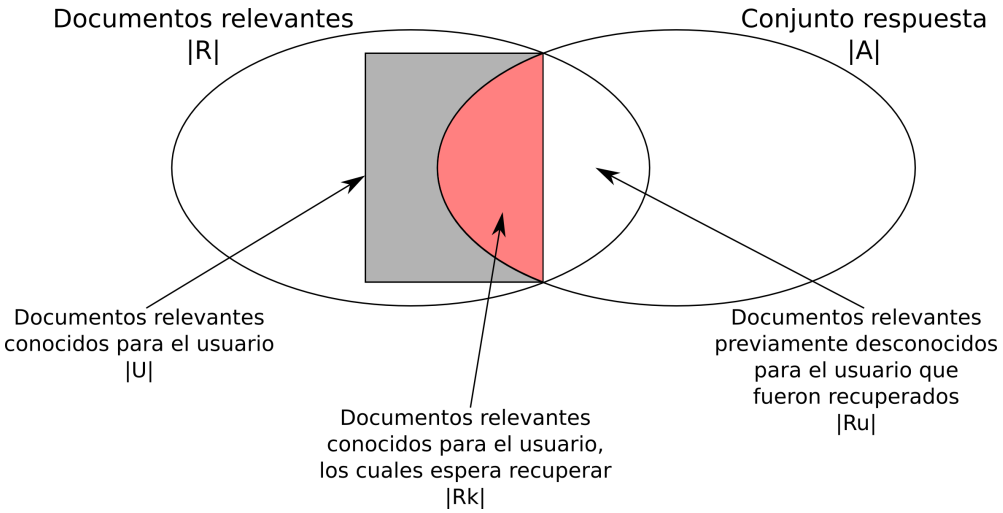


Figura 2.4. Definición de cociente de cobertura y novedad [BYRN99].

2.4. Modelo de espacios vectoriales

Los algoritmos de agrupamiento que se analizan en esta tesis utilizan el modelo de espacios vectoriales para representar cada documento a agrupar.

La idea principal de este modelo es crear una matriz de términos y documentos M, en donde las filas representan los documentos y las columnas los términos que éstos contienen.

El valor que contiene cada celda de la matriz M_{ij} es un valor que determina qué tanto el término i sirve para describir al documento j.

[ZK01] propone dos maneras para calcular este valor, las cuales se presentan a continuación:

- *Frecuencia del Término* (tf) en donde cada documento se representa con un vector de la forma
$$d_{tf} = (tf_1, tf_2, ..., tf_m) \tag{2.20}$$
donde tf_i es la frecuencia del i-ésimo término en el documento d_{tf}.

- *Frecuencia Inversa del Documento* (idf) en donde cada documento se representa con un vector de la forma
$$d_{tfidf} = (tf_1 \log(N/df_1), tf_2 \log(N/df_2), ..., tf_m \log(N/df_m)) \tag{2.21}$$
donde m es el número de términos distintos en la colección de documentos, tf_i es la frecuencia del i-ésimo término en el documento, df_i es el número de documentos que contienen al i-ésimo término, y N es el número total de documentos en la colección.

Este modelo también es utilizado para realizar la recuperación de información. La herramienta Hermes, que es un componente de bibliotecas digitales que permite recuperar información relevante mediante el uso de diferentes modelos de recuperación de información [MN02], implementa este modelo.

Capítulo 3

Análisis de algoritmos de agrupamiento

Este capítulo describe la realización de las pruebas a los algoritmos de agrupamiento jerárquicos que ofrece DocCluster y CLUTO, y presenta los resultados obtenidos.

Se utiliza el software CLUTO ya que permite evaluar 3 diferentes clases de algoritmos de agrupamiento, a saber, de partición, aglomerativos y basados en grafos particionados. También se usa DocCluster [FWE03], pues es parte de la investigación previa a esta tesis. El software seleccionado ha sido ampliamente utilizado en investigaciones previas relacionadas con algoritmos de agrupamientos, se consideran plataformas de prueba y referencia [SKK00].

El objetivo de estas pruebas es determinar qué algoritmo jerárquico genera mejores resultados para la colección de prueba y con qué parámetros. Una vez identificado el algoritmo con mejores resultados, se utiliza la implementación existente en el desarrollo del sistema propuesto en esta tesis.

3.1. Herramientas de agrupamiento

3.1.1. DocCluster

DocCluster es una herramienta para clasificar documentos que implementa el algoritmo de agrupamiento *Frequent Itemset-based Hierarchical Clustering* (FIHC), el cual se basa en la identificación de grupos de términos frecuentes. La premisa de este algoritmo es que existen algunos términos frecuentes en cada grupo del conjunto de documentos, y que diferentes grupos comparten pocos términos frecuentes [FWE03].

Algunas características del algoritmo FIHC son:

- Reduce el número de dimensiones[1].
- La precisión del agrupamiento es alta.
- El número de grupos es un parámetro de entrada opcional.

[1]Se entiende por dimensiones al número de características asociadas a los datos a clasificar

- Facilita la exploración de la colección, ya que cada grupo está descrito por grupos de términos frecuentes.

3.1.2. CLUTO

CLUTO es una familia de programas y librerías de análisis de grupos computacionalmente eficientes, adecuados para conjuntos de datos de baja y alta dimensión. El paquete de esta familia que se usa en esta tesis se denomina CLUTO; su función es agrupar conjuntos de datos de baja y alta dimensión, así como analizar las características de diferentes grupos [Kar06].

CLUTO consta de dos programas *stand-alone* (vcluster y scluster) y una biblioteca a través de la cual una aplicación puede acceder directamente a las diferentes agrupaciones y análisis de algoritmos implementados.

Algunas características de CLUTO son:

- Contiene múltiples funciones de similitud: distancia euclidiana, coseno, coeficiente de correlación, Jaccard, y definidos por el usuario.
- Implementa el esquema de fusión aglomerativo tradicional: enlace simple (*single-link*), enlace completo (*complete-link*) y *Unweighted Pair Group Method with Arithmetic mean* (UPGMA).
- Emplea diversos formatos de salida para las visualizaciones de los grupos: postscript, SVG, gif, xfig, entre otros.
- Soporta múltiples métodos para resumir eficazmente los grupos: más descriptivos, cliqués (pequeños grupos exclusivos trabajando en forma conjunta con un interés común), y términos frecuentes (*frequent itemsets*).
- Puede escalar conjuntos de datos muy grandes que contienen cientos de miles de objetos y decenas de miles de dimensiones.

3.2. Descripción de la colección

Para formar la colección de tesis sobre las que se hacen las pruebas, se toman 20 tesis de la biblioteca de la UTM de 4 carreras diferentes. Al iniciar este trabajo, en la biblioteca existían 236 tesis digitales, de las cuales, aproximadamente sólo el 50 % de ellas cuentan con resúmen (parte esencial en el desarrollo de esta investigación). De las tesis con resúmen (118 tesis) se seleccionaron de forma aleatoria el 17 % (20 tesis), muestra que produce un error estándar máximo del 10 %, parámetro que se considera aceptable en esta tesis.

Al buscar las tesis con resúmenes dentro de la biblioteca digital, se desconocía la carrera a la que pertenecen, por lo que al final la elección de los 20 documentos se contabilizó el número de tesis por carrera. Estos datos se muestran en la Tabla 3.1,

Carrera	Número de tesis usadas	Identificador
Ingeniería en Computación	9	tC1 al tC9
Ingeniería en Alimentos	5	tA1 al tA5
Ingeniería en Electrónica	4	tE1 al tE4
Licenciatura en Ciencias Empresariales	2	tL1 al tL2

Tabla 3.1. Conjunto de prueba.

donde se usa un identificador para que posteriormente se haga referencia a los documentos dentro de la clasificación.

3.3. Modelo de datos

En esta tesis, el modelo de datos que se va a utilizar para representar los documentos forma parte del estándar que proporciona la Iniciativa de Archivos Abiertos (*Open Archives Initiative - OAI*) [LVdS01].

OAI es una iniciativa para desarrollar y promover estándares de interoperabilidad que faciliten la diseminación del contenido a través de Internet. Esta iniciativa propone el protocolo OAI-PMH (*OAI - Protocol for Metadata Harvesting*), el cual define un mecanismo para compartir registros que contengan metadatos de un repositorio.

En OAI, un recurso (*resource*) es cualquier objeto que tenga identidad, tal como un documento electrónico, una imagen, un servicio o la recopilación de otros recursos. Un registro (*record*) son varios metadatos estructurados acerca de un recurso, que representan una o más propiedades así como sus valores asociados.

Los metadatos se pueden representar en varios formatos. Sin embargo, el formato por omisión es Dublin Core (DC) [DCMI07a], cuyos elementos se muestran en la Tabla 3.2.

En la Tabla 3.3 se presentan los elementos de Dublin Core que se utilizarán para describir una tesis de la UTM, cabe mencionar que ninguno de estos elementos es obligatorio dentro de la iniciativa, sin embargo, algunos de ellos sí lo son para alcanzar los objetivos de esta tesis, pues apoyan las búsquedas y en especial, las búsquedas por contenido. En el Ejemplo 3.1 se muestra la descripción de una tesis de la UTM dentro de la colección utilizando el conjunto de elementos de Dublin Core (*Dublin Core Metadata Element Set, Version 1.1*) descrito en [DCMI07b].

Ejemplo 3.1.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF PUBLIC "-//DUBLIN CORE//DCMES DTD 2002/07/31//EN"
"http://dublincore.org/documents/2002/07/31/dcmes-xml/dcmes-xml-dtd.dtd">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

```
        xmlns:dc ="http://purl.org/dc/elements/1.1/">
   <rdf:Description rdf:about="http://biblioteca.utm.mx/">
      <dc:title>
         Clasificación Automática de Resúmenes de Tesis basada en Algo-
         ritmos de Agrupamiento Jerárquicos
      </dc:title>
      <dc:creator>Adriana Gabriela Ramírez de la Rosa</dc:creator>
      <dc:subject>
         Algoritmos  de agrupamiento jerárquicos, clasificación automá-
         tica de documentos, recuperación de información, CLUTO.
      </dc:subject>
      <dc:description>
         Esta tesis presenta un  análisis de los resultados  de pruebas
         realizadas a diferentes algoritmos de agrupamiento jerárquicos
         donde  se hace uso de una colección de  tesis digitales  de la
         UTM, escritas utilizando el formato propuesto en la Iniciativa
         de Archivos Abiertos (OAI), Dublin Core. Se describe la imple-
         menta del sistema de clasificación, exploración y búsqueda  de
         tesis digitales denominado Luna, el cual utiliza la herramien-
         ta CLUTO para clasificar las tesis digitales de la UTM.

         Entre las características que presenta el sistema desarrollado
         resalta la exploración de la colección generada para encontrar
         de forma rápida y sencilla documentos  potencialmente relevan-
         tes. Utiliza la  herramienta Hermes para recuperar  tesis rele
         vantes de la colección e  implementa un  método para la futura
         evaluación de la precisión de la recuperación.
      </dc:description>
      <dc:publisher>UTM</dc:publisher>
      <dc:contributor>M.C. María Auxilio Medina Nieto</dc:contributor>
      <dc:date>2008-02-01</dc:date>
      <dc:identifier>Ing en Computación</dc:identifier>
      <dc:language>es</dc:language>
      <dc:rights>UTM</dc:rights>
   </rdf:Description>
</rdf:RDF>
```

3.3.1. Proveedor de datos

Como trabajo previo a esta tesis, se realizó el sistema *dTesis*. Este sistema tiene como objetivo crear el proveedor de datos de la UTM, en donde los tesistas pueden registrar sus trabajos. Para construir tal proveedor dentro de la OAI, el sistema *dTesis* exige que el tesista introduzca los datos mostrados en la Tabla 3.3, con estos datos se construye el archivo XML correspondiente por cada tesis, como el del Ejemplo 3.1.

Elemento	Etiqueta de DC	Descripción
Título	dc:title	El nombre de un recurso, regularmente dado por el autor.
Autor o creador	dc:creator	La persona u organización responsable de la creación del contenido intelectual del recurso.
Palabras claves	dc:subject	Palabras clave o códigos de clasificación que describen el contenido del recurso.
Descripción	dc:description	Una descripción o resumen textual del recurso.
Editor	dc:publisher	La entidad responsable de hacer que el recurso se encuentre disponible.
Colaborador	dc:contributor	Una persona que haya tenido un contribución intelectual significativa en la creación del recurso.
Fecha	dc:date	La fecha asociada a un evento relacionado con el recurso.
Tipo del recurso	dc:type	La naturaleza o género del contenido del recurso.
Formato	dc:format	La manifestación física o digital del recurso.
Identificador del recurso	dc:identifier	Una referencia no ambigua para el recurso dentro de un contexto dado.
Fuente	dc:source	Una referencia a un recurso del cual se deriva el recurso tratado.
Lenguaje	dc:language	El lenguaje del contenido intelectual del recurso.
Relación	dc:relation	Una referencia a un recurso relacionado.
Cobertura	dc:coverage	La extensión o el alcance del contenido de un recurso. Incluye generalmente la localización geográfica, el periodo temporal o la jurisdicción.
Derechos	dc:rights	Información sobre los derechos del recurso.

Tabla 3.2. Elementos de Dublin Core.

Elemento	Uso dado	Obligatorio
Título	Título de la tesis	⋆
Autor o Creador	Autor de la tesis	⋆
Palabras Claves	Palabras claves que representan el contenido de la tesis	⋆
Descripción	Resumen del contenido de la tesis	⋆
Editor	Se establecerá como UTM	
Colaborador	Asesor de la tesis	⋆
Fecha	Fecha de presentación de la tesis	⋆
Identificador del recurso	Carrera del autor de la tesis	⋆
Lenguaje	Se establecerá como español (es)	
Derechos	Todos los derechos pertenecen a la UTM y a su autor, por lo que se establecerá como UTM	

Tabla 3.3. Elementos de Dublin Core utilizados en las tesis de la UTM.

n	
término 1	f_1
término 2	f_2
...	...
término n	f_n

Tabla 3.4. Formato de un *documento compacto*.

3.4. Preparación de los datos

De las 20 tesis que se eligieron para hacer las pruebas se toma el título y el resumen que contienen para formar los correspondientes 20 documentos de texto a los cuales se les llama *documentos originales*, posteriormente se procesan estos documentos para producir otros, a los que llamamos *documentos compactos*, que contienen $n+1$ números de renglones y 2 columnas a partir del segundo renglón tal como se muestra en la Tabla 3.4, donde n es el número de términos en el *documento original* sin repeticiones y f es la frecuencia del término i en el *documento original*, es decir, el número de veces que aparece el término i en el *documento original*.

Para verificar la representatividad del vector de términos usado, se procedió de esta forma:

1. Se tomó una muestra de 5 tesis de las 20 que se utilizan,

2. Se realizó un resumen de la tesis considerando todo su contenido (actividad realizada por estudiantes de la universidad),

3. Se calculó el porcentaje de aparición de los términos calculados previamente

Documento	% de ocurrencia
T1	68.18
T2	100.00
T3	60.71
T4	66.66
T5	89.58
Promedio	**77.03**

Tabla 3.5. Porcentajes de ocurrencia de términos

término 1_1	término 1_2	...	término 1_{f_1}
término 2_1	término 2_2	...	término 2_{f_2}
...	...	...	...
término n_1	término n_2	...	término n_{f_n}

Tabla 3.6. Formato de documentos de entrada en DocCluster.

con los que aparecen en el resumen.

Los resultados de las actividades anteriores se presentan en la Tabla 3.5, en donde se concluye que el porcentaje de ocurrencia de los términos de los vectores que representan a cada documento comparado con los términos que aparecen en el resumen realizado por los humanos es mayor al 77 %.

Una vez obtenidos los 20 *documentos compactos*, se procesan nuevamente para obtener documentos con el formato mostrado en la Tabla 3.6, en el cual cada término se repite f_i veces, donde f_i es la frecuencia de ese término en los *documentos compactos*; esto por los requerimientos de formatos de entrada de DocCluster.

Igual que para DocCluster, los datos de entrada para CLUTO es una matriz de tamaño n X m que se construye a partir de los *documentos compactos*, como el ejemplo que se muestra en la Tabla 3.7, donde n es el número de documentos de la colección y m es número de términos diferentes en toda la colección; en el ejemplo, $n = 4$ y $m = 6$. De aquí se forma el documento de entrada para CLUTO, cuyo formato se explica a detalle en [Kar03]. La Tabla 3.8 muestra un ejemplo de este formato basado en la matriz de la Tabla 3.7.

0.06	0.27		0.34		
	0.15				
0.56		0.17		0.20	
			0.13		0.51

Tabla 3.7. Matriz de valores de cada término por cada documento.

4	6 9
1	0.06 2 0.27 4 0.34
2	0.15
1	0.56 3 0.17 5 0.20
4	0.13 6 0.51

Tabla 3.8. Ejemplo del formato de entrada para CLUTO.

3.5. Pruebas con algoritmos de agrupamiento

En esta sección se analizan los algoritmos de agrupamiento que proporcionan las herramientas antes descritas, con la finalidad de determinar cuál obtiene mejores resultados de acuerdo a las medidas para determinar la calidad de un grupo.

3.5.1. Pruebas con DocCluster

La Tabla 3.9 muestra los parámetros de entrada del algoritmo (soporte global, soporte de grupo y número de grupos a formar) y si existen hijos (niveles) dentro de los grupos, el valor de las medidas F y entropía que se obtiene al ejecutar el algoritmo FIHC para los datos de prueba. De aquí, se puede concluir que los parámetros de entrada que arrojan una mejor medida F y mejor entropía son los sombreados en la Tabla 3.9.

En la Tabla 3.10 se muestran los resultados de las pruebas para cuando no se introduce el número de grupos dentro de los parámetros de entrada del algoritmo FIHC. Como puede verse, los mejores resultados se obtienen con los mismos parámetros que en la Tabla 3.9 cuando el número de grupos es 4.

La clasificación que se obtuvo con los mejores resultados en las pruebas se presenta en la Figura 3.1 y en la Figura 3.2, en ellas cada grupo está definido por la etiqueta que el algoritmo asigna, además de representar cada tesis con un cuadro con el identificador de la tesis a la que corresponde. Se usan los colores para agrupar las tesis que pertenecen a una misma carrera dentro de la clasificación actual de la UTM; ésto es útil para identificar rápidamente la ubicación de las tesis de prueba en los resultados de los algoritmos que se analizan.

3.5.2. Pruebas con CLUTO

Dentro de los parámetros más importantes que utiliza CLUTO, se usan los siguientes: *clmethod* (método de agrupamiento), *sim* (función de similitud) y *crfun* (función de criterio). Las pruebas fueron hechas con todas las combinaciones posibles de estos parámetros. En las Tablas 3.11, 3.12 y 3.13 se muestran sólo las pruebas con los resultados más sobresalientes, y en azul los renglones de las pruebas con los mejores resultados. Entre más pequeño sea el valor que le corresponde a la columna Entropía y mayor sea el valor correspondiente a la columna Pureza, mejor será el

Soporte global	Soporte de grupo	Medida F	Entropía (plana)	Número de grupos	Niveles
0.20	0.10	0.227451	2.356700	3	0
0.20	0.20	0.271569	2.169667	3	1
0.20	0.30	0.271569	2.169667	3	1
0.25	0.10	0.235417	2.265144	3	0
0.25	0.20	0.240000	2.193914	3	0
0.25	0.30	0.235417	2.265144	3	0
0.20	0.10	0.277083	2.169667	4	0
0.20	0.20	0.320833	1.985970	4	1
0.20	0.30	0.320833	1.985970	4	1
0.25	0.10	0.285000	2.081447	4	0
0.25	0.20	0.289524	2.013791	4	0
0.25	0.30	0.285000	2.081447	4	0
0.30	0.10	0.244444	2.408231	4	0
0.30	0.30	0.244444	2.408231	4	0
0.20	0.10	0.350641	1.794365	5	0
0.20	0.20	0.392949	1.622264	5	1
0.20	0.30	0.392949	1.622264	5	1
0.25	0.10	0.326667	1.985970	5	0
0.25	0.20	0.338974	1.837518	5	0
0.25	0.30	0.326667	1.985970	5	0

Tabla 3.9. Resultados de pruebas en FIHC con 4 y 5 grupos.

Soporte global	Soporte de grupo	Medida F	Entropía (plana)	Grupos creados	Hijos
0.20	0.10	0.277083	2.169667	4	0
0.20	0.20	0.320833	1.985970	4	1
0.20	0.30	0.320833	1.985970	4	1
0.25	0.10	0.235417	2.265144	3	0
0.25	0.20	0.289524	2.013791	4	0
0.25	0.30	0.235417	2.265144	3	0
0.30	0.10	0.244444	2.408231	4	0
0.30	0.30	0.244444	2.408231	4	0

Tabla 3.10. Resultados de pruebas en FIHC sin especificar número de grupos.

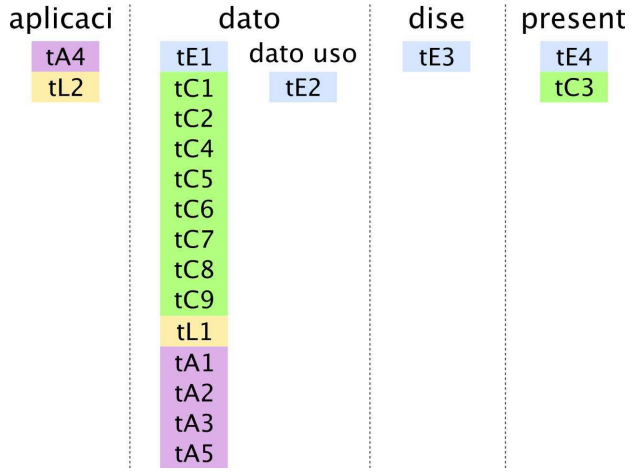


Figura 3.1. Resultado de FIHC para 4 grupos.

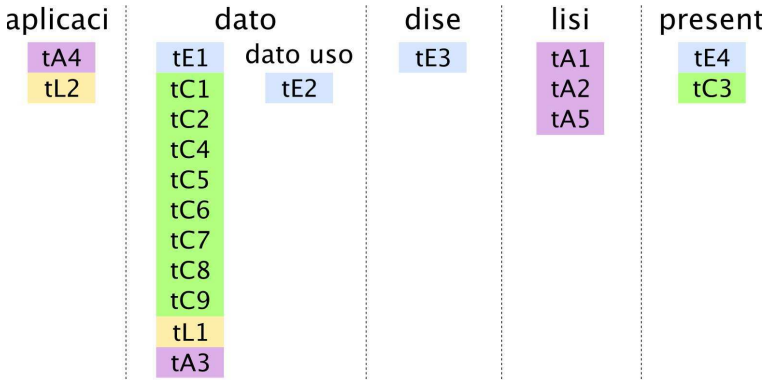


Figura 3.2. Resultado de FIHC para 5 grupos.

resultado que el algoritmo proporcione. La consideración de los mejores resultados se basa en los valores de estas medidas, las cuales son parte de la salida del algoritmo.

La Figura 3.3 muestra la clasificación de los documentos cuando se usa el método de clasificación *directo*, la función de similitud *coseno* y la función de criterio *H1* para 4 grupos. La Tabla 3.12 muestra que el valor de la entropía y pureza para esta combinación de parámetros es de 0.302 y 0.800, respectivamente. En la Figura 3.4 se muestra la clasificación de los documentos cuando se usa el método de clasificación *directo*, la función de similitud *coeficiente de correlación* y la función de criterio *E1* o *H2* con 4 grupos, donde los valores para la entropía y pureza son de 0.399 y 0.750, respectivamente. Usando el método de clasificación *directo*, la función de similitud *coeficiente de correlación* y la función de criterio *G1'*y con 4 grupos, se obtiene la clasificación de la Figura 3.5 donde los valores para la entropía y pureza son de

Método de Agrupamiento	**Función de similitud**	**Función de Criterio**	**Entropía**	**Pureza**
Bisecciones repetidas (optimizado)	Coseno	I1	0.502	0.650
Bisecciones repetidas (optimizado)	Coseno	E1	0.492	0.650
Bisecciones repetidas (optimizado)	Coseno	G1'	0.542	0.650
Bisecciones repetidas (optimizado)	Coseno	H1	0.492	0.650
Bisecciones repetidas (optimizado)	Coseno	H2	0.492	0.650
Bisecciones repetidas (optimizado)	Coeficiente de correlación	I1	0.430	0.650
Bisecciones repetidas (optimizado)	Coeficiente de correlación	I2	0.492	0.650
Bisecciones repetidas (optimizado)	Coeficiente de correlación	E1	0.402	0.700
Bisecciones repetidas (optimizado)	Coeficiente de correlación	G1	0.473	0.650
Bisecciones repetidas (optimizado)	Coeficiente de correlación	G1'	0.402	0.700
Bisecciones repetidas (optimizado)	Coeficiente de correlación	H1	0.492	0.650

Tabla 3.11. Resultados de pruebas en CLUTO con 4 grupos.

Método de Agrupamiento	Función de similitud	Función de Criterio	Entropía	Pureza
Bisecciones repetidas (optimizado)	Coeficiente de correlación	H2	0.402	0.700
Directo	Coseno	I1	0.516	0.600
Directo	Coseno	E1	0.492	0.650
Directo	Coseno	G1	0.527	0.650
Directo	Coseno	G1'	0.492	0.650
Directo	Coseno	H1	0.302	0.800
Directo	Coseno	H2	0.492	0.650
Directo	Coeficiente de correlación	I1	0.500	0.650
Directo	Coeficiente de correlación	I2	0.492	0.650
Directo	Coeficiente de correlación	E1	0.399	0.750
Directo	Coeficiente de correlación	G1	0.500	0.650
Directo	Coeficiente de correlación	G1'	0.352	0.800
Directo	Coeficiente de correlación	H1	0.492	0.650
Directo	Coeficiente de correlación	H2	0.399	0.750
Aglomerativo	Coseno	I1	0.471	0.650
Aglomerativo	Coseno	E1	0.459	0.650
Aglomerativo	Coseno	G1'	0.290	0.800
Aglomerativo	Coseno	H1	0.460	0.650
Aglomerativo	Coseno	H2	0.460	0.650
Aglomerativo	Coeficiente de correlación	I1	0.495	0.650
Aglomerativo (optimizado)	Coseno	I1	0.508	0.600
Aglomerativo (optimizado)	Coseno	I2	0.516	0.600
Aglomerativo (optimizado)	Coseno	E1	0.516	0.600

Tabla 3.12. Resultados de pruebas en CLUTO con 4 grupos (Continuación).

Método de Agrupamiento	**Función de similitud**	**Función de Criterio**	**Entropía**	**Pureza**
Aglomerativo (optimizado)	Coseno	G1	0.538	0.600
Aglomerativo (optimizado)	Coseno	G1'	0.516	0.600
Aglomerativo (optimizado)	Coseno	H1	0.516	0.600
Aglomerativo (optimizado)	Coseno	H2	0.516	0.600
Aglomerativo (optimizado)	Coseno	Ligadura simple	0.520	0.650
Aglomerativo (optimizado)	Coseno	Ligadura simple (grupos anchos)	0.520	0.650
Aglomerativo (optimizado)	Coseno	Ligadura completa	0.432	0.750
Aglomerativo (optimizado)	Coseno	Ligadura completa (grupos anchos)	0.432	0.750
Aglomerativo (optimizado)	Coseno	UPGMA	0.522	0.650
Aglomerativo (optimizado)	Coeficiente de correlación	I2	0.499	0.650
Aglomerativo (optimizado)	Coeficiente de correlación	G1	0.541	0.600
Aglomerativo (optimizado)	Coeficiente de correlación	H1	0.499	0.650
Aglomerativo (optimizado)	Coeficiente de correlación	H2	0.444	0.650

Tabla 3.13. Resultados de pruebas en CLUTO con 4 grupos (Continuación).

0.352 y 0.800, respectivamente. Mientras que la Figura 3.6 muestra la clasificación de los documentos uando se usa el método de clasificación *aglomerativo*, la función de similitud *coseno* y la función de criterio *G1'* con 4 grupos, y donde se obtuvo una entropía de 0.290 y un valor de pureza de 0.800; el valor de entropía más bajo y el valor de pureza más alto de los tres mejores resultados de la Tabla 3.12. Igual que para las figuras anteriores, se utiliza un código de colores para identificar las tesis que pertenecen a la misma carrera.

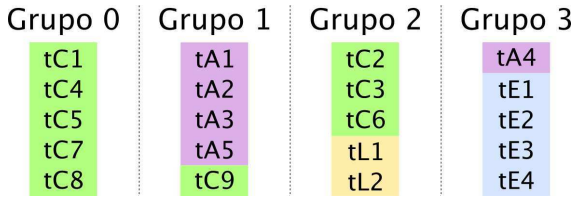


Figura 3.3. Resultado de CLUTO con los parámetros *directo*, *coseno*, *h1* y 4 grupos.

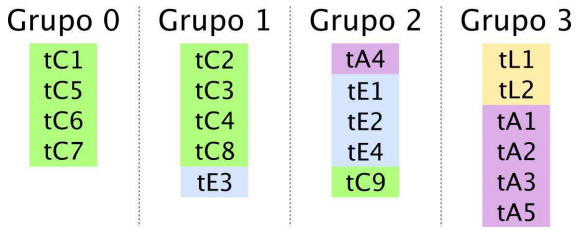


Figura 3.4. Resultado de CLUTO con los parámetros *directo*, *coeficiente de correlación*, *E1* y 4 grupos.

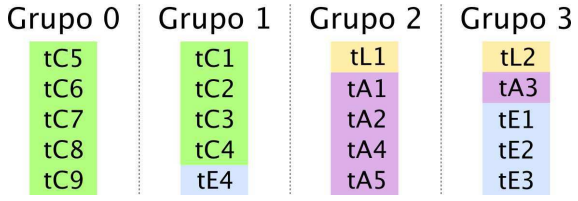


Figura 3.5. Resultado de CLUTO con los parámetros *directo*, *coeficiente de correlación*, *G1'* y 4 grupos.

Después de analizar los resultados de las pruebas, se concluye que el algoritmo que se utilizará en el desarrollo del sistema propuesto en esta tesis es el algoritmo *jerárquico aglomerativo* con parámetros: *coseno* como función de similitud, *G1'* como función de criterio y 4 grupos, ya que en las pruebas realizadas se obtuvo que es el número de grupos que arroja los mejores resultados. De forma intuitiva, este

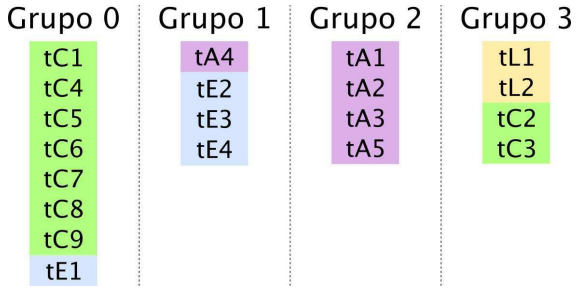


Figura 3.6. Resultado de CLUTO con los parámetros *aglomerativo*, *coseno*, *G1'* y 4 grupos.

resultado indica que los términos empleados en las tesis de cada carrera, se pueden utilizar para agrupar de forma automática la colección de tesis digitales de la UTM.

Capítulo 4

Diseño del sistema Luna

4.1. Esquema general

La Figura 4.1 muestra el entorno del sistema de clasificación, exploración y búsqueda de la colección de tesis digitales de la UTM (Luna). Esto es, como el sistema Luna deberá obtener las tesis del proveedor de datos de la UTM. El proceso se describe a continuación:

Cada tesista de la universidad deberá dar de alta su tesis en el sistema dTesis[1], que a su vez, procesará los documentos y los escribirá en formato XML tal como se describió en capítulos anteriores, para que con estos documentos se forme el proveedor de datos de la UTM.

Luna accede al proveedor de datos para obtener la colección existente y realizar las tareas que se describen en las siguientes secciones. Los usuarios en este entorno son las personas que buscarán información dentro de las tesis digitales de la universidad.

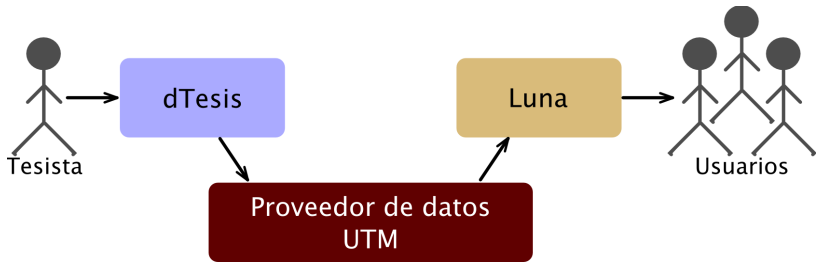


Figura 4.1. Entorno del sistema Luna.

[1]Sistema de tesis digitales, diseñado y desarrollado como trabajo previo a esta tesis

4.2. Análisis del sistema

4.2.1. Descripción general

El sistema de clasificación, exploración y búsqueda de la colección de tesis digitales de la UTM (Luna), hará uso de los diferentes algoritmos de agrupamiento proporcionados por la herramienta CLUTO. Así mismo, permitirá que los usuarios puedan explorar la clasificación de manera rápida y sencilla, y realizar búsquedas en los metadatos o el contenido de las tesis. La siguiente sección describe los componentes que cumplen estos requerimientos.

4.3. Diseño del sistema

En la Figura 4.2 muestra la arquitectura del sistema. Las tareas principales de los subsistemas se describen a continuación:

1. *Clasificación.* Esta tarea permite clasificar una colección de documentos digitales (tesis) cuando lo requiera el usuario administrador de acuerdo al algoritmo que elija.

2. *Exploración.* Esta tarea permite que los usuarios puedan explorar la colección de tesis, es decir, consultar la clasificación generada por el algoritmo de agrupamiento.

3. *Búsqueda de Metadatos.* Esta tarea permite que el sistema encuentre las tesis tales que, en los campos palabras clave, carrera, autor, asesor o título coincidan con algunas palabras de la consulta.

4. *Búsqueda en Contenido.* Esta tarea permite que el sistema encuentre tesis relevantes al comparar las consultas con los resúmenes con base en el modelo de espacios vectoriales.

5. *Evaluación.* Esta tarea almacena la opinión de los usuarios del sistema sobre la precisión de la recuperación. Para ello es necesario que los usuarios evalúen las respuestas sugeridas por el sistema.

6. *Ayuda.* Presenta información de cómo realizar las tareas dentro del sistema.

4.3.1. Actividades de los subsistemas

La Figura 4.3 muestra el diagrama de actividades del sistema de clasificación, exploración y búsqueda de tesis digitales. Estas actividades se describen a continuación y se agrupan de acuerdo a la tarea principal que soportan.

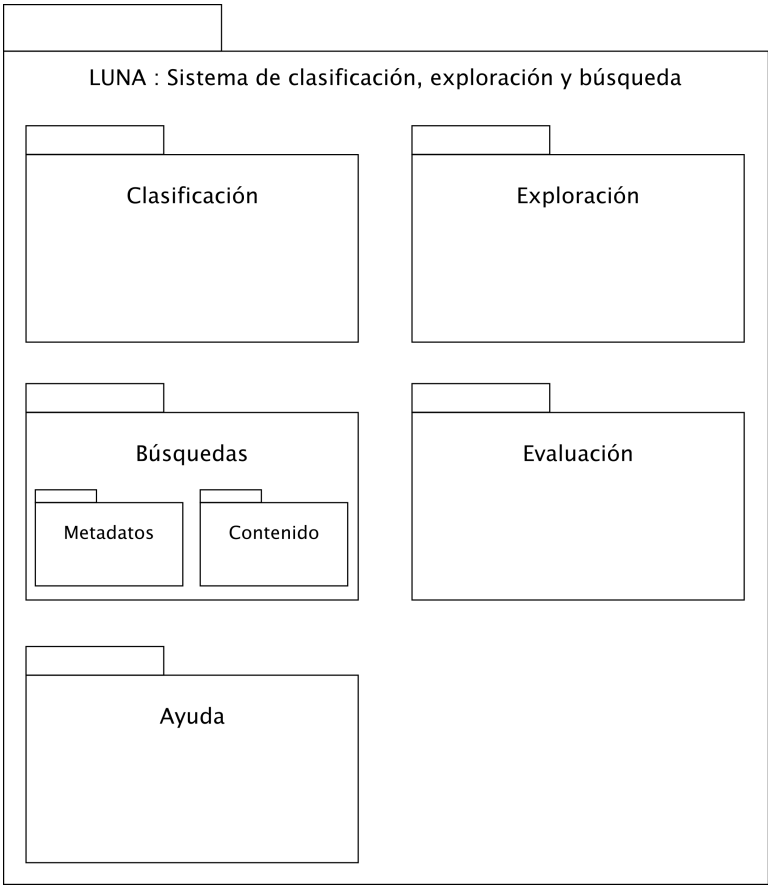


Figura 4.2. Arquitectura del sistema.

Clasificación

- *Cargar documentos de la colección.* En esta actividad se procesa la colección de tesis digitales de la UTM con el objetivo de dejarla lista para que el software CLUTO pueda acceder a la colección para clasificarla posteriormente.

- *Establecer parámetros de clasificación.* Actividad que espera que el usuario encargado de realizar la clasificación de la colección indique los parámetros tales como algoritmo de clasificación, función de similitud, función de criterio y número de grupos para ser almacenados en la base de datos del sistema y utilizados posteriormente como entrada para la clasificación.

- *Clasificar.* Esta actividad hace uso de la aplicación *vcluster.exe* del software CLUTO obtenido en [Kar06] para que con los parámetros indicados realice la agrupación de la colección.

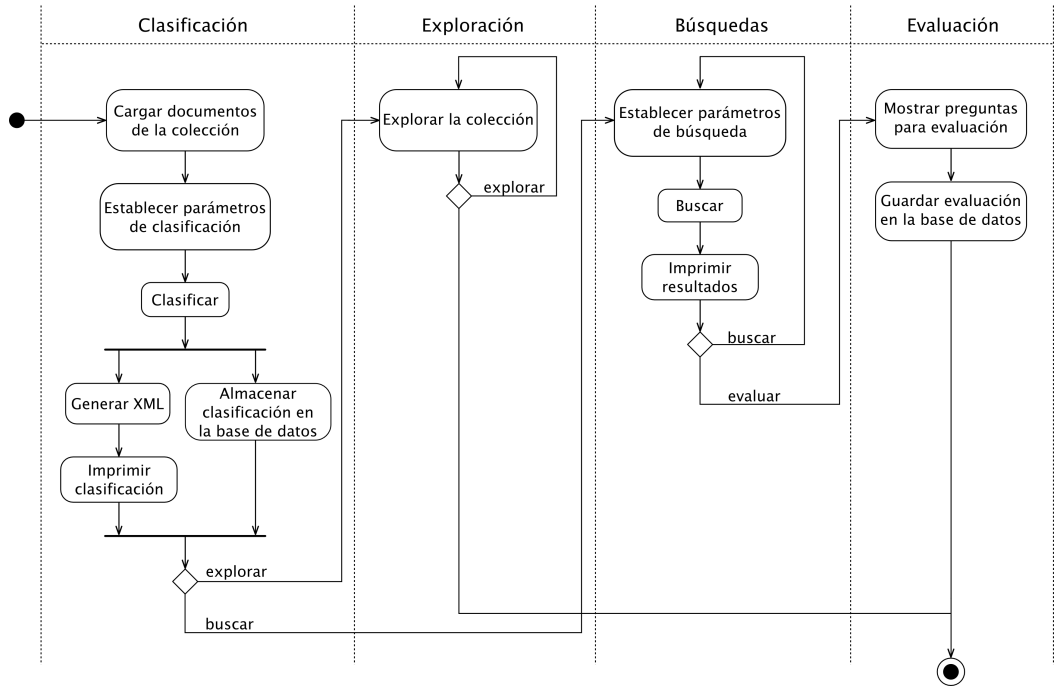


Figura 4.3. Diagrama de actividades del sistema.

- *Generar XML.* Una vez que se haya clasificado, el software CLUTO proporciona un archivo de texto que en esta actividad se interpreta para crear un único archivo XML que represente a la colección. Este archivo se escribe usando el DTD del Ejemplo 4.1, que usa como base el DTD propuesto en [MSR06]. En el Ejemplo 4.2 se muestra parte de un documento XML generado en esta actividad.

Ejemplo 4.1.

```
<?xml version="1.0" encoding="UTF-8"?>
   <!ELEMENT hierarchy (algorithm, cluster+)>
   <!ATTLIST hierarchy
      date CDATA #REQUIRED
      num_clusters CDATA #REQUIRED >
   <!ELEMENT algorithm EMPTY>
   <!ATTLIST algorithm
      name CDATA #REQUIRED
      clmethod CDATA #REQUIRED
      sim CDATA #REQUIRED
      crfun CDATA #REQUIRED>
    <!ELEMENT cluster (label, level, document*)>
    <!ATTLIST cluster
      num_documents CDATA #REQUIRED >
```

```
<!ELEMENT label (#PCDATA)>
<!ELEMENT level (#PCDATA)>
<!ELEMENT document (dc:title, dc:creator, dc:subject, dc:description,
                    dc:contributor, dc:date, dc:identifier)>
<!ELEMENT dc:title (#PCDATA)>
<!ELEMENT dc:creator (#PCDATA)>
<!ELEMENT dc:subject (#PCDATA)>
<!ELEMENT dc:description (#PCDATA)>
<!ELEMENT dc:contributor (#PCDATA)>
<!ELEMENT dc:date (#PCDATA)>
<!ELEMENT dc:identifier (#PCDATA)>
```

Ejemplo 4.2.

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF PUBLIC "-//DUBLIN CORE//DCMES DTD 2002/07/31//EN"
"http://dublincore.org/documents/2002/07/31/dcmes-xml/dcmes-xml-dtd.dtd">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dc ="http://purl.org/dc/elements/1.1/">
   <rdf:Description rdf:about="http://biblioteca.utm.mx/">
      <hierarchy date="2007-11-22 1:50:56" num_clusters="2">
         <algorithm name="CLUTO" clmethod="agglo"
                    sim="cos" crfun="g1p"/>
         <cluster num_documents="4">
            <label>0</label>
            <level>1</level>
            <document origen="8">
               <dc:title>
                  Caracterización fisicoquímica y funcional de
                  la clara  deshidratada de  huevo de codorniz
                  (coturnix coturnix japonica)
               </dc:title>
               <dc:creator>Roberto Pérez Hernández</dc:creator>
               <dc:subject>pendiente</dc:subject>
               <dc:description>
                  La clara deshidratada de huevo  de gallina es
                  utilizada  como ingrediente  funcional en di-
                  versos alimentos debido a las propiedades que
                  presenta, pero dado su costo elevado, ...
               </dc:description>
               <dc:contributor>
                  M.C. Jesús Godofredo López Luna
               </dc:contributor>
               <dc:date>2004-01-01</dc:date>
               <dc:identifier>Ing en Alimentos</dc:identifier>
            </document>
            <document origen="9">
               ...
```

```
            </document>
            <document origen="11">
            ...
            </document>
            <document origen="14">
            ...
            </document>
          </cluster>
          <cluster num_documents="4">
            ...
          </cluster>
          <cluster num_documents="3">
            ...
          </cluster>
          <cluster num_documents="4">
            ...
          </cluster>
        </hierarchy>
    </rdf:Description>
</rdf:RDF>
```

- *Almacenar clasificación en la base de datos.* Una vez se haya clasificado, se procede a guardar la clasificación en la base de datos con el objetivo de mantener un control de las clasificaciones hechas y ayudar en la recuperación de la información.

- *Imprimir clasificación.* Actividad que muestra en pantalla los documentos de la colección de manera jerárquica de acuerdo al archivo XML generado previamente. El objetivo de esta actividad es que los usuarios puedan navegar de manera sencilla y rápida por los documentos de la colección.

Una vez se haya clasificado, el usuario puede elegir entre explorar la colección o realizar búsquedas en esta clasificación.

Exploración

- *Explorar la colección.* En esta actividad se permite a los usuarios del sistema interacturar con la clasificación automática. Se mostrarán los grupos que forman la colección y el número de documentos por grupo. Una vez que el usuario seleccione el grupo a explorar, se mostrarán los títulos de las tesis que pertenecen al grupo, y al seleccionar una tesis particular, podrá consultar el título, el resumen, las palabras clave, el nombre del autor de la tesis, así como el del asesor, la carrera del autor de la tesis y la fecha de presentación.

Búsquedas

Las búsquedas pueden realizarse sobre los metadatos de las tesis digitales o bien, sobre los títulos o resúmenes. Para ambos casos, el procedimiento es similar y se describe a continuación:

- *Establecer parámetros de búsqueda.* Actividad en la cual el sistema espera que el usuario indique las palabras que quiere buscar en las tesis de la colección.
- *Buscar.* Para búsquedas en metadatos, el sistema realiza una consulta a la base de datos con los parámetros indicados por el usuario. Si se indica una búsqueda por contenido, se hace uso de la herramienta HERMES para recuperar los documentos que coincidan con los parámetros de búsqueda introducidos por el usuario.
- *Imprimir resultados.* Una vez que se obtengan ya sea de la base de datos o de la salida de HERMES las tesis digitales que resulten de la búsqueda, se muestra en la pantalla en forma de lista para que el usuario pueda explorarlas y pueda determinar qué tan útiles le fueron los documentos recuperados.

El usuario puede decidir entre continuar buscando documentos o evaluar el resultado propuesto.

Evaluación

- *Mostrar preguntas para evaluación.* En esta actividad se muestra un cuestionario para que el usuario pueda expresar qué tan útiles le fueron los resultados que dio el sistema al realizar una búsqueda por contenido.
- *Guardar evaluación en la base de datos.* Una vez que el usuario haya contestado el cuestionario propuesto, el sistema guarda los resultados en la base de datos con la finalidad de que en el futuro se puedan interpretar y decidir sobre éstos.

4.3.2. Diseño de la base de datos

En la Figura 4.4 se muestra el diagrama Entidad-Relación de la base de datos. En ella se marcan dos regiones, la región inferior, que encierra las entidades Usuarios y Resultados; se usa para almacenar las evaluaciones de los usuarios sobre la precisión de la recuperación. Por otro lado, la región izquierda que encierra las entidades Tesis y Palabras y la relación entre ellas, Tesis-Palabras, son usadas para facilitar la recuperación a Hermes, en especial la relación entre estas entidades. En la Tabla 4.1 se muestran los atributos usados de cada entidad y una breve descripción.

Tabla	Atributos	Descripción
Tesis	clave_archivo, id_grupo_antes, id_grupo_ahora, titulo, autor, pal_claves, resumen, asesor, fecha, carrera	Tabla que almacena los metadatos de las tesis digitales y el nombre del archivo XML que la contiene.
Grupos	id_grupo, id_clasificacion, etiqueta	Almacena los grupos generados por una clasificación.
Clasificación	id_clasificacion, identificador, id_parametro, fecha	Tabla que guarda cada clasificación realizada a la colección de documentos, así como la fecha en que fue realizada y el identificador de quien la realizó.
ParametrosClas	id_parametro, clmethod, sim, crfun, num_grupos	En esta tabla se guardan los parámetros usados por el algoritmo de agrupamiento.
UsuariosAdmon	identificador, contraseña, nombres, apellidop, apellidom, correo_elec	Almacena información del o de los administradores del sistema.
Resultados	correo_e, id_clasificacion, precision, consulta	Tabla que guarda tanto la evaluación de los usuarios a la respuesta del sistema como la consulta realizada que produjo tales resultados.
Usuarios	correo_e, nombres, apellidop, apellidom, area	Guarda información sobre el usuario que realiza una evaluación.
Palabras	id_word, word, tfMax, idf	En esta tabla se almacenan las palabras de toda la colección y es usada por la herramienta Hermes para realizar la recuperación de información.
Tesis_Palabras	id_word, clave_archivo, tf	Almacena las palabras que contiene cada tesis junto con la frecuencia de la palabra en el documento, esta tabla también es usada por Hermes para la recuperación de las tesis digitales.

Tabla 4.1. Descripción de las tablas de la base de datos.

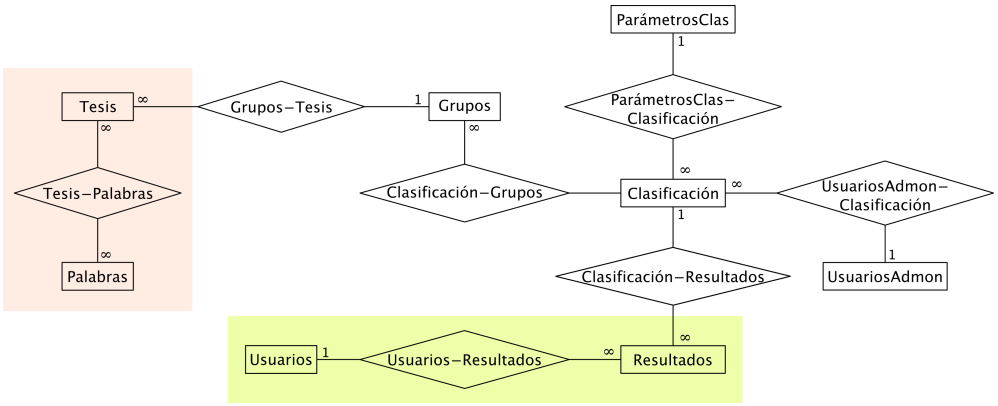


Figura 4.4. Diagrama Entidad - Relación de la base de datos.

Capítulo 5

Implementación del sistema Luna

El sistema de clasificación, exploración y búsqueda de tesis digitales denominado Luna, tiene las características siguientes:

- *Es accesible vía web.* El sistema está programado utilizando tecnologías web. Para la interfaz y la incorporación de herramientas se utiliza PHP 4.4[1]. Para almacenar los datos se utiliza el manejador MySQL 5.0[2], y como servidor web se emplea Apache 2.2.3[3].

- *Clasifica tesis digitales utilizando diferentes algoritmos.* El sistema incorpora dos aplicaciones que se desarrollaron en JAVA[4], una de ellas se encarga de preparar los documentos para que la aplicación *vcluster* del software CLUTO[5] realice la clasificación de documentos digitales usando 4 diferentes algoritmos.

- *Recupera tesis digitales.* La segunda aplicación que incorpora el sistema Luna es la que utiliza la herramienta Hermes[6] para realizar la recuperación por contenido de documentos digitales de la colección.

5.1. Inicio al sistema

El proceso de clasificación debe estar restringido al administrador de la biblioteca digital de la UTM. Por tal motivo, el sistema maneja dos tipos de usuarios, *administrador* y *usuario normal*. En la Tabla 5.1 se muestran las tareas que cada tipo de usuario puede realizar, y en la Figura 5.1 se muestra la pantalla de bienvenida al sistema.

[1] http://www.php.net/
[2] http://www.mysql.com/
[3] http://www.apache.org/
[4] http://www.java.com/
[5] http://glaros.dtc.umn.edu/gkhome/views/cluto
[6] http://catarina.udlap.mx:9090/u_dl_a/hermes/espanol.htm

Tarea	Administrador	Usuario normal
Clasificar	Si	No
Explorar la colección	Si	Si
Realizar búsquedas de metadatos	Si	Si
Realizar búsquedas por contenido	Si	Si
Evaluar precisión de los documentos recuperados	Si	Si

Tabla 5.1. Tareas que pueden realizar los dos tipos de usuarios.

Figura 5.1. Pantalla de bienvenida al sistema Luna.

5.2. Clasificación de la colección de tesis digitales

Para implementar esta sección, ésta se dividió en dos partes: el procesamiento de los documentos y la clasificación en sí misma. Dado que el procesamiento de los documentos es necesario para realizar las pruebas a los algoritmos de agrupamiento mencionadas anteriormente, la implementación reutiliza esas clases[7], las cuales se describen a continuación:

- *Clases SimpleParse, Term y DropStopWords.* Necesarias para quitar las *stopwords* (palabras que no aportan información útil en el contenido de algún documento).

- *Clase TerminosOrdenados.* Esta clase contiene a todos los términos que aparecen en toda la colección, los ordena utilizando el algoritmo *quicksort* y les agrega una llave para su recuperación posterior.

- *Clase Termino.* Clase que almacena un término (palabra) con los siguientes atributos: *palabra* el término almacenado; *id* el número de término que le corresponde en relación a toda la colección, este número se calcula cuando se almacena la colección; *tf* la frecuencia del término en el documento; *df* el número de documentos que contiene el término; *d_tfidf* contiene el resultado de $tflog(N/df)$; *d_tfidf_normal* el valor de d_tfidf normalizado y N el número de documentos en la colección.

- *Clase Documento.* Esta clase genera un objeto Documento que contiene como atributos el nombre del documento y un vector de objetos de tipo Termino.

- *Clase Main.* En esta clase se leen los documentos que pertenecen a la colección, utiliza las clases anteriores y crea los siguientes archivos: *colecciónInput.txt* que es el archivo de entrada para CLUTO, y *terminosTF.txt* archivo que contiene todos los términos de la colección de tesis digitales.

Una vez que se procesan los documentos de la colección, el sistema muestra la pantalla de la Figura 5.2. En ella, el administrador puede realizar la clasificación de la colección utilizando los parámetros que obtuvieron mejores resultados en las pruebas a los algoritmos (opción *Clasificar*) o puede establecer sus propios parámetros de clasificación (opción *Establecer parámetros de clasificación*) como lo muestra la Figura 5.3. Para cualquiera de las dos opciones, el sistema utiliza la aplicación *vcluster* para clasificar la colección. Los parámetros posibles que se pueden establecer, se muestran en la Tabla 5.2.

[7]El Apéndice A muestra el diagrama de clases correspondiente

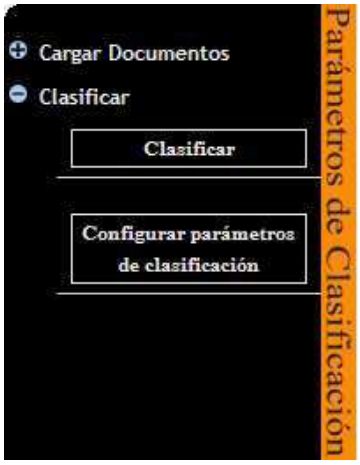


Figura 5.2. Pantalla que muestra las dos opciones para clasificar una colección.

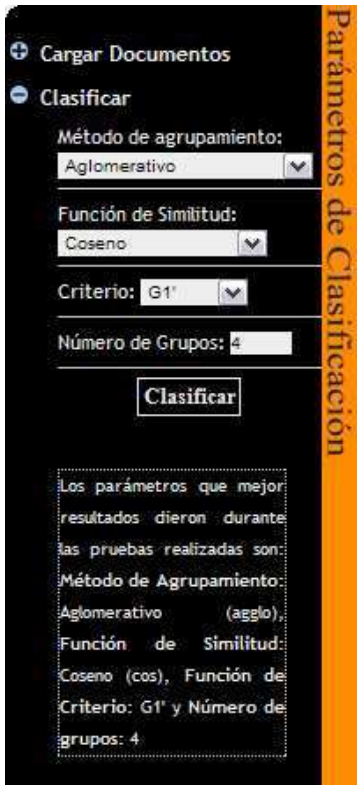


Figura 5.3. Pantalla para ingresar los parámetros y algoritmo con los que se quiera clasificar la colección de documentos.

Parámetro	Valor
Método de agrupamiento	Aglomerativo, Bisecciones repetidas, Bisecciones repetidas (optimizado), Directo, Aglomerativo (optimizado)
Función de similitud	Coseno, Correlación, Distancia euclidiana
Criterio	I1, I2, E1, G1, G1′, H1, H2, SLink (enlace simple), WSLink (enlace simple con grupos ponderados), Clink (enlace completo), WCLink (enlace completo con grupos ponderados), UPGMA (*Unweighted Pair Group Method with Arithmetic mean*)
Número de grupos	Número entero mayor que cero

Tabla 5.2. Parámetros de entrada para clasificar.

5.3. Exploración de la colección

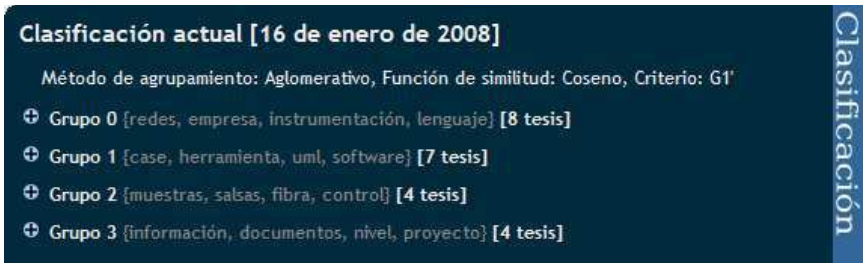
Uno de los objetivos de desarrollar un sistema que clasifique tesis digitales es acceder a estos documentos de forma ordenada, rápida y sencilla. La clasificación generada se muestra en la Figura 5.4 y está compuesta como sigue:

- *Información de la clasificación.* Se muestran datos como fecha de la clasificación, método de agrupamiento, función de similitud y función de criterio usados (Figura 5.4 (a)).

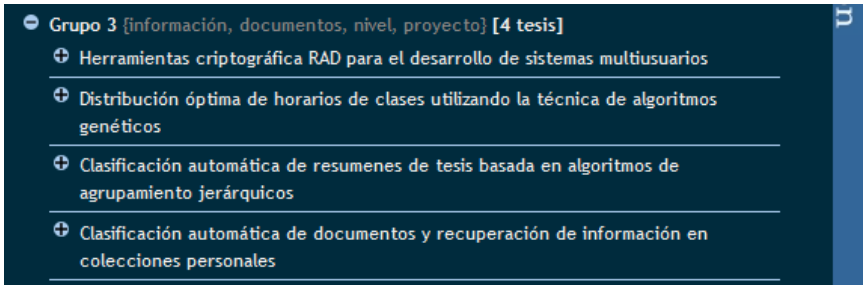
- *Grupos generados.* Muestra una lista de los grupos generados con la etiqueta correspondiente, los 4 términos más representativos del grupo y entre corchetes el número de documentos que contiene (Figura 5.4 (a)). Para obtener los términos más representativos del grupo, se realiza una consulta a la base de datos para buscar los términos que contenidos en los documentos del grupo con mayor frecuencia; cabe mencionar que estos datos no son proporcionados por el algoritmo y sólo dan una idea de los temas de los documentos de ese grupo.

- *Tesis contenidas en los grupos.* Por cada grupo se listan los títulos de las tesis que contienen (Figura 5.4 (b)).

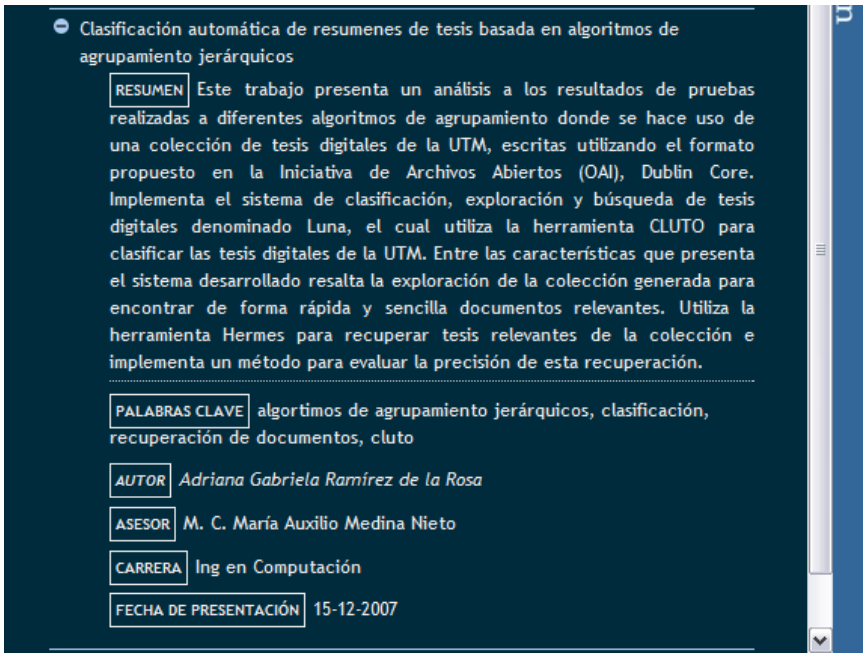
- *Información de las tesis.* Muestra el título de la tesis, el resumen, las palabras clave, el nombre del autor, nombre del asesor, la carrera del autor de la tesis y la fecha de presentación (Figura 5.4 (c)).



(a) Grupos



(b) Tesis en los grupos



(c) Datos de las tesis

Figura 5.4. Información mostrada al explorar la colección.

5.4. Búsquedas en la colección

El sistema Luna realiza búsquedas sobre el contenido de las tesis digitales almacenadas en el sistema, así como búsquedas sobre los metadatos de los mismos. Para las búsquedas de metadatos se realiza una consulta a la base de datos donde se encuentran almacenadas las tesis con sus respectivos metadatos. En la Figura 5.5 se muestra un ejemplo de búsquedas de metadatos en el sistema (derecha) junto con el resultado obtenido (izquierda). La siguiente sección describe las búsquedas por contenido.

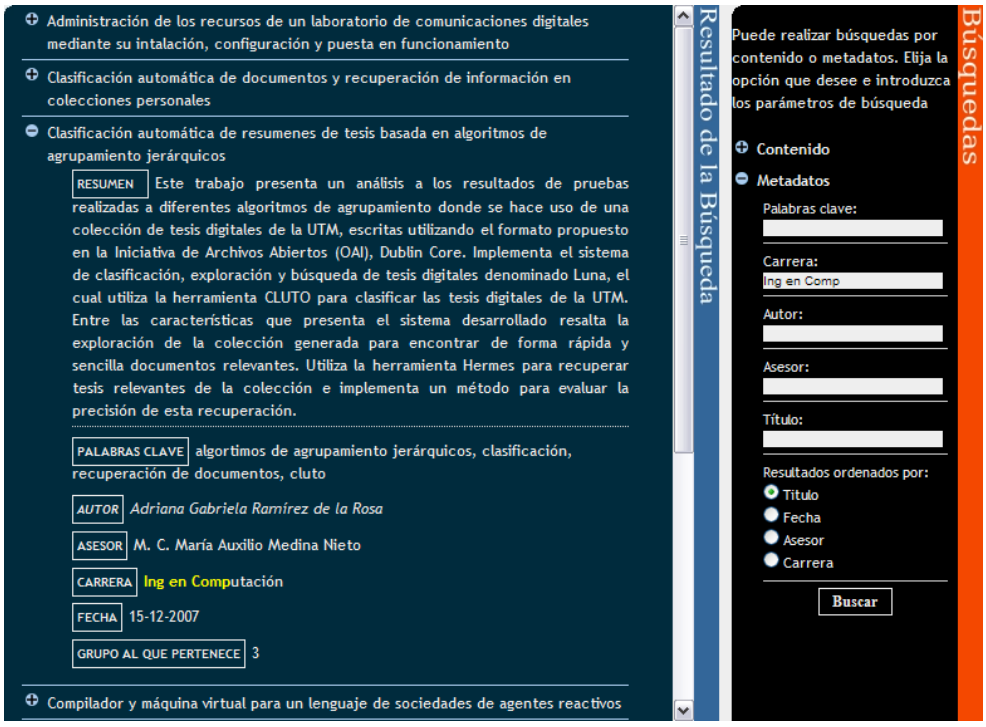


Figura 5.5. Pantalla que muestra una búsqueda de metadatos y el resultado obtenido.

5.4.1. Recuperación de información

Para realizar esta tarea se usa la herramienta Hermes, ésta proporciona una interfaz (*MiTesisCollection*) en la cual se especifica la colección a la cual debe acceder, así como la base de datos en la cual se encuentra la información de dicha colección. Luego se emplea una clase en Java (*BuscadorTesis*) que permita recuperar documentos con parámetros de búsquedas dados desde la interfaz de Luna.

Hermes proporciona como resultado de la recuperación un objeto que contiene el identificador del documento recuperado de la base de datos, el término por el cual

ID: id_1
Term: t_1
Idf: i_1
Maxtf: m_1
...
ID: id_n
Term: t_n
Idf: i_n
Maxtf: m_n

Tabla 5.3. Formato del archivo de texto que contiene los documentos recuperados por Hermes.

fue recuperado y la frecuencia de este término; estos datos se escriben en un archivo de texto en la clase BuscadorTesis, con el objetivo de leerlo y recuperar los datos de las tesis desde el sistema Luna. El archivo generado tiene el formato mostrado en la Tabla 5.3.

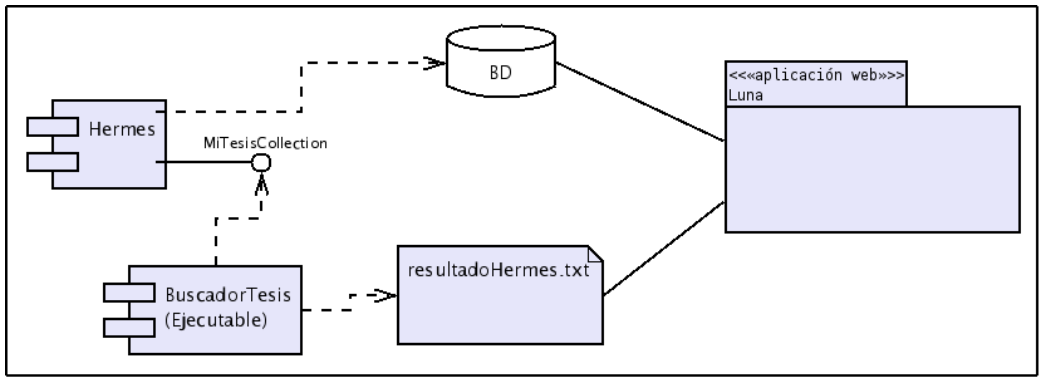


Figura 5.6. Diagrama de componentes que ilustra la interacción de la herramienta Hermes con el sistema Luna.

En la Figura 5.6 se muestra la interacción de la herramienta Hermes con el sistema Luna. Además de todos los componentes con que consta Hermes, existe una clase que es la interfaz entre la herramienta y la base de datos relacional en donde se almacenan los documentos y las palabras que éstos contienen, esta clase (MiTesisCollection) pertenece al paquete *irserver.jar* de Hermes. Se programó una aplicación que se utiliza en la recuperación (BuscadorTesis). El sistema Luna ejecuta la aplicación BuscadorTesis, ésta indica a Hermes que realice la recuperación utilizando la base de datos relacional a través de la interfaz (MiTesisCollection). Al finalizar, Luna recupera la información que Hermes proporciona y completa la respuesta para el usuario utilizando información de la base de datos.

5.5. Evaluación de la precisión de la recuperación

El sistema Luna tiene un módulo que muestra al usuario un cuestionario sencillo para que posteriormente sea posible calcular la precisión de la recuperación de forma automática. Las respuestas dadas se almacenan en la base de datos con el objetivo de poder tomar decisiones sobre estos datos; sin embargo no corresponde a este trabajo ver o interpretar los datos obtenidos en esta tarea, pues ésto nada aporta al cumplimiento del objetivo general de esta tesis. En la Figura 5.7 se muestra dicho cuestionario. La primera pregunta se realiza con el objetivo de obtener el número de documentos recuperados en una consulta. La siguiente pregunta se hace para conocer cuántos documentos de los recuperados son relevantes para el usuario; aquí se presentan 5 opciones para responder, ya que es difícil que el usuario recuerde exactamente el número de documentos que le fueron útiles; para obtener el valor numérico de esta respuesta, se utiliza una regla de tres donde a la respuesta *Todos* le corresponde el 100 % de los documentos recuperados, es decir que el número de documentos relevantes es igual al número de los documentos recuperados; así como a *Ninguno* le corresponde el valor numérico cero. La tercer pregunta se realiza con el fin de poder reproducir las consultas realizadas. Por último, se le hacen tres preguntas para conocer más acerca del evaluador. Con el objetivo de disminuir el trabajo al usuario al contestar este cuestionario, el sistema almacena temporalmente la última consulta por contenido realizada, así el usuario deberá contestar dos preguntas menos, en la Figura 5.8 se muestra un ejemplo de este caso.

El usuario tiene la decisión de evaluar el resultado de sus búsquedas. El sistema no obliga a realizar esta tarea a todos los usuarios o cada que muestra el resultado de una búsqueda.

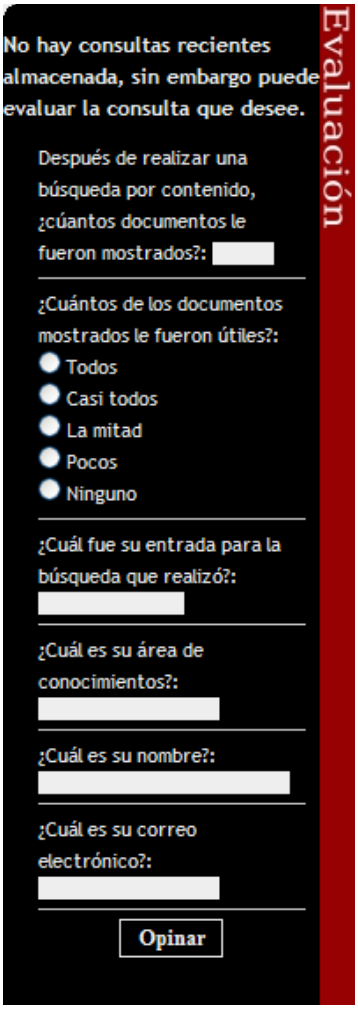


Figura 5.7. Cuestionario usado para la evaluación de la recuperación.

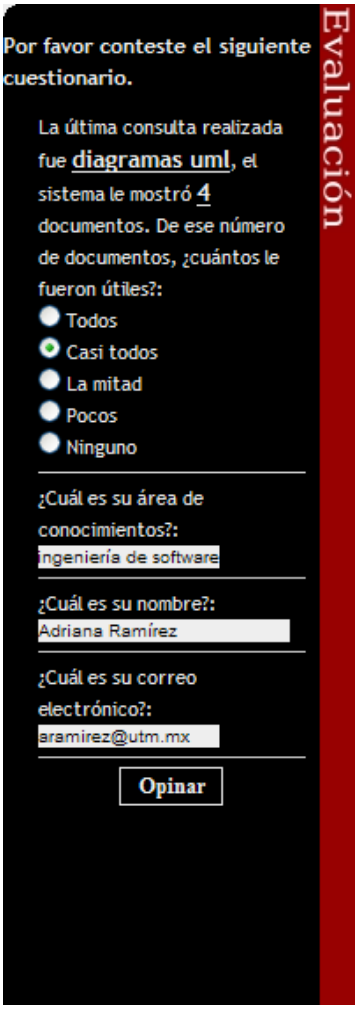


Figura 5.8. Cuestionario usado para la evaluación de la recuperación cuando ya se ha realizado alguna consulta por contenido

Capítulo 6

Conclusiones

Este capítulo resume los aspectos más importantes de esta tesis y menciona el trabajo a futuro.

Para implementar un mecanismo automático de clasificación y exploración que permitiera consultar las tesis de la UTM con base en el análisis de resúmenes, se realizaron las actividades que se describen en los párrafos siguientes.

Se eligió de la colección de tesis de la UTM 20 de ellas para realizar las pruebas que determinarían el mejor algoritmo de agrupamiento, de acuerdo a las medidas entropía y pureza. Se investigó los tipos de algoritmos de agrupamiento y sus características más sobresalientes, así como los requerimientos técnicos, como la entrada de los algoritmos, para realizar la clasificación de la colección de prueba. Posteriormente, se procesaron los documentos para cubrir los requerimientos técnicos de los algoritmos antes de realizar las pruebas; para ésto se utilizaron dos herramientas que implementan distintos algoritmos de agrupamiento, estas herramientas son DocCluster y Cluto.

Se obtuvo que el algoritmo de clasificación que da mejores resultados con la colección de prueba, de acuerdo a las medidas que determinan la calidad de los grupos, fue el *jerárquico aglomerativo* con parámetros: *coseno* como función de similitud, *G1'* como función de criterio y 4 grupos. Este algoritmo está implementado en la herramienta Cluto.

Se modificó la estructura (el DTD)[MSR06] de cómo debe guardarse la colección realizada por el algoritmo elegido; el objetivo es mostrar de forma rápida y sencilla la clasificación generada.

Las búsquedas son una parte importante del sistema de clasificación, exploración y búsqueda que se implementó, por lo que se diseñó una base de datos capaz de almacenar los metadatos de las tesis de la colección. Aunado a ésto, se hizo uso de la herramienta Hermes, misma que recupera documentos con base en el modelo de espacios vectoriales.

Para evaluar la precisión del sistema de recuperación por contenido, en el diseño de la base de datos se consideró almacenar las evaluaciones de los usuarios del sistema para su análisis posterior.

Por todo lo anterior se concluye que es posible encontrar tesis relevantes a través

de mecanismos de recuperación de información o al explorar la organización de los documentos en la colección de tesis digitales de la UTM.

Como puede constatarse a lo largo del documento, el sistema que se implementa en esta tesis, utiliza componentes existentes tales como Cluto y Hermes. Sin embargo, no existe hasta el momento una herramienta que integre las tareas de los componentes utilizados, además de que para utilizar Cluto o Hermes, los usuarios tienen que tener conocimiento básico de algoritmos de clasificación y recuperación de información, respectivamente. Es por lo anterior que Luna tiene una aportación muy importante, pues se pueden realizar tareas complejas de una forma sencilla y sin necesidad de conocimiento especializado.

6.1. Trabajo a futuro

Esta sección propone tres líneas de investigación como trabajo futuro. La primera considera la aplicación del sistema propuesto a otros dominios o fuentes de referencia con modificaciones mínimas si los datos de éstas se representan como registros de metadatos de la Iniciativa de Archivos Abiertos. Por ejemplo, al material que se almacena en la biblioteca de la UTM tal como DVD's, revistas o CD's.

La segunda línea propone la extensión del sistema de clasificación, exploración y búsqueda para clasificar la colección de tesis considerando el contenido. A la fecha, sólo se utilizan dos elementos del formato de metadatos Dublin Core: título y resúmen. La clasificación basada en estos elementos junto con el contenido, podría ser más detallada y específica. Sin embargo, para demostrarlo, sería necesario contar con la evaluación de la precisión de la recuperación.

En este sentido, la tercer línea de investigación consiste en la evaluación automática de la precisión, independientemente del módulo que calcule la precisión de la recuperación con base en el tipo de algoritmo seleccionado y las respuestas del cuestionario descrito en el Capítulo 5, de forma que después de un tiempo de poner a prueba el sistema, se pudiera sugerir los parámetros y el algoritmo más adecuado para cada aplicación.

Apéndice A

Diagramas de clases de Java

En esta sección se muestran las clases de Java utilizadas tanto en las pruebas de los algoritmos de agrupamiento como en el sistema de clasificación, exploración y búsqueda (Luna).

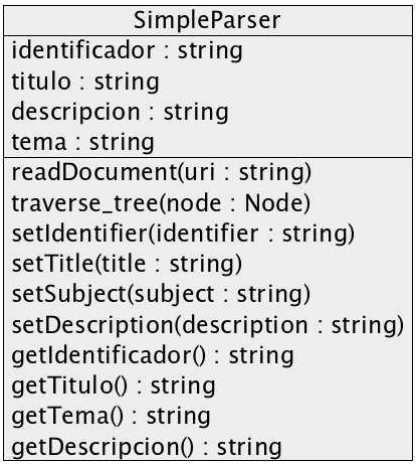


Figura A.1. Diagrama de la clase SimpleParser.

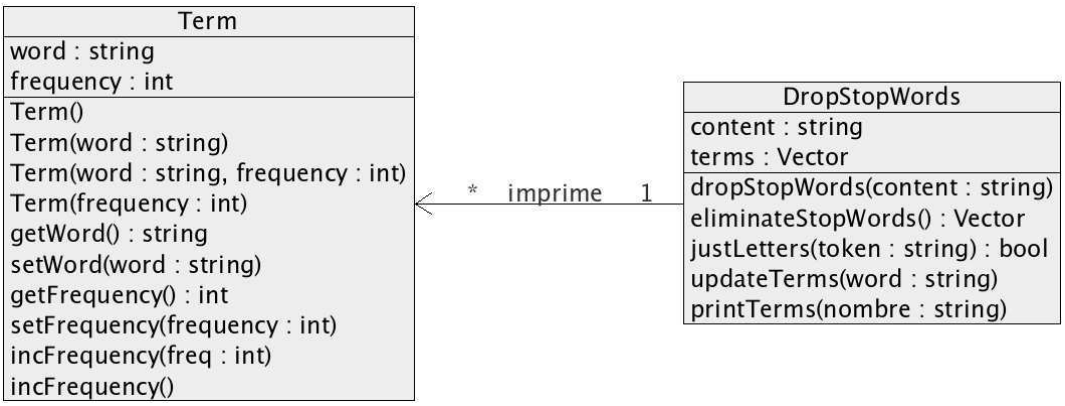


Figura A.2. Diagrama de las clases Term y DropStopWords.

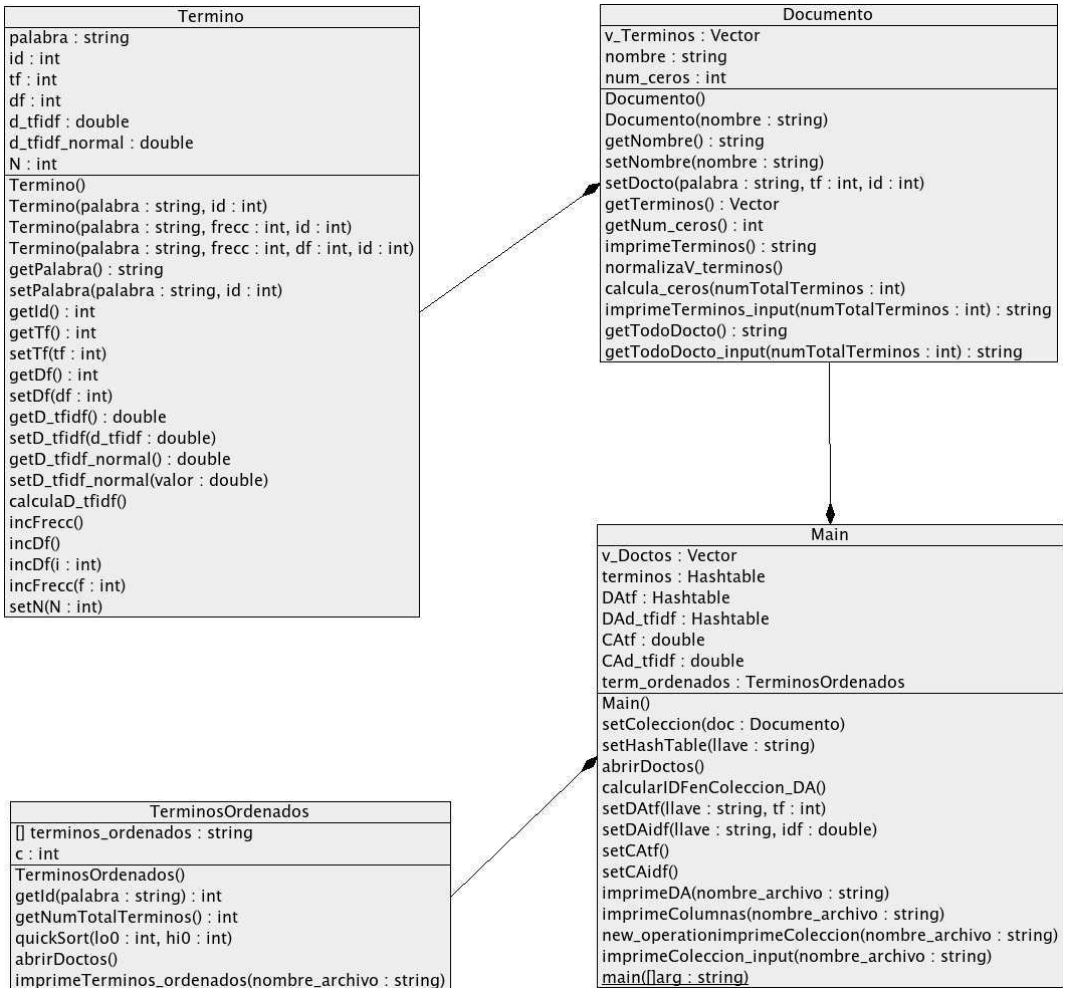


Figura A.3. Diagrama de clases de la sección que se usa para describir una colección.

Apéndice B

Dendrogramas generados por gCLUTO

Este apéndice muestra los dendrogramas resultantes de las 4 pruebas con los mejores resultados que se obtuvieron utilizando el paquete de la herramienta CLUTO denominado gCLUTO.

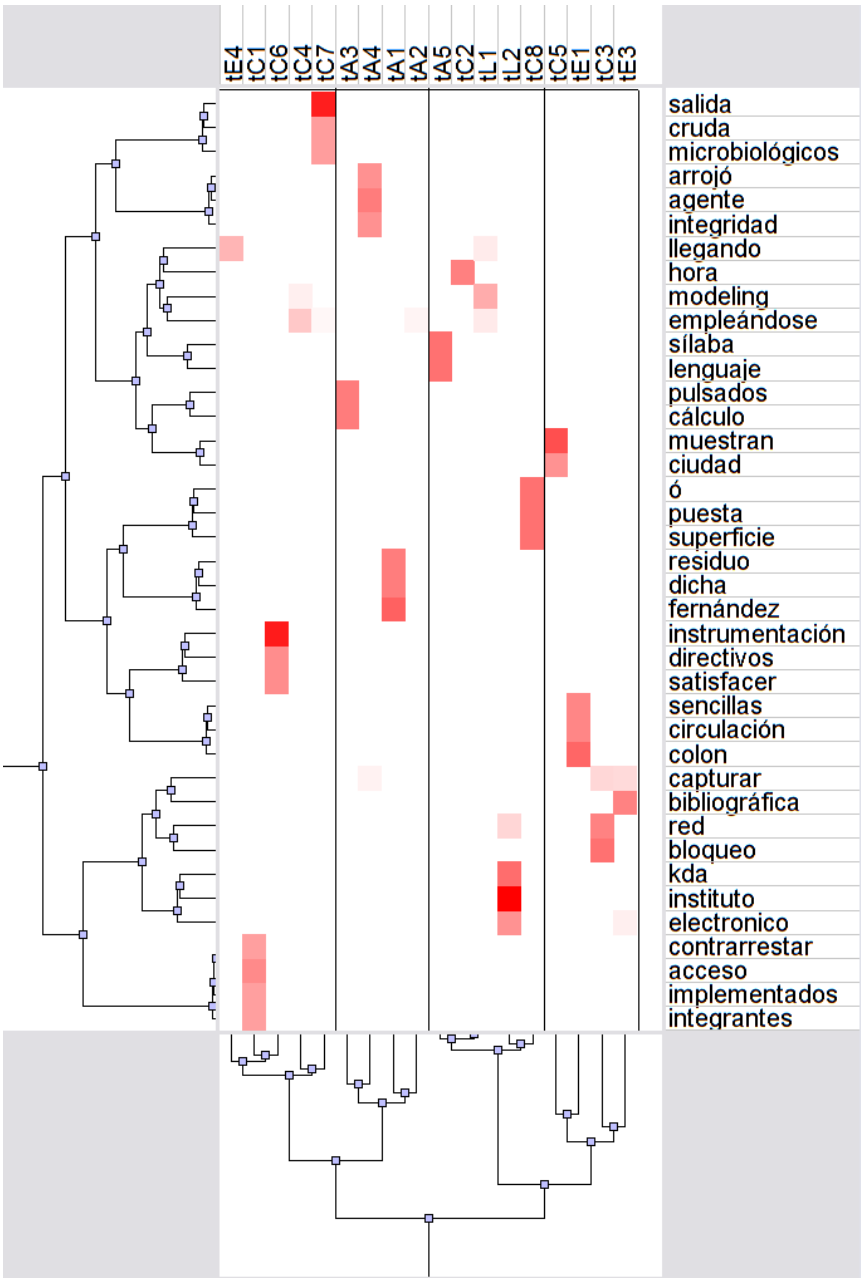


Figura B.1. Dendrograma generado utilizando el algoritmo Directo con función de similitud coseno, criterio H1 y 4 grupos.

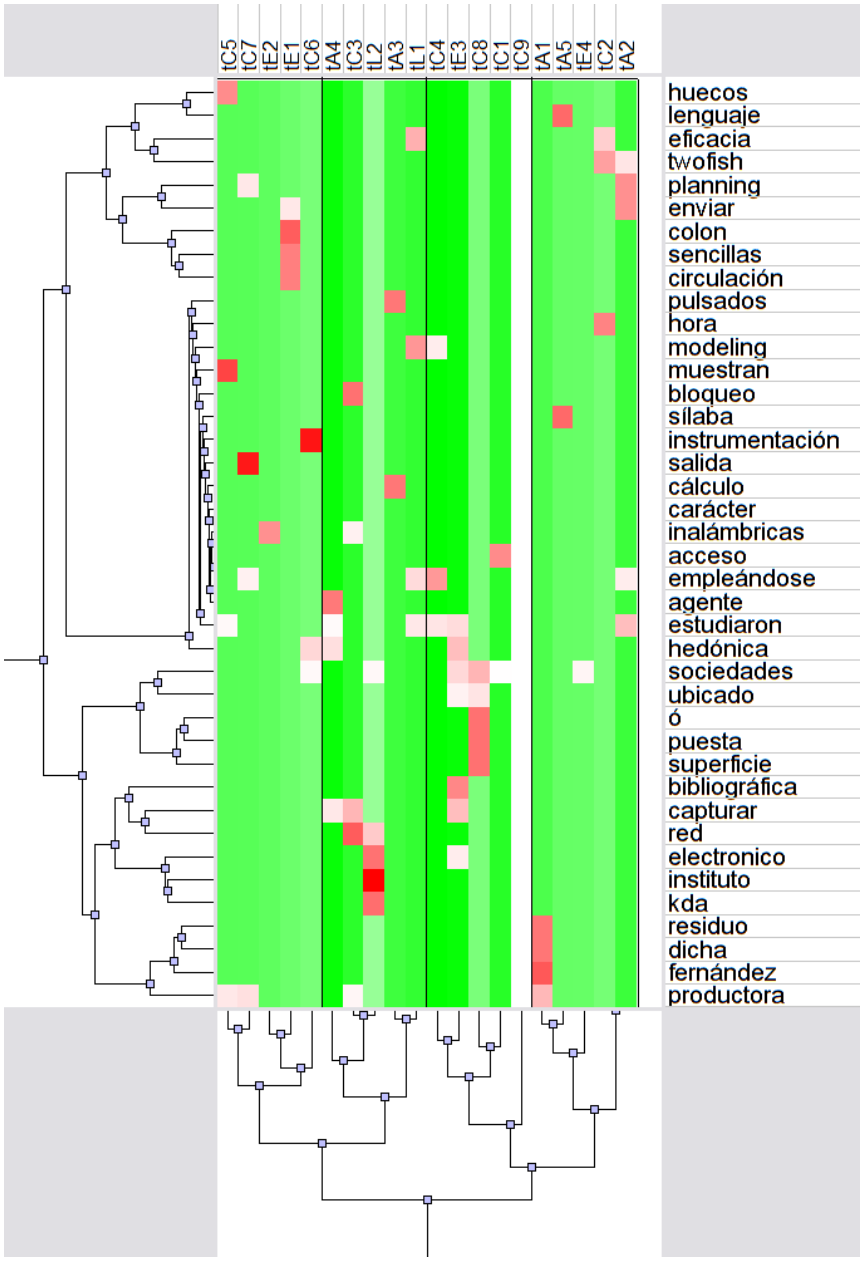


Figura B.2. Dendrograma generado utilizando el algoritmo Directo con función de similitud coeficiente de correlación, criterio E1 y 4 grupos.

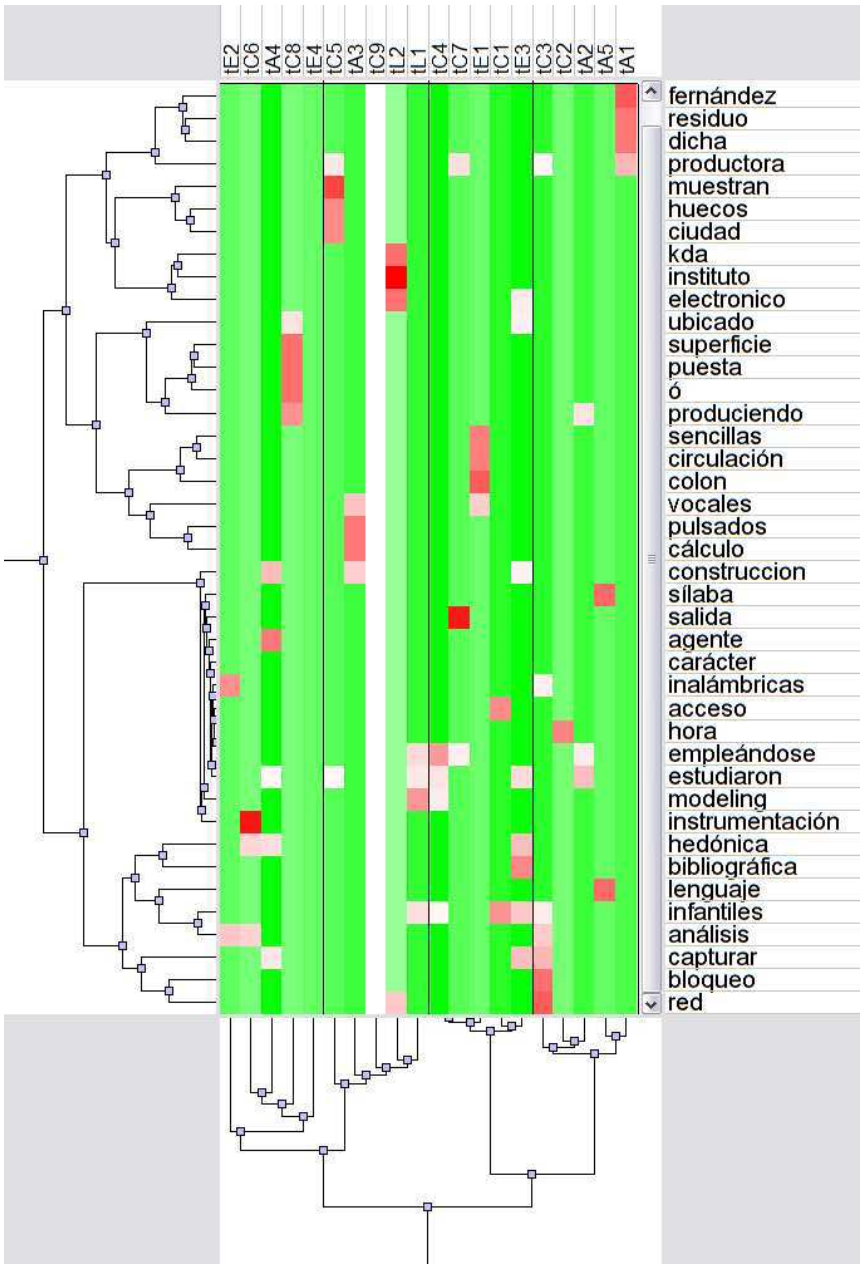


Figura B.3. Dendrograma generado utilizando el algoritmo Directo con función de similitud coeficiente de correlación, criterio H2 y 4 grupos.

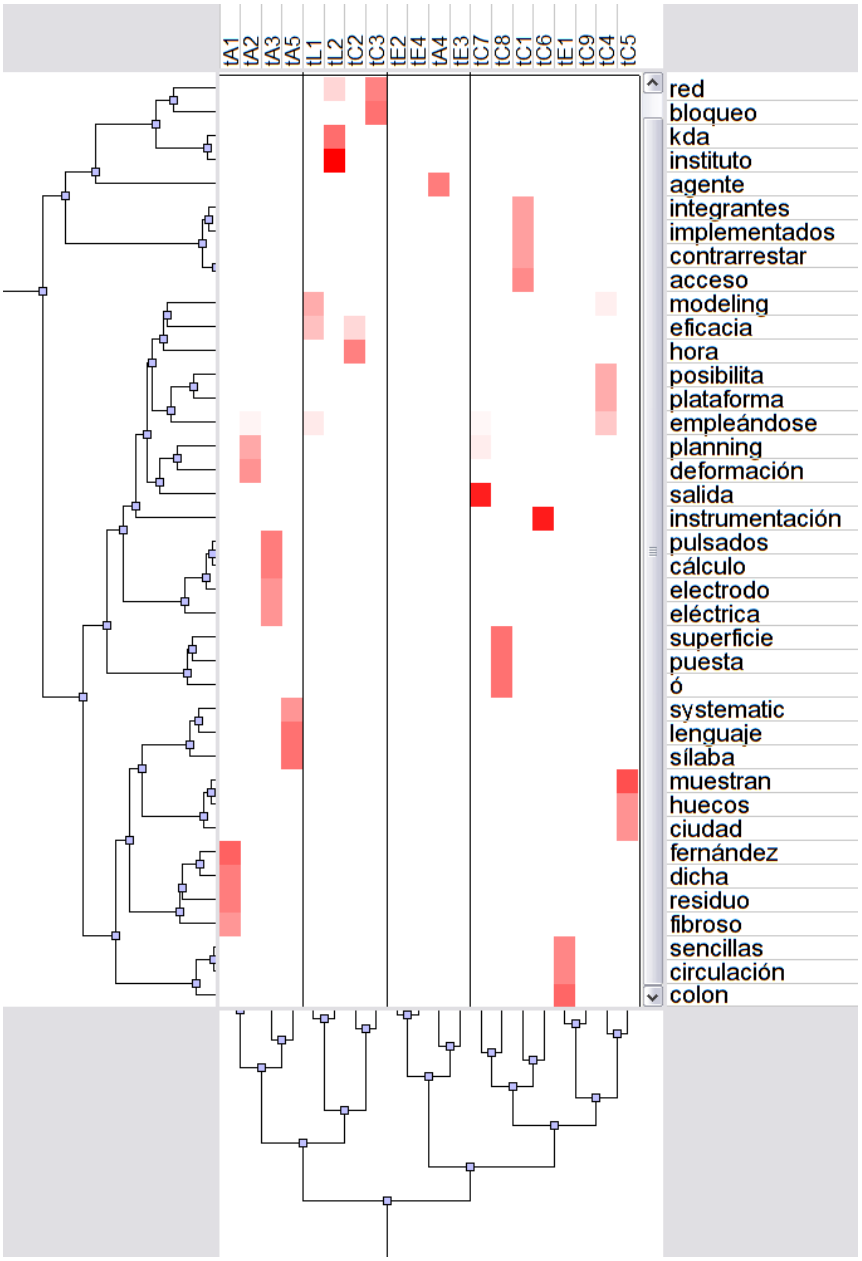


Figura B.4. Dendrograma generado utilizando el algoritmo jerárquico aglomerativo con función de similitud coseno, criterio G1'y 4 grupos.

Apéndice C

Lista de StopWords

En esta sección se muestran las *stopwords* usadas en esta tesis.

a	actualmente	adelante	además	afirmo
afirmó	agrego	agregó	ahi	ahora
ajena	ajenas	ajeno	ajenos	al
algo	algún	alguna	algunas	alguno
algunos	allá	alli	alrededor	ambos
ante	anterior	antes	apenas	aproximadamente
aquel	aquella	aquellas	aquello	aquellos
aquí	aseguro	asi	así	aun
aunque	ayer	bajo	bien	buen
buena	buenas	bueno	buenos	cada
casi	cerca	cierta	ciertas	cierto
ciertos	comento	como	cómo	con
conmigo	conocer	considera	considero	consigo
contigo	contra	cosas	creo	cual
cuales	cualquier	cualquiera	cualquieras	cuan
cuando	cuanta	cuantas	cuanto	cuantos
cuenta	da	dado	dan	dar
de	debe	deben	debido	decir
dejar	dejo	del	demás	demasiada
demasiadas	demasiado	demasiados	dentro	describan
describen	desde	después	dice	dicen
dicho	dieron	diferente	diferentes	dijeron
dijo	dio	donde	durante	e
ejemplo	el	él	ella	ellas
ellas	ello	ellos	embargo	en
encuentra	entonces	entre	era	eran
eras	eres	es	esa	esas
ese	eso	esos	esta	está
ésta	estaba	estabais	estábamos	estaban

estabas	estad	estada	estadas	estado
estados	estáis	estamos	estan	están
estando	estar	estara	estará	estarán
estarás	estaré	estaréis	estaremos	estaría
estaríais	estaríamos	estarían	estarías	estas
éstas	este	esté	estéis	estemos
estén	estés	esto	estos	éstos
estoy	estuve	estuviera	estuvierais	estuviéramos
estuvieran	estuvieras	estuvieron	estuviese	estuvieseis
estuviésemos	estuviesen	estuvieses	estuvimos	estuviste
estuvisteis	estuvo	ex	existe	existen
explico	fin	fue	fuera	fuerais
fuéramos	fueran	fueras	fueron	fuese
fueseis	fuésemos	fuesen	fueses	fui
fuimos	fuiste	fuisteis	gran	grandes
ha	habéis	haber	había	habíais
habíamos	habían	habías	habida	habidas
habido	habidos	habiendo	habrá	habrán
habrás	habré	habréis	habremos	habría
habríais	habríamos	habrían	habrías	hace
hacen	hacer	hacerlo	hacia	haciendo
han	has	hasta	hay	haya
hayáis	hayamos	hayan	hayas	he
hecho	hemos	hicieron	hizo	hoy
hube	hubiera	hubierais	hubiéramos	hubieran
hubieras	hubieron	hubiese	hubieseis	hubiésemos
hubiesen	hubieses	hubimos	hubiste	hubisteis
hubo	igual	incluso	indico	informo
jamas	junto	juntos	la	lado
las	le	les	llego	lleva
llevar	lo	los	luego	lugar
manera	manifesto	mas	más	mayor
me	mediante	mejor	menciono	menos
mi	mí	mía	mías	mientras
mío	míos	mis	misma	mismas
mismo	mismos	mucha	muchas	muchisima
muchisimas	muchisimo	muchisimos	mucho	muchos
muy	nada	nadie	ni	ningun
ninguna	ningunas	ninguno	ningunos	no
nos	nosotras	nosotros	nuestra	nuestro
nuestros	nueva	nuevas	nuevo	nuevos
nunca	o	ó	os	otra
otras	otro	otros	para	parece

parecer	parte	partir	pasada	pasado
pero	pesar	poca	pocas	poco
pocos	podemos	podra	podran	podria
podrian	poner	por	porque	posible
primer	primera	primero	primeros	principalmente
propia	propias	propio	propios	proximo
proximos	pudo	pueda	puede	pueden
pues	que	qué	quedo	queremos
querer	quien	quienes	quienesquiera	quienquiera
quiere	realizado	realizar	realizo	respecto
se	sea	seáis	seamos	sean
seas	segun	segunda	segundo	sentid
sentida	sentidas	sentido	sentidos	ser
sera	será	seran	serán	serás
seré	seréis	seremos	seria	sería
seríais	seríamos	serían	serías	si
sí	sido	siempre	siendo	siente
sigue	siguiente	siguientes	sin	sino
sintiendo	sobre	sola	solamente	solas
solo	sólo	solos	somos	son
soy	Sr	Sra	Sres	Srita
Sta	su	sus	suya	suyas
suyo	suyos	tal	tales	también
tampoco	tan	tanta	tantas	tanto
tantos	te	tendrá	tendrán	tendrás
tendré	tendréis	tendremos	tendría	tendríais
tendríamos	tendrían	tendrías	tened	tenéis
tenemos	tener	tenga	tengáis	tengamos
tengan	tengas	tengo	tenía	teníais
teníamos	tenían	tenías	tenida	tenidas
tenido	tenidos	teniendo	tercera	ti
tiene	tienen	tienes	toda	todas
todavia	todo	todos	tomar	total
tras	trata	traves	tu	tú
tus	tuve	tuviera	tuvierais	tuviéramos
tuvieran	tuvieras	tuvieron	tuviese	tuvieseis
tuviésemos	tuviesen	tuvieses	tuvimos	tuviste
tuvisteis	tuvo	tuya	tuyas	tuyo
tuyos	última	últimas	último	últimos
un	una	unas	uno	unos
usted	ustedes	va	vamos	van
varias	varios	veces	ver	vez
vosostras	vosostros	vuestra	vuestras	vuestro

vuestros	y	ya	yo

Bibliografía

[BYRN99] R. Baeza Yates and B. Ribeiro Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.

[DCMI07a] Dublin Core Metadata Initiative. Dublin Core Metadata Initiative, 2007. `http://dublincore.org/`.

[DCMI07b] Dublin Core Metadata Initiative. Expressing Simple Dublin Core in rdf/xml, 2007. `http://www.dublincore.org/documents/dcmes-xml/`.

[FWE03] Benjamin C. Fung, Ke Wang, and Martin Ester. Hierarchical Document Clustering Using Frequent Itemsets. In *SIAM International Conference on Data Mining (SDM'03)*, pages 59–79, San Francisco, CA, May 2003.

[Gle01] Fung Glenn. A Comprehensive Overview of Basic Clustering Algorithms. Disponible en `http://pages.cs.wisc.edu/~gfung/clustering.pdf`, 2001.

[Kar03] George Karypis. *CLUTO: A Clustering Toolkit*, 2003.

[Kar06] George Karypis. Karypis Lab, 2006. `http://glaros.dtc.umn.edu/gkhome/views/cluto/`.

[LLB03] Martín Lopera, Ortega Lobo, and William Branch. Agrupamiento de Resultados Obtenidos de Búsquedas Hechas sobre la Web para un Catálogo de Acceso Público en Línea. *Dyna, Universidad Nacional de Colombia*, 2003. Disponible en `http://redalyc.uaemex.mx/redalyc/src/inicio/ArtPdfRed.jsp?iCve=49614206`.

[LVdS01] C. Lagoze and H. Van de Sompel. The Open Archives Initiative: Building a low-barrier interoperability framework. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries, (JCDL'01)*, pages 24–28,54–62, Roanoke, Virginia, United States, June 2001.

[MN02] M. F. Maldonado Naude. Hermes: Servidor y Biblioteca de Modelos de Recuperación de Información, 2002. Tesis. Universidad de las Américas de Puebla - UDLAP, 2002. Disponible en `http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/maldonado_n_mf/`.

[MNSBY03] M. F. Maldonado Naude, Alfredo Sánchez, and R. Baeza Yates. Using Hermes-f: Experiences with a Framework for Developing Information Retrieval Applications. In *Fourth Mexican International Conference on Computer Science, ENC 2003*, pages 101–108, Tlaxcala, México, September 2003.

[MS03] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 2003.

[MSR06] María A. Medina, Alfredo Sánchez, and Adriana G. Ramírez. Describing Document Hierarchies by Using Markup Languages. In *Proceedings of the Mexican Internacional Conference on Computer Science, ENC 06*, San Luis Potosí, México, September 2006.

[SKK00] Michael Steinbach, George Karypis, and Vipin Kumar. *A Comparison of Document Clustering Techniques*, 2000. Disponible en `http://www.cs.umn.edu/research/technical_reports.php?page=report&report_id=00-034`.

[VdSL02] H. Van de Sompel and C. Lagoze. Notes from the Interoperability front: a Progress Report on the Open Archives Initiative. In *6th European Conference on Research and Advanced Technology for Digital Libraries*, pages 144–157, Rome, Italy, September 2002.

[VdSL07] H. Van de Sompel and C. Lagoze. The oai-pmh, 2007. `http://www.openarchives.org/OAI/openarchivesprotocol.html`.

[Zai06] Osmar Zaiane. Data Clustering, 2006. `http://www.cs.ualberta.ca/~zaiane/courses/cmput690/slides/Chapter8/index.htm`.

[ZK01] Ying Zhao and George Karypis. Criterion Functions for Document Clustering: Experiments and Analysis. In *UMN CS 01-040*, 2001.

[ZK05] Ying Zhao and George Karypis. Hierarchical Clustering Algorithms for Document Datasets. In *Data Mining and Knowledge Discovery, Vol. 10, No. 2*, pages 141–168, 2005.