Universidad de las Ciencias Informáticas

Facultad 1

"Empleo de las Técnicas Star para la Recuperación de la Información"

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas.

Autor:

Ricardo González Chang.

Tutores:

Ing. Emilio Suri López.

Ing. Yirianni Rivero Escalona.

Declaración de Autoría

Declaro por este medio que yo, Ricardo González Chang con carnet de identidad 89091442020, soy el autor de este trabajo y que autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Para que así conste, firma la presente declaración jurada de autoría en La Habana a los ____ días del mes _____ del año _____.

Ricardo González Chang	Emilio Suri López	Yirianni Escalona Rivero
Autor	**Tutor**	**Tutor**

AGRADECIMIENTOS

A mi familia por apoyarme siempre en mis decisiones, por ayudarme en todo lo que pudieron durante el transcurso de la carrera.

A todas las personas que ayudaron de una forma u otra la realización del presente trabajo de diploma.

A todos los amigos que hice durante los 5 años de la carrera.

A mis compañeros de aula y proyecto.

A mis tutores Yirianni y Emilio.

Al profesor Maikel Fernández.

A los compañeros del apartamento.

A Hector (El Puro), por ayudarme siempre que estaba en problemas con alguna asignatura y con la realización de la tesis.

DEDICATORIA

Dedico el presente trabajo:

A mi mamá y mi papá, por su apoyo en todo momento, por su cariño.

A toda mi familia, que siempre estuvieron para mí, por sus consejos, por todo.

Resumen

El presente trabajo de diploma se centra en la construcción de un módulo que permite agrupar los documentos de una base de datos teniendo en cuenta la semejanza que exista entre estos, con el objetivo de mejorar la precisión en el proceso de búsquedas de información en los sistemas de recuperación de información.

Para lo cual se realizó un estudio de las técnicas *Star* para el agrupamiento de documentos, analizándose las características, ventajas y desventajas presentes en cada uno de los algoritmos que componen a dichas técnicas.

Para la construcción del módulo se seleccionó el algoritmo *Extended Star*, basándose en la arquitectura de Drupal 7. Se utilizaron un conjunto de tecnologías y lenguajes que permitieron un correcto desarrollo de la solución implementada.

Se realizaron pruebas de precisión, exhaustividad sobre el módulo implementado, analizándose el comportamiento del mismo a través de los valores arrojados por estas medidas. También se hace una valoración del tiempo de ejecución de la solución y se determinan los factores que influyen en esta.

Palabras claves: Recuperación de Información, algoritmo, agrupamiento, Técnicas *Star*, *Extended Star*.

Índice

Índice de Ilustraciones

Índice de tablas

INTRODUCCIÓN

La información ha sido un elemento clave para el desarrollo del hombre a lo largo de su existencia. Llegando a ser en la actualidad de vital importancia para el desarrollo de sus actividades. Debido al acelerado desarrollo de las Tecnologías de la Información y las Comunicaciones, al aumento significativo de instituciones dedicadas a la investigación y desarrollo de productos y servicios informáticos, la generación de diferentes tipos de información ha crecido considerablemente al igual que su consumo.

Para facilitar a los usuarios el acceso a la información, son implementados los Sistemas de Recuperación de Información (SRI). Estos sistemas tienen almacenado un conjunto de entes de información, y procesan consultas realizadas por los usuarios permitiendo el acceso al contenido que estos necesiten para su satisfacción.

Los SRI emplean diversas herramientas y algoritmos con el objetivo de lograr mejores resultados en la realización de las búsquedas de documentos, o sea, obtener resultados con alto grado de precisión y en un tiempo adecuado. Por lo cual, una de las actividades en las que se ha enfocado la recuperación de información es en el estudio y aplicación de la categorización automática de documentos o agrupamiento de documentos.

El agrupamiento de documentos electrónicos se investiga dentro del campo de la recuperación de información como una herramienta capaz de mejorar la calidad de los sistemas que la empleen.

Existen varios algoritmos orientados a la agrupación de documentos, un ejemplo de estos son el grupo de algoritmos o técnicas *Star*, las cuales han sido utilizadas en varias aplicaciones de filtrado y organización de la información.

Las técnicas *Star* están compuestas por un conjunto de algoritmos, siendo estos, algoritmo *Star*, *Extended Star*, *Generalized Star*, *CStar y CStar+*.

Se puede afirmar que no existe un Sistema de Recuperación de Información que sea completamente exacto, algunos cumplen con las expectativas del usuario, mientras que otros se quedan por debajo.

Este problema se debe a que la implementación de estos sistemas se realiza de diferentes formas, cada cual respondiendo a sus necesidades e intereses, aplicando diferentes técnicas de búsqueda. Por ejemplo los SRI basados en la arquitectura de Drupal, que aprovechan el sistema de búsqueda que provee dicho gestor de contenidos, el cual indexa todos los contenidos del sitio web, realizando las búsquedas sobre estos.

La exactitud de estos sistemas se ve afectada debido a que no implementan técnicas para el agrupamiento de documentos en el proceso de recuperación de la información, provocando que las búsquedas de información por parte del usuario se vuelvan engorrosas, debido a la falta de precisión en las respuestas dadas por el sistema, documentos que son irrelevantes son mostrados como resultado de la búsqueda. Además de consumir un tiempo no adecuado para obtener los resultados esperados.

Por lo que se definió el siguiente **problema de investigación**: ¿Cómo utilizar el agrupamiento de documentos para mejorar el proceso de recuperación de información?

Para darle solución al problema planteado se define como **objetivo general**, Implementar un módulo utilizando la arquitectura de Drupal 7 para aplicar las técnicas de agrupamiento *Star* en la recuperación de la información.

Determinándose como **objeto de estudio** las técnicas de agrupamiento *Star*, y como **campo de acción** las técnicas de agrupamiento *Star* para la recuperación de información.

Objetivos específicos:

- Describir los aspectos teóricos relacionados con las técnicas de agrupamiento *Star.*
- Desarrollar un módulo que posibilite la aplicación de algoritmos de agrupamiento *Star* para la recuperación de información.
- Realizar las pruebas de precisión, exhaustividad y rendimiento al módulo implementado.
- Analizar el comportamiento del módulo implementado.

Tareas de investigación:

1. Estudio de las técnicas de agrupamiento *Star* para la recuperación de información.
2. Selección de la técnica a utilizar en el módulo propuesto.
3. Descripción del módulo propuesto y de cada una de sus componentes.
4. Implementación del módulo propuesto.
5. Realización de las pruebas de precisión, exhaustividad y rendimiento al módulo implementado.

Pregunta de investigación:

- ✓ ¿Se podrá mejorar la precisión en la realización de las búsquedas en los SRI implementando las técnicas *Star* para el agrupamiento de documentos?

Métodos de investigación:

Métodos teóricos:

Histórico-lógico: Permitió hacer un estudio de los fundamentos teóricos de las técnicas de agrupamiento *Star*, su repercusión nacional e internacional, así como sus tendencias actuales.

Analítico-sintético: Se utilizó para el análisis del funcionamiento de las técnicas de agrupamiento *Star*, y obtener el conocimiento necesario para la implementación de una solución para el problema planteado.

Métodos empíricos:

Observación: Permitió realizar un estudio de diferentes SRI, analizando su comportamiento, las soluciones que presenta, así como los resultados.

Encuesta: Se empleó con el objetivo de realizar una valoración de los resultados obtenidos con la aplicación del módulo implementado.

Justificación de la investigación:

La presente investigación se realiza debido a que existen Sistemas de Recuperación de Información que no emplean algoritmos para el agrupamiento de la información en la realización de sus búsquedas. Por lo cual, se hace un necesario estudio de las técnicas de agrupamiento *Star*, para desarrollar un módulo en Drupal 7 que implemente dichas técnicas, y así lograr que las búsquedas de información se realicen de forma más precisa y en un tiempo adecuado para el usuario.

Para una mejor organización y entendimiento del contenido del presente documento, el mismo se estructuró de la siguiente forma:

Capítulo 1: Estudio de procedimientos y tecnologías relacionados con la recuperación de información: Se realizó un estudio de los elementos y conceptos relacionados con la recuperación de información. Se describen cada una de los algoritmos que componen a las técnicas *Star*, y sus aplicaciones en Cuba y el mundo. Además se especifican las herramientas a utilizar en el desarrollo de la solución propuesta.

Capítulo 2: Características del módulo para el agrupamiento de documentos: Se describieron cada uno de los elementos de la solución propuesta. Se definen los requerimientos a implementar, arquitectura y otros elementos que permiten una mayor organización para la construcción del módulo.

Capítulo 3: Implementación y Pruebas del módulo para el agrupamiento de documentos: Se describieron los componentes fundamentales de la implementación. De igual forma se describe el proceso de pruebas del módulo implementado.

CAPÍTULO 1 ESTUDIO DE PROCEDIMIENTOS Y TECNOLOGÍAS RELACIONADOS CON LA RECUPERACIÓN DE INFORMACIÓN.

1.1 Introducción

En el presente capítulo se realiza un estudio de los conceptos asociados a la Recuperación de Información (RI), para lograr un mejor entendimiento del tema que se aborda.

Se hace una descripción de cada uno de los algoritmos que componen a las técnicas *Star* para el agrupamiento de documentos, y de los conceptos relacionados con los mismos para entender, de forma teórica, el funcionamiento de estos.

Se realiza un estudio de aplicaciones y trabajos donde se hace uso de estos algoritmos, se analizan las características, bondades y deficiencias de los mismos para definir cuál es el que va a ser usado en el módulo a implementar.

Por último se especifican las herramientas, lenguajes y metodología a utilizar para el desarrollo de la solución propuesta.

1.2 Recuperación de la información

La Recuperación de Información es un término que no es nuevo y en la actualidad juega un papel muy importante debido al valor que la misma posee. La información es una necesidad de las personas a diario y es necesario brindar la misma de una manera rápida y precisa, siendo este uno de los principales objetivos de esta rama(1).

Según Ricardo Baeza-Yates (2), el cual ha sido muy referenciado en trabajos relacionados con el tema, "*la Recuperación de Información trata con la representación, el almacenamiento, la organización y el acceso a ítems de información*".

En resumen la Recuperación de Información es una rama que tiene que ver con todo lo relacionado con el almacenamiento de diferentes tipos de información, la cual es clasificada y organizada con el objetivo de facilitar a las personas el acceso a esta.

1.2.1 Hipótesis de agrupamiento

En el proceso de recuperación de información uno de los principales objetivos es la obtención de la documentación con precisión y eficiencia, por lo que dicha rama investiga y aplica la clasificación de los documentos. La hipótesis de agrupamiento en resumen plantea que, en las búsquedas de información, aquellos documentos asociados entre ellos, existen grandes posibilidades de que sean relevantes a la misma consulta realizada.

La categorización de documentos dentro de la recuperación está destinada principalmente a mejorar la calidad de los sistemas que la implementan.

Sus principales aplicaciones son:

- *"Mejorar el rendimiento de los motores de búsqueda de información mediante la categorización previa de todos los documentos disponibles."*
- *"Facilitar la revisión de los resultados por parte del usuario final, agrupando los resultados luego de realizar la búsqueda*(3)*."*

De forma general se mejora la recuperación, si se recuperan no solo los documentos que sean relevantes, sino también los documentos similares a ellos (los que pertenecen al mismo grupo).

Precisión: Representa la cercanía de los valores medidos. Proximidad de concordancia entre valores medidos obtenida por mediciones repetidas de un mismo objeto, o de objetos similares, bajo condiciones especificadas, en este caso documentos electrónicos(4).

1.2.2 Sistemas de Recuperación de Información

Los Sistemas de Recuperación de Información son una clase de sistemas de información que tratan con bases de datos compuestas por documentos y procesan las consultas de los usuarios permitiendo el acceso a la información relevante en un intervalo de tiempo apropiado(5).

Estos sistemas para la obtención de los diferentes tipos de información se apoyan en la implementación de varias técnicas y algoritmos de búsqueda, dentro de los cuales se encuentran los algoritmos de agrupamiento de documentos, que se han ido perfeccionando debido al

aumento de la información y a las necesidades de los usuarios. Un ejemplo de esto se refleja en las técnicas de agrupamiento *Star* para la recuperación de información.

1.3 Técnicas de agrupamiento Star

Están compuestas por un conjunto de algoritmos basados en grafos, los cuales se identifican por la creación de sub-grafos en forma de estrella, de aquí su nombre.

A continuación se realiza una descripción de los algoritmos que componen a dichas técnicas, especificando sus principales características, definiciones, bondades y deficiencias que presentan cada uno de ellos. Para lo mismo se necesitan tener conocimientos de los siguientes conceptos:

1.3.1 Función de semejanza

Se denomina función de semejanza *w* a una función que asocia a cada par de objetos de un universo de objetos $U = \{O_1, \dots, O_n\}$ una magnitud que evalúa su semejanza o parecido(6).

1.3.2 Grafo de semejanza

Se llama grafo de semejanzas *G = (V, E, w)*, al grafo completo donde los vértices *V* son los objetos a agrupar y las aristas se etiquetan con las semejanzas entre los objetos *E*, calculada por una función de semejanza *w*.

1.3.3 Objeto *β*-semejantes

Dos objetos cuya semejanza es mayor o igual que un cierto umbral *β* (definido por el usuario) se denominan *β*-semejantes. Si un objeto no es *β*-semejante con ningún otro objeto se denomina *β*-aislado(6).

1.3.4 Grafo de β-semejanza

Un grafo de *β*-semejanza se denota $G_\beta = (V, E_\beta)$, el cual es un sub-grafo del grafo de semejanzas, donde se eliminan las aristas con peso menor que *β*, donde solamente quedan conectados los objetos semejantes(6; 7).

1.3.5 Algoritmo *Star*

Desarrollado por Javed Aslam, el cual se basa en la construcción de un grafo de semejanza G_β cuyos vértices representan a los documentos. De este grafo se obtienen los documentos estrellas o centros de clústeres que son los vértices del grafo que tengan mayor cantidad de aristas y el resto de los vértices son considerados satélites de estas estrellas(8).

Un sub-grafo en forma de estrella, es un sub-grafo de m + 1 vértices, en el cual existe un vértice llamado “centro", m vértices denominados “satélites" y se cumple que:

- ✓ El centro tiene un grado mayor o igual que el resto de los vértices del sub-grafo.
- ✓ Existe una arista del centro a cada uno de los satélites.

El problema de encontrar los sub-grafos en forma de estrella se reduce al problema de determinar el conjunto X de vértices centro(7).

Este algoritmo presenta dos deficiencias significativas, siendo la primera de estas que el resultado de la agrupación está en dependencia del orden en que se realice el análisis de los vértices del grafo.

Y como segunda deficiencia es que independientemente del orden en que se realice el análisis de los vértices, se obtienen grupos “ilógicos”. Un grupo g_1 se considera ilógico si cumple las siguientes condiciones(7):

- Existe un elemento *e* que pertenece a g_i que es más denso que el vértice centro *c* que define a g_1.
- El elemento *e* puede agrupar, si se considera como centro, a los vértices que son agrupados solo por el centro *c*.

Estas condiciones vienen dadas debido a que el algoritmo *Star* no permite que dos vértices adyacentes sean centros.

1.3.6 Algoritmo *Extended Star*

El algoritmo *Extended Star*, en español Estrella Extendida, fue desarrollado por Reynaldo J. Gil García(9), para dar solución a las deficiencias del algoritmo *Star*. Dicho algoritmo presenta las mismas características de su antecesor, pero realizando una nueva definición del vértice centro, el cual obtiene un conjunto diferente de estos y por lo tanto, un conjunto diferente de grupos.

Este algoritmo plantea que, para ser vértice centro *v* debe tener un vecino o adyacente v´ el cual cumple con lo siguiente:

- *v´* no tiene vértices centros adyacentes
- Si se considera a *v´´* como el vértice centro de mayor grado que es adyacente a *v´*, entonces se cumple que *v´´* tiene un grado mayor al de *v.*

Este algoritmo para determinar el conjunto de vértices que definen el agrupamiento además de utilizar el grado de los vértices trabaja con el grado complemento de los mismos. El grado complemento de un vértice *v* es la cantidad de adyacentes que tiene *v* que no pertenecen a ninguno de los grupos formados a partir de los vértices anteriormente incluidos en el conjunto.

Sin embargo, el algoritmo *Extended Star* presenta deficiencias, y es que se pueden obtener grupos redundantes, esto se debe a que este algoritmo permite que más de un vértice que cumpla con las condiciones para ser centro sea considerado como tal(7).

1.3.7 Algoritmo *Generalized Star*

Propuesto por Ariel Pérez Suárez y José E. Medina, donde, manteniendo el planteamiento de Javed Aslam(10) de que, el cubrimiento del grafo G_β a través de sub-grafos en forma de estrella: permite obtener grupos con una semejanza relativamente alta entre los documentos que lo componen. Desarrollaron un nuevo concepto de sub-grafo en forma de estrella, posibilitando a este algoritmo a partir del cubrimiento del grafo con la nueva definición de sub-grafo, construir un conjunto de grupos que pueden ser solapados.

Star, para la definición del sub-grafo utiliza solamente el grado de los vértices, mientras que el algoritmo *Generalized Star* define un grupo de conjuntos para cada uno de los vértices del grafo,

el cual apoyado en estos define el sub-grafo en forma de estrella generalizada (EG), además de una heurística que define como sería el agrupamiento que se obtenga(7).

Los conjuntos de Satélites Débiles (SD) y Satélites Potenciales (SP) de un vértice v, se definen por las siguientes expresiones:

$$v.SD = \{s \in v.Ady || v.Ady| \geq |s.Ady|\}. (1)$$

$$v.SP = \{s \in v.Ady || v.SD| \geq |s.SD|\}. (2)$$

El grado SD y SP de un vértice v, se define como la cardinalidad de los conjuntos de satélites débiles y potenciales de v respectivamente.

Teniendo en cuenta las definiciones anteriores, un sub-grafo en forma de estrella generalizada (sub-grafo EG) se define como un sub-grafo de m + 1 vértices, en el cual existe un vértice c denominado “centro" y m vértices llamados “satélites", cumpliéndose que existe una arista entre cada satélite y el centro satisface la siguiente expresión:

$$\forall s \in c.SP |c.SP| \geq |s.SP| (3)$$

El algoritmo *Generalized Star* posee algunas ventajas sobre sus predecesores, dando solución a algunas deficiencias del los mismos, no produce grupos ilógicos ni grupos redundantes.

Pero también es importante tener en cuenta las deficiencias de este algoritmo. La primera de ellas es que elimina grupos densos previamente encontrados, lo cual puede afectar la calidad del agrupamiento realizado.

Otra deficiencia está relacionada con el consumo de memoria por este algoritmo, y es que el mismo necesita calcular varios conjuntos para cada vértice, siendo estos los adyacentes, satélites débiles y satélites potentes, los cuales tiene que mantenerse almacenados en memoria hasta el final del algoritmo, pudiendo llegar a ser ineficiente con grandes colecciones de documentos(7).

1.3.8 Algoritmo *CStar*

El algoritmo *CStar* introduce una nueva definición de sub-grafo, el cual es nombrado "*sub-grafo en forma de estrella condensada"*(7). Con este algoritmo se obtienen grupos que pueden tener traslape, manteniendo los puntos fuertes de sus predecesores y trabajando sobre las deficiencias anteriores.

La idea principal del algoritmo *CStar* es determinar un criterio que establezca cuándo un sub-grafo del tipo estrella condensada (EC) es más denso que otro y partiendo de éste, realizar un cubrimiento del grafo de β-semejanza utilizando los sub-grafos EC más densos y posteriormente aplicar un proceso de filtrado que reduzca la cantidad de éstos.

Un problema que presenta este algoritmo es que puede obtener diferentes agrupamientos cuando se ejecutan sobre una misma colección, debido esto a que existe una dependencia del orden de análisis de los documentos entre otras características de este algoritmo(7).

1.3.9 Algoritmo *CStar+*

Se describe como una variante de su antecesor *CStar*. Este algoritmo utiliza sub-grafos EC para realizar un cubrimiento sobre las componentes conexas del grafo de β-semejanza. Transformando el problema de determinar un agrupamiento de G_β usando sub-grafos EC en el problema de realizar un cubrimiento utilizando sub-grafos EC de cada componente conexa.

Es importante tener en cuenta que aunque obtener un cubrimiento de estas componentes a través de sub-grafos EC reduce el encadenamiento, también podría afectar la calidad del agrupamiento si dicha componente tiene un alto grado de conexión entre sus vértices, pues se estaría dividiendo en sub-grupos un grupo que ya es altamente cohesionado.

Este algoritmo también presenta el problema de su antecesor de que, se pueden obtener diferentes agrupamiento si se aplican en una misma colección de documentos(7).

Después del estudio realizado de cada uno de los algoritmos que componen a las técnicas *Star*, se definió para la implementación del módulo propuesto la utilización del algoritmo *Extended Star*.

Se seleccionó el algoritmo *Extended Star* debido a las siguientes razones:

- ✓ Primeramente tiene como objetivo dar solución a las deficiencias de su antecesor *Star*, eliminando el problema de que, en dependencia del orden en que se analizaran los vértices del grafo, era la calidad de los resultados obtenidos. Además elimina la posibilidad de obtener grupos ilógicos.
- ✓ En las aplicaciones donde se ha utilizado *Extended Star* se han obtenido excelentes resultados, evidenciando la precisión y similitud global de las respuestas dadas por el mismo.
- ✓ A diferencia de otros, este algoritmo mantiene un consumo de memoria adecuado, debido a que es relativamente fácil de implementar, no tiene una gran cantidad de funciones asociadas que se tengan que construir para el funcionamiento del mismo. Un algoritmo que presenta esta deficiencia es el *Generalized Star*, el cual depende de una gran cantidad de cálculos para su funcionamiento. Este último además presenta como desventaja que, elimina grupos densos obtenidos, lo cual afecta considerablemente la calidad de sus resultados.
- ✓ Otros de los problemas que *Extended Star* soluciona es que, los resultados obtenidos siempre van a ser los mismos sin importar la cantidad de veces que se aplique este algoritmo a una misma colección de documentos. Esta deficiencia está presente en los algoritmos *CStar* y su variante *CStar+*, la cual está dada por la dependencia del orden en que se realice el análisis de los documentos, presente también en *Star*.

1.4 Aplicaciones del algoritmo *Extended Star*

En el siguiente epígrafe se describirán aplicaciones y trabajos en las cuales ha sido utilizado el algoritmo *Extended Star*, donde se evidencian las bondades que el mismo brinda sobre la recuperación de la información.

1.4.1 Herramientas para un observatorio de información semi-automatizado

Herramientas desarrolladas en la Facultad Regional de Granma, por Roberto Oscar Labrada, la cual está encaminada a dar solución a un conjunto de deficiencias presentes en los observatorios de información de internet, los cuales deben cubrir las siguientes actividades:

1. La captura de información, en la cual el observatorio obtiene la información en su estado "natural".
2. El filtrado de información, que se encarga de filtrar la información de interés de aquella que no lo es.
3. El análisis de la información relevante.

Las deficiencias están dadas por la ausencia de un sistema semi-automatizado el cual debería(11):

- ✓ Monitorear de forma automática, en la frecuencia requerida y con profundidad deseada de los distintos sitios web.
- ✓ Realizar resúmenes.
- ✓ Agrupar la información de acuerdo a la similitud de los artículos.

Una de las actividades de los observatorios de información es la detección de temas que se repitan en varias fuentes de información observadas, para lo cual es necesaria la agrupación de documentos por su contenido.

Para la agrupación de textos, después de haber hecho una comparación entre varios algoritmos de clasificación de documentos, este sistema utilizó el algoritmo de agrupamiento ***Extended Star***, el cual se adecuaba a las necesidades del trabajo.

Fases cubiertas por el sistema(11):

1. Obtención de la información desde las fuentes.
2. Filtrar las páginas que contienen información relevante.
3. Agrupar la información obtenida por su contenido.
4. Visualizar los grupos de contenido, lo que facilita a los trabajadores del observatorio:
    - La detección rápida de temas "calientes", estos serían los grupos más voluminosos.
    - La detección de duplicados: Dos documentos idénticos estarán en el mismo clúster.

1.4.2 *CorpusMiner* 1.0: Herramienta para el agrupamiento de documentos

Herramienta construida por Leticia Arco García, para la extracción de resúmenes, categorización, clasificación y verificación de homogeneidad de un corpus textual(12).

Al trabajar con corpus textuales se puede realizar un análisis léxico, sintáctico o semántico, o combinaciones de estos. *CorpusMiner* realiza un análisis léxico, es decir, se sigue la idea de analizar el corpus como una bolsa de palabras, donde únicamente se tiene en cuenta la frecuencia de aparición de los términos en los documentos.

Esta herramienta consta con cuatro módulos:

1. Transformar.
2. Extraer términos y representar espacio vectorial.
3. Seleccionar los rasgos.
4. Agrupar los documentos.

Para la construcción del cuarto módulo se implementaron tres algoritmos de agrupamiento, una extensión del algoritmo *Star (Extended Star)*, el algoritmo *SKWIC* y *Fuzzy SKWIC.* El primero mencionado constituyendo el método interno del proceso de agrupamiento de documentos para la herramienta, entendiéndose por método interno aquel que inicia dicho proceso, siendo el algoritmo de mayor valor para el módulo.

En las pruebas realizadas a esta herramienta se determinó que el algoritmo *Extended Star* logra los mejores valores de precisión, entropía y similitud global(12).

1.5 Herramientas y lenguajes.

Para el desarrollo de la propuesta de solución se necesita la utilización de un conjunto de tecnologías, lenguajes, plataformas y sistemas de gestión de bases de datos, las cuales fueron seleccionadas teniendo en cuenta el alcance de sus prestaciones y las características de la solución a desarrollar.

A continuación se hace una descripción de las herramientas y lenguajes a utilizar, especificándose sus principales características y bondades aportadas por las cuales fueron seleccionadas.

1.5.1 Plataforma NetBeans 7.2

NetBeans es un entorno de desarrollo integrado (IDE), modular, de base estándar (normalizado), escrito en el lenguaje de programación Java. El proyecto NetBeans consiste en un IDE de código abierto y una plataforma de investigación, las cuales pueden ser usadas como una estructura de soporte general (framework) para compilar cualquier tipo de aplicación(13).

NetBeans IDE dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones web, control de versiones, colaboración entre varias personas, creación de aplicaciones compatibles con teléfonos móviles, resaltado de sintaxis y por si fuera poco sus funcionalidades son ampliables mediante la instalación de paquetes.

1.5.2 Sistema de Gestión de Contenidos (CMS) Drupal 7.19

Sistema instalado sobre un servidor web que se utiliza para la creación de sitios web dinámicos y con gran variedad de funcionalidades, donde el diseño está separado del contenido, entendiéndose por **contenido** a textos, fotos, imágenes, y **diseño** la forma en que se presenta el contenido, menús, colores y bloques.

Drupal cuenta con un conjunto de módulos que facilitan el trabajo de implementación, los cuales proveen diversas funcionalidades para el desarrollo de las aplicaciones web, además de poder incluir otros módulos existentes en el sitio oficial de Drupal.

Al ser este CMS código abierto permite la modificación del mismo además de que posibilita la creación de módulos por parte de usuarios y comunidades que se dedican a la creación de estos.

1.5.3 Gestor de base de datos MySQL 5.5.24

Sistema gestor de bases de datos relacionales rápido, sólido y flexible. Es idóneo para la creación de bases de datos con acceso desde páginas web dinámicas, así como para la creación

de cualquier otra solución que implique el almacenamiento de datos, debido su velocidad a la hora de realizar las operaciones y al bajo consumo de recursos que este necesita(14).

Características de MySQL(15)

- Está desarrollado en C/C++.
- Se distribuyen ejecutables para cerca de diecinueve plataformas diferentes.
- La API[1] se encuentra disponible en C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL.
- Está optimizado para equipos de múltiples procesadores.
- Es muy destacable su velocidad de respuesta.
- Se puede utilizar como cliente-servidor o incrustado en aplicaciones.
- Cuenta con un rico conjunto de tipos de datos.
- Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes para poder optimizar el SGBD[2] a cada caso concreto, como por ejemplo: Archive, Aria, AWSS3, BDB, Blackhole, Cassandra SE, entre otros.
- Su administración se basa en usuarios y privilegios.
- Sus opciones de conectividad abarcan TCP/IP, Sockets UNIX y Sockets NT, además de soportar completamente ODBC[3].

1.5.4 Servidor web Apache 2.2.22

El servidor web Apache es una tecnología que, gracias a su robustez y estabilidad es usada por millones de sistemas y con un alto grado de confianza. Es una herramienta gratuita, de código abierto, que puede ser usado en varios sistemas operativos.

Es un servidor altamente configurable de diseño modular, lo cual permite ampliar sus capacidades a través de una gran cantidad de módulos existentes para el mismo.

Permite la creación de ficheros de log a medida del administrador, de este modo se puede tener un mayor control sobre lo que sucede en el servidor(16).

[1] API: del inglés *Application Programming Interface*, en español Interfaz de programación de aplicaciones.
[2] SGBD: Sistema Gestor de Bases de Datos.
[3] ODBC: Open Database Connectivity (Estándar de acceso a bases de datos).

1.5.5 Visual Paradigm 8.0

Visual Paradigm para UML es una herramienta CASE[4] que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, implementación y pruebas. Ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite construir diagramas de diversos tipos, código inverso, generar código desde diagramas y generar documentación. La herramienta también proporciona abundantes tutoriales, demostraciones interactivas y proyectos UML(17).

1.5.6 Lenguaje de programación PHP 5.3.13

PHP, acrónimo de *Hypertext Preprocessor*, es un lenguaje de programación interpretado del lado del servidor, el cual fue diseñado para el desarrollo de sitios web dinámicos. Es un lenguaje relativamente fácil de usar, rápido e integrable, el mismo es introducido dentro del código HTML, permite el uso de técnicas de programación orientada a objeto, biblioteca nativa de funciones sumamente amplia e incluida, no requiere definición de tipos de variables y tiene manejo de excepciones.

Es un lenguaje multiplataforma por lo que puede ser usado en diferentes sistemas operativos, al igual que con diferentes servidores web. Por cumplir la condición de ser código abierto, cuenta con el apoyo de un gran grupo de programadores para corregir errores, además de estar actualizándose continuamente con mejoras para ampliar las capacidades del mismo(18; 19).

1.6 Conclusiones parciales

Con el estudio de los aspectos teóricos relacionados con los algoritmos *Star* se logró comprender el funcionamiento de los mismos y se obtuvo un conjunto de conocimientos que facilitan el desarrollo del módulo propuesto.

El análisis de cada uno de las características y deficiencias de los algoritmos *Star*: permitió la selección del algoritmo *Extended Star* como candidato para aplicarse en la solución propuesta por presentar características favorables para la recuperación de información y ventajas superiores a sus homólogos.

[4] CASE: *Computer Aided Software Engineering*, en español Ingeniería de Software Asistida por Computadora.

El estudio de aplicaciones donde se usa el algoritmo *Extended Star* permite entender su funcionamiento, así como conocer resultados obtenidos por la aplicación del mismo.

CAPÍTULO 2 CARACTERÍSTICAS DEL MÓDULO PARA EL AGRUPAMIENTO DE DOCUMENTOS.

2.1 Introducción

En el presente capítulo se describe la propuesta de solución así como los pasos para la construcción de la misma. Se expone el modelo de dominio, se definen los requerimientos a implementar y la arquitectura por la cual se regirá la construcción de la solución, dando estructura y una adecuada organización.

2.1.1 Descripción general de la propuesta de solución

La propuesta de solución constituye la creación de un módulo utilizando la arquitectura de Drupal 7: el cual aplicará el algoritmo *Extended Star* para el agrupamiento de documentos de una base de datos determinada, con el objetivo de mejorar la recuperación de la información.

El módulo propuesto, aplicando el algoritmo *Extended Star* toma los documentos indexados de una colección, y realizar una clasificación de los mismos dependiendo de la similitud que exista entre ellos.

Para lo cual se deberá primeramente, implementar una función de semejanza que determinará la misma entre cada par de objetos (documentos). Existen varias funciones para determinar el valor de similitud entre los documentos, entre las más usadas se encuentran los coeficientes de Jaccard, de Dice y el coeficiente del coseno, resultando ser las de mejores resultados(8).

Para la implementación del módulo propuesto se utilizará el coeficiente del coseno, debido a que ha demostrado mejor eficacia en diferentes problemas relacionados con la recuperación de información(20).

Partiendo del resultado de la similitud entre cada documento se construye el grafo de semejanza: donde estarán incluidos todos los elementos de la colección, siendo el peso de las aristas el valor de similitud.

Luego se construye el grafo de β-semejanza en el cual se eliminarán las aristas que su peso sea menor que β, o sea que solamente estarán conectados los objetos que sean β-semejantes.

Aquellos vértices que no sean β-semejantes, o también nombrados vértices aislados son insertados en la lista *X*, donde estarán los centros con sus satélites.

A continuación se insertan en una lista *L* los vértices β-semejantes del grafo.

Digamos que *M* es un subconjunto del grafo en el cual se guardarán los vértices con mayor grado complemento, que no son más que aquellos que posean la mayor cantidad de adyacentes que no pertenezcan a ninguno de los grupos formados a partir de los vértices insertados en el conjunto *X* hasta el momento. Y *M´* un subconjunto de *M* donde estarán los vértices de mayor grado.

Los elementos de *M´* son analizados y los que cumplan las condiciones para ser centros y formen un grupo diferente a los determinados hasta el momento son insertados en *X*. Luego se eliminan los elementos restantes en *M´* de la lista *L*. Después se actualiza el grado complemento de los vértices que quedaron en *L*, y por último se repite el proceso hasta que los elementos estén agrupados.

2.2 Modelo de dominio

A continuación se muestra el modelo de dominio que determina el entorno de la solución a implementar, donde se describen las clases que interactúan así como la asociación entre estas.

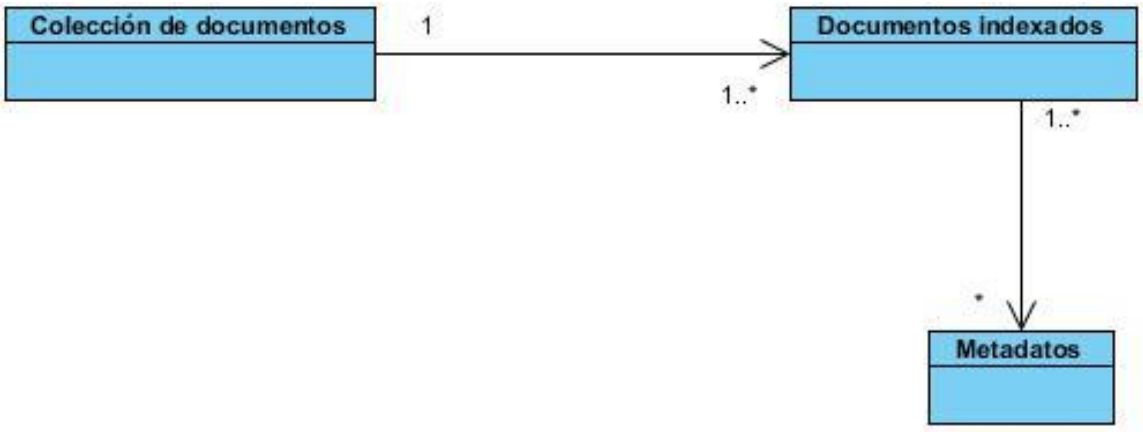


Ilustración 1. Modelo de dominio

2.2.1 Descripción de los términos del dominio.

Metadatos: Elementos principales que describen al documento, por ejemplo, título, autor, descripción y palabras claves.

Colección de documentos: Base de datos donde son almacenados los metadatos de los documentos después de haberse realizado el proceso de indexación.

Documentos indexados: Documentos los cuales han sido organizados, tomando los metadatos pertenecientes a los mismos.

2.3 Requerimientos de la solución

En el presente epígrafe se exponen los requerimientos funcionales que caracterizan a la propuesta de solución, facilitando el proceso de construcción de la misma.

2.3.1 Requerimientos funcionales

El módulo está centrado en un requerimiento el cual se descompone en varias funcionalidades necesarias para su funcionamiento.

RF1: Agrupar documentos

- Calcular similitud entre documentos.
- Construir grafo de semejanza.
- Construir grafo de β-semejanza.
- Calcular vértices de mayor grado complemento.
- Calcular vértices de mayor grado.

2.4 Modelo de Datos

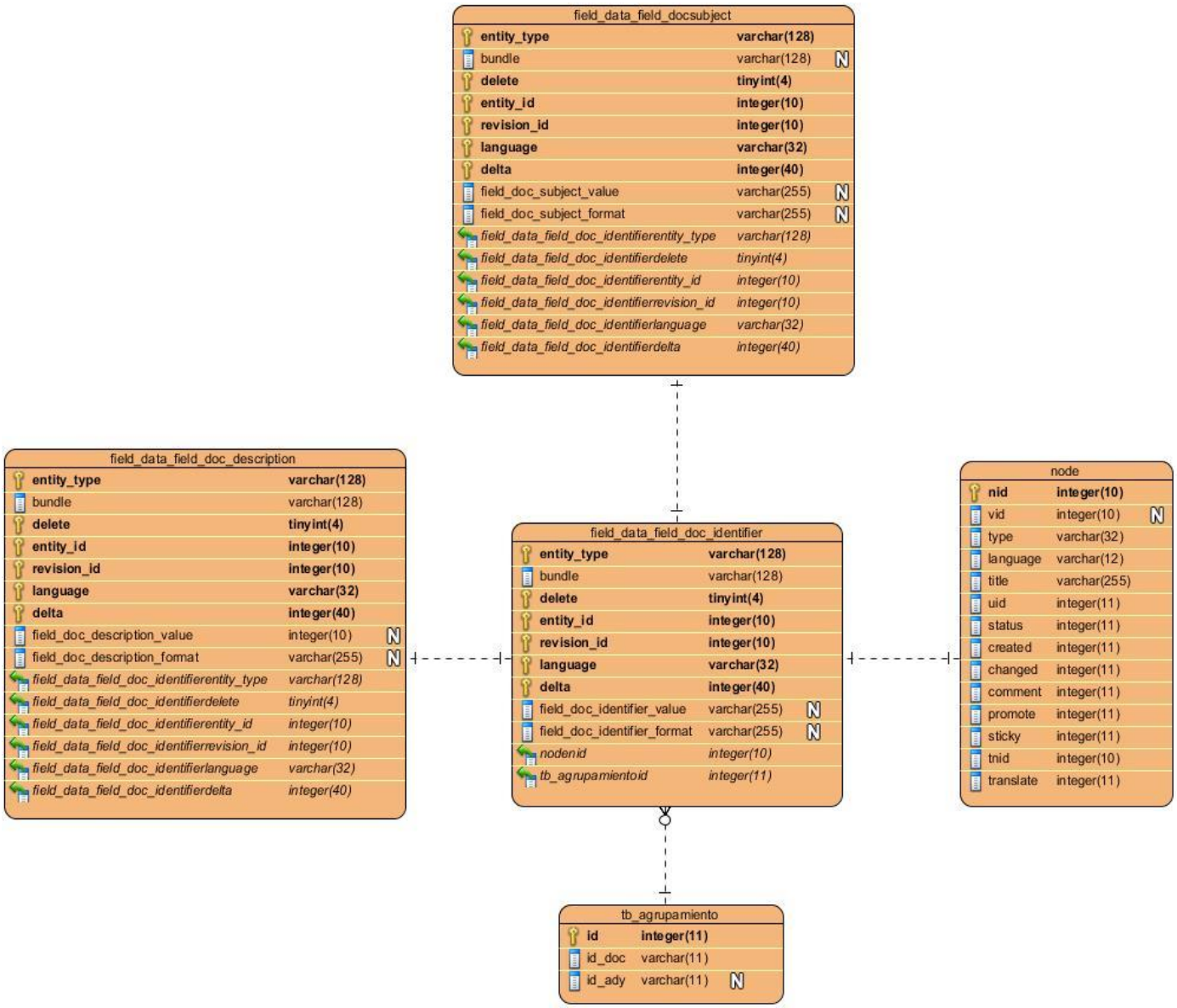


Ilustración 2. Modelo de Datos.

La propuesta de solución crea la tabla “tb_agrupamiento”, la cual almacenará los grupos formados por la aplicación del algoritmo. En esta se guardan los vértices centros en la columna “id_doc” y los vértices satélites (adyacentes) en la columna “id_ady”.

2.5 Arquitectura

La solución a construir va a estar basada en la arquitectura Drupal 7, la cual se caracteriza por ser una arquitectura modular. La misma está formada por el siguiente conjunto de elementos(21):

- **Núcleo:** El núcleo aporta a Drupal la base necesaria para su funcionamiento y para la incorporación del resto de componentes de la arquitectura.
- **Módulos:** Los módulos aportan funcionalidades adicionales al núcleo de Drupal.
- **Área de administración:** El menú de administración se divide en grupos de tareas teniendo inicialmente las siguientes opciones principales: Panel de control, Contenido, Estructura, Apariencia, Personas, Módulos, Configuración, Informes y Ayuda.
- **Nodos y tipos de contenido:** los tipos de contenido en Drupal derivan de un tipo de contenido básico denominado nodo. El tipo de contenido principal es la página básica, que se utiliza para contenidos estáticos del sitio.
- **Entidades y campos:** Las entidades son elementos a los que se pueden añadir campos de información de diferentes tipos (textos, imagen, archivo, número y fecha).
- **Menús:** Facilitan la organización de los nodos publicados. Los menús se pueden colocar en distintas áreas o regiones de un tema y se adaptan al diseño gráfico del sitio.
- **Bloques:** Contenidos principalmente dinámicos que se pueden habilitar en distintas zonas (denominadas regiones) del tema del sitio.
- **Temas:** Define un diseño específico para el sitio web.
- **Usuarios, roles y permisos:** El control de acceso de los usuarios a las distintas funcionalidades del sitio se realiza a través de los roles y permisos.
- **Taxonomías:** Permite la clasificación de los contenidos del sitio.

La arquitectura modular de Drupal 7 permite ampliar sus funcionalidades a través de unos métodos uniformes de desarrollo e integración de nuevos módulos(22). A continuación se muestra en la figura 3 los elementos que componen al CMS Drupal.

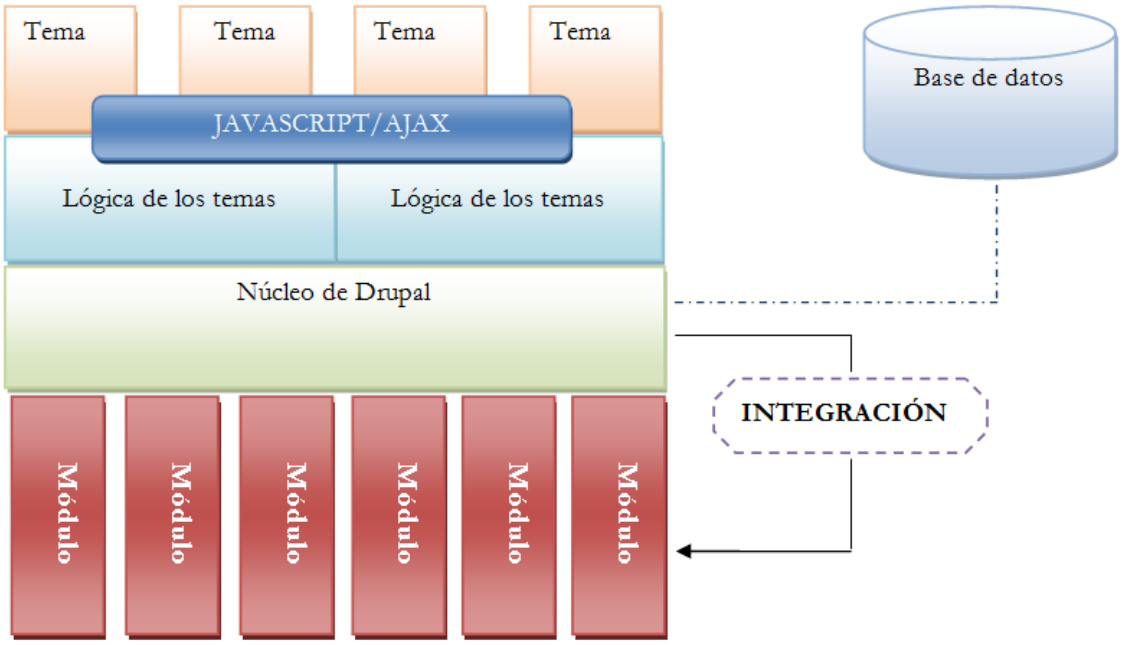


Ilustración 3. Elementos de Drupal.

2.5.1 Arquitectura del módulo Agrupamiento

La arquitectura del módulo propuesto tendrá la siguiente estructura:

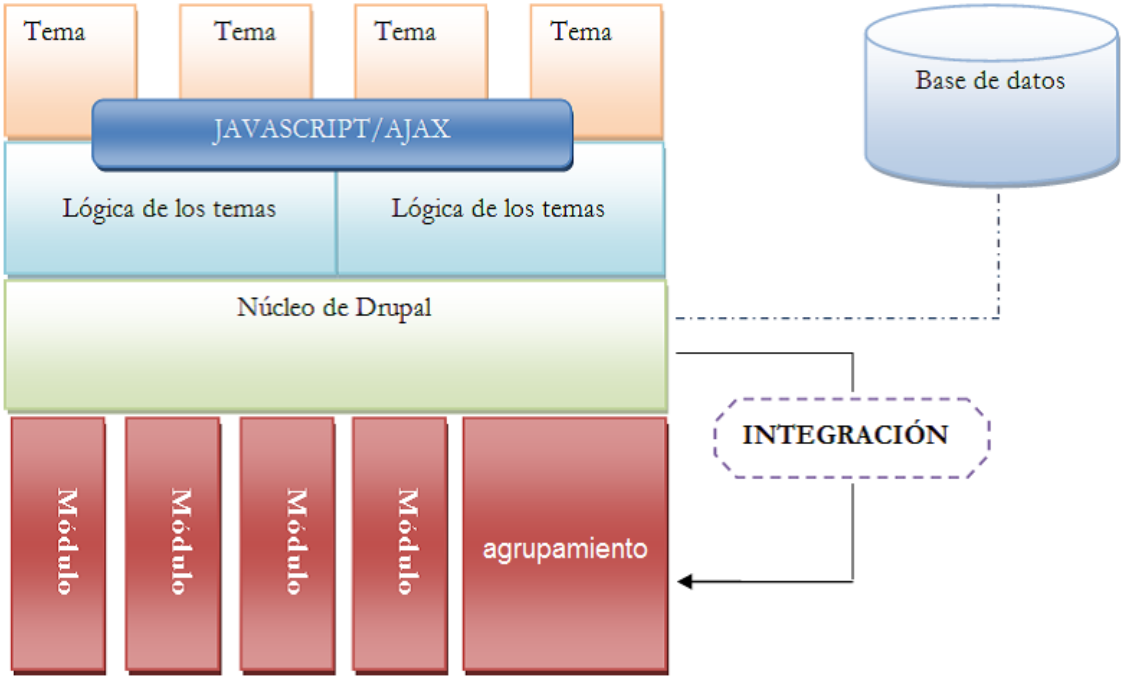


Ilustración 4. Arquitectura del módulo agrupamiento.

2.6 Conclusiones parciales.

La descripción de la propuesta de solución permitió un correcto entendimiento de cada uno de los pasos a seguir para la construcción del módulo, comprendiendo la importancia de cada uno de las funciones a implementar y el orden en que se deben construir.

La definición de requerimientos permite conocer las funcionalidades necesarias para la construcción de la solución propuesta.

CAPÍTULO 3 IMPLEMENTACIÓN Y PRUEBAS DEL MÓDULO.

3.1 Introducción

En el presente capítulo se expondrán los elementos fundamentales para la implementación del módulo propuesto, se describen las fórmulas utilizadas para la implementación del algoritmo y funciones que componen a la solución. También se describen las pruebas realizadas a la solución implementada.

3.2 Implementación del algoritmo *Extended Star.*

A continuación se realiza una descripción detallada de los componentes del algoritmo a implementar, donde se explican las fórmulas y variables que se necesitan para la construcción del mismo.

3.2.1 Función del coeficiente del coseno

El coeficiente del coseno constituye la función utilizada para el cálculo de la similitud para cada par de documentos de la base de datos, la cual se define de la siguiente forma(23):

$$S(d_i, d_j) = \frac{\sum_{h=1}^{k} peso_{ih} * peso_{jh}}{\sqrt{\sum_{h=1}^{k} peso_{ih}^2 * \sum_{h=1}^{k} peso_{jh}^2}} \quad (4)$$

Donde ***di*** y ***dj*** son los documentos a comparar, ***k*** es el número de términos (palabras claves) que caracterizan los documentos, y peso ***xy*** es el peso del término ***y*** en el documento ***x***, calculado teniendo en cuenta la frecuencia de aparición de ese término en el documento(23; 24).

3.2.2 Construcción del grafo de semejanza

El grafo de semejanza se construye a partir de la similitud entre cada par de documentos, donde los vértices representan los documentos y las aristas son etiquetadas con la semejanza existente entre ellos, definiéndose de la siguiente forma:

$$G = (V, E, w) \quad (6)$$

Donde:

G: grafo a construir.

V: vértices (documentos) del grafo.

E: aristas etiquetadas.

w: función que evaluará la semejanza entre cada par de documentos.

3.2.3 Grafo de β-semejanza.

Este grafo se construye a partir del grafo de semejanza que dependiendo de un umbral de semejanza β, se reduce el grafo dejando conectados aquellos documentos más semejantes entre sí. Se define de la siguiente forma:

$$G_\beta = (V, E_\beta) \ (7)$$

Donde:

β: Umbral de semejanza con un valor entre 0-1.

G_β : Grafo a construir.

V: Vértices (documentos) del grafo.

E_β : Aristas etiquetadas cumpliendo que: peso de las aristas sea mayor que *β*.

3.2.4 Cálculo del grado complemento

El grado complemento se calcula para cada vértice del grafo, este se determina por la cantidad de vértices adyacentes que tenga el vértice, los cuales no pertenezcan a ningún grupo formado que sea solución.

$$GC(v) = ady(v) \neq ady(s) \ s \in S \ (8)$$

Donde:

v: vértice del grafo

GC (v): grado complemento del vértice.

ady(v): vértices adyacentes a ***v***.

S: conjunto donde están los vértices solución y sus adyacentes.

s: vértices solución.

ady(s): vértices adyacentes a ***s***.

3.2.5 Cálculo del grado del vértice

Se calcula para todo vértice del grafo y es la cantidad de aristas que estos tengan.

3.2.6 Determinación de vértices centros

La determinación de los vértices centros se realiza partiendo de los vértices de mayor grado complemento y de estos los de mayor grado.

Con la obtención de los vértices centros se construyen los grupos formados por estos y sus adyacentes, quedando los documentos agrupados dependiendo de la similitud que exista entre ellos.

3.2.7 Pseudo-código del algoritmo

La construcción del algoritmo *Extended Star* de forma general, se describe en el siguiente pseudo-código:

Algoritmo *Extended Star:*

Inicio

Entrada: D= $\{d_1; d_2;.....; d_n\}$ – colección de documentos, ***β*** – umbral de semejanza.

Salida: CG – conjunto de grupos.

"Construir $G = (V, E, w)$ desde D**"**

"Construir $G_\beta = (V, E_\beta)$ desde D**"**

CG = *v* que cumplan que: $ady(v) \neq$ ***centro***

L = vértices del grafo ≠ CG.

Calcular grado complemento GC de cada vértice.

Calcular grado complemento GV de cada vértice.

Mientras (exista un *v* no agrupado) hacer:

$M_0 = v$ mayor (*GC*);

$M = v$ mayor (*G*) ∈ M_0;

Para todo vértice $c \in M$ **hacer**:

Si (*c* cumple condición de *centro*)

$C = c$ con $ady(c)$;

Si ($C \notin GC$)

CG = *CG* unión *C*;

Fin Para

Fin Mientras

$L = L \neq M$;

Actualizar *GC* de cada vértice de *L*;

Fin Inicio

3.3 Diagrama de componentes

Los diagramas de componentes muestran las piezas del software que conformarán un sistema, donde estas piezas representan todos los tipos de elementos software que se incluyen en la fabricación de aplicaciones informáticas, por tanto pueden ser simples archivos, paquetes.

A continuación se muestra el diagrama de componentes del módulo desarrollado:

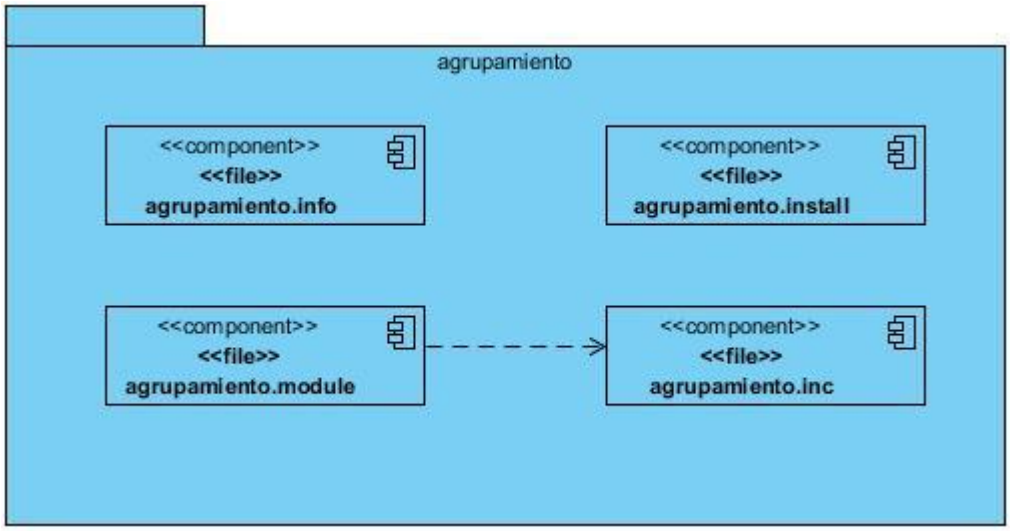


Ilustración 5. Diagrama de componentes del módulo "agrupamiento"

3.3.1 Descripción del diagrama de componentes.

El diagrama de componentes ayuda a comprender como está estructurado el módulo implementado, reflejando los elementos que lo componen. Estos elementos son: el archivo “agrupamiento.info”, que es donde se especifican las características del módulo; “agrupamiento.install”, archivo donde se encuentran los parámetros de instalación; “agrupamiento.module”, en este archivo se implementa la llamada del menú del módulo; “agrupamiento.inc”, aquí se encuentra la implementación del formulario que permite agrupar los documentos, también se implementa las funciones que componen el algoritmo *Extended Star*.

3.4 Pruebas

Para realizar las pruebas al algoritmo implementado se tomó una colección de documentos indexados los cuales están almacenados en una biblioteca desarrollada en Drupal 7. Este sistema obtiene los documentos de diferentes fuentes como: el Repositorio Institucional de la Universidad de las Ciencias Informáticas, Revista de Ingeniería Mecánica CUJAE, Revista Cubana de Ingeniería, ACIMED y Revista Cubana de Ciencias Informáticas.

3.4.1 Pruebas de precisión y exhaustividad*.*

Para la realización de las pruebas de precisión y exhaustividad al algoritmo implementado se tomó una muestra de 100 documentos de la base de datos. Se define "**precisión**" como lo cerca que los valores medidos están unos de otros, en este caso la similitud que exista entre cada uno de los documentos de los grupos formados por el algoritmo. Y "**exhaustividad**" como la proporción entre los documentos semejantes agrupados y los documentos semejantes, esto se resume preguntándose si, en los grupos formados se encuentran todos los documentos que son, o faltan algunos.

Los grupos obtenidos fueron mostrados a 10 personas con conocimientos informáticos y sobre el tema tratado, con el objetivo de que emitieran sus criterios después de un análisis de la semejanza existente entre los documentos que conforman los grupos. De cada documento fueron mostrados los siguientes elementos:

- Título.
- Descripción.

La encuesta realizada a los usuarios se centró en la siguiente pregunta:

- ¿Qué documento no debería estar incluido en el grupo?

Las medidas de precisión y exhaustividad se definen por las siguientes ecuaciones:

$$Precisión = \frac{CDSA}{CDA} \quad (5)$$

Donde:

CDSA: Cantidad de documentos semejantes agrupados por el criterio de los usuarios.

CDA: Cantidad de documentos del grupo analizado resultante de la aplicación del algoritmo *Extended Star*.

$$Exhaustividad = \frac{CDSA}{CDS} \quad (6)$$

Donde:

CDSA: Cantidad de documentos semejantes agrupados por el criterio de los usuarios.

CDS: cantidad de documentos semejantes.

3.4.1.1 Resultados de precisión

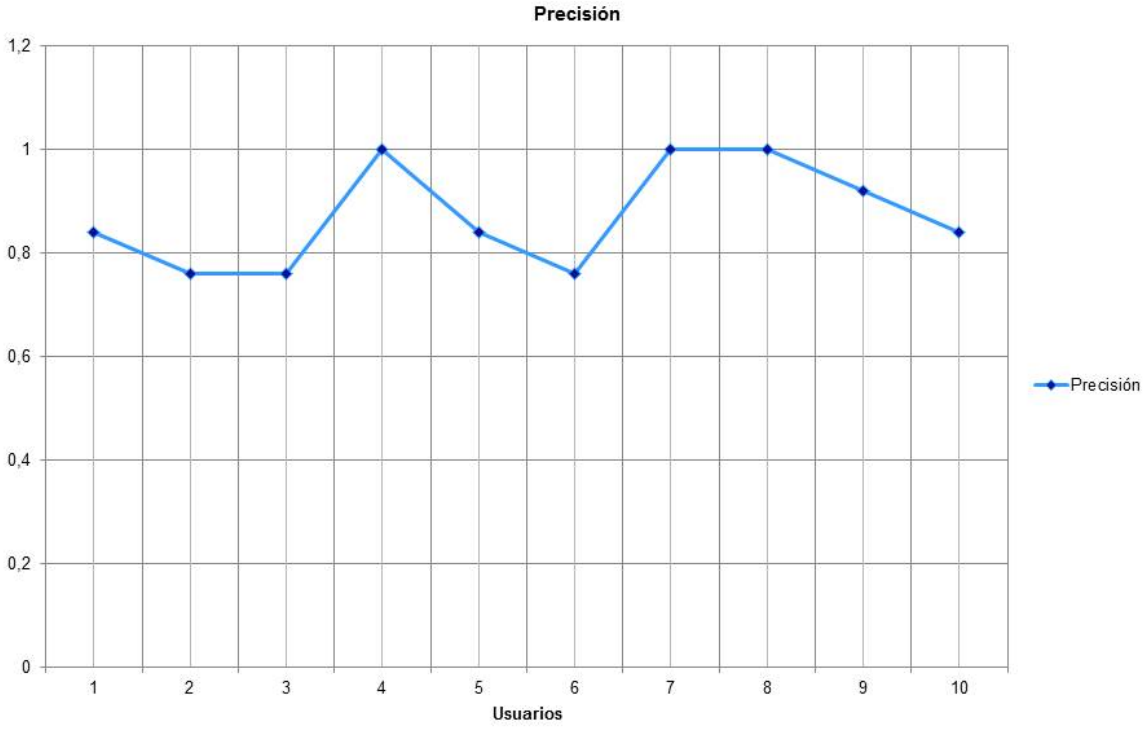
En este experimento se utilizaron 100 documentos de la colección, utilizando un umbral de semejanza de 0,7.

Se obtuvieron 92 grupos, de ellos 4 con más de 2 documentos clasificados, sobre los cuales se aplica la encuesta. Esta cantidad de grupos se debe a que mientras mayor sea el umbral definido mayor cantidad de documentos quedarán aislados, a estos documentos el algoritmo los define también como un grupo.

Los resultados de la encuesta y la aplicación de la función para el cálculo de la precisión se pueden apreciar en la Tabla 1:

Usuarios	1	2	3	4	5	6	7	8	9	10
Documentos semejantes	11	10	10	13	11	10	13	13	12	11
Valores de precisión	0.84	0.76	0.76	1	0.84	0.76	1	1	0.92	0.84

Tabla 1. Resultados encuesta y valores de precisión.



Gráfica 1. Precisión general de la solución

Como se puede observar en la Tabla 1 y Gráfica 1 los valores de precisión de la agrupación se mantienen altos, promediando una precisión de 0.87. Esto demuestra que los grupos formados cuentan con una semejanza adecuada entre los documentos que recoge.

El análisis realizado también demostró que los grupos con menos documentos van a tener una mayor precisión que los de mayor cantidad.

En la Tabla 2 se puede observar el comportamiento del agrupamiento, analizando la precisión y exhaustividad para los grupos obtenidos por el agrupamiento con al menos dos documentos.

La cantidad de documentos semejantes agrupados es la siguiente:

Grupo 1: 4 documentos.

Grupo 2: 3 documentos.

Grupo 3: 2 documentos.

Grupo 4: 2 documentos.

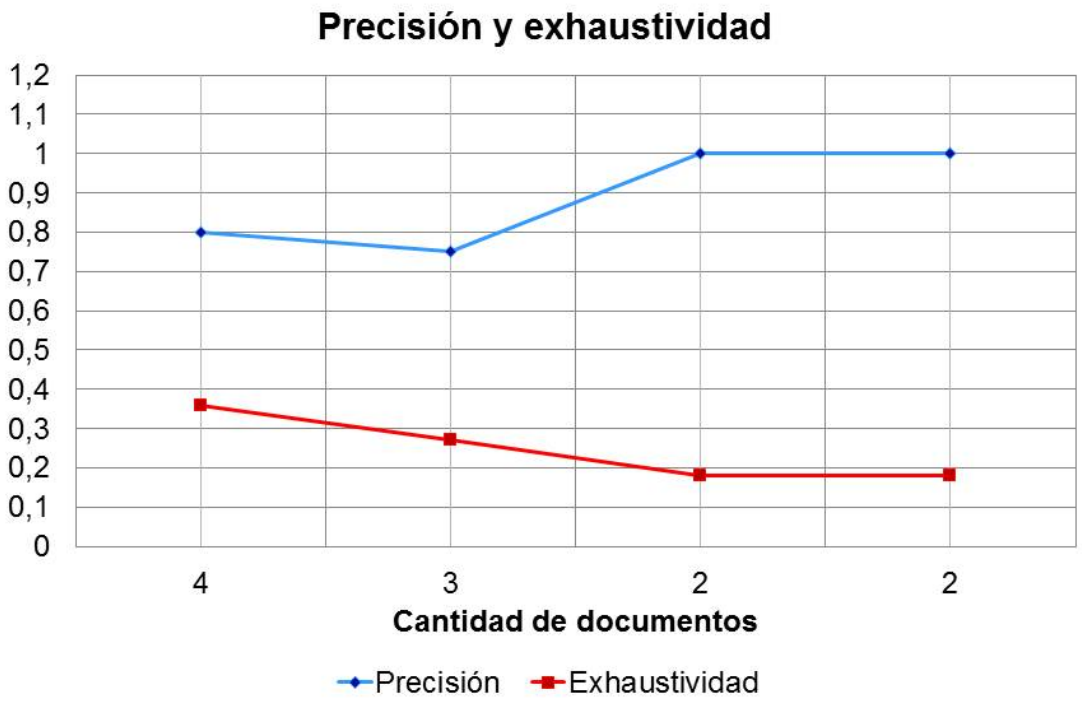
Para el cálculo de la exhaustividad se tomó como la cantidad de documentos semejantes (CDS) la suma de los documentos semejantes por cada grupo.

CDS = 11.

Grupos	CDSA	CDA	Precisión	Exhaustividad
1	4	5	0.8	0.36
2	3	4	0.75	0.27
3	2	2	1	0.18
4	2	2	1	0.18

Tabla 2. Valores de precisión y exhaustividad.

En la siguiente gráfica se observa el comportamiento de las medidas evaluadas:



Gráfica 2. Comportamiento de precisión y exhaustividad.

Como se observa anteriormente, los grupos obtenidos poseen altos valores de precisión. También se demuestra que mientras menor sean los valores de exhaustividad para un grupo, la precisión tiende a aumentar, es decir, que mientras menos documentos contengan los grupos formados existen más posibilidades de que estos documentos sean semejantes entre sí.

3.4.2 Análisis del rendimiento del módulo implementado

Para analizar el rendimiento del módulo se tomó en cuenta el tiempo de ejecución de este, para lo cual se tuvieron en cuenta los siguientes elementos:

- Cantidad de documentos a agrupar.
- Tiempo de respuesta de la solución.

Se introdujeron diferentes cantidades de documentos para luego hacer un análisis de los tiempos de ejecución arrojados por la solución, y con esto llegar a conclusiones sobre los factores que influyen en los mismos.

3.4.2.1 Resultados experimento para el tiempo de ejecución.

En la siguiente tabla se puede observar el comportamiento de los tiempos de respuestas del algoritmo *Extended Star*, donde fueron utilizadas cantidades de documentos desde 100 hasta 400:

Cant. de documentos	100	200	300	400
Tiempo de respuesta	17 s	1min: 7s	2 min: 38 s	4 min: 40 s

Tabla 3. Tiempo de respuesta del algoritmo implementado.

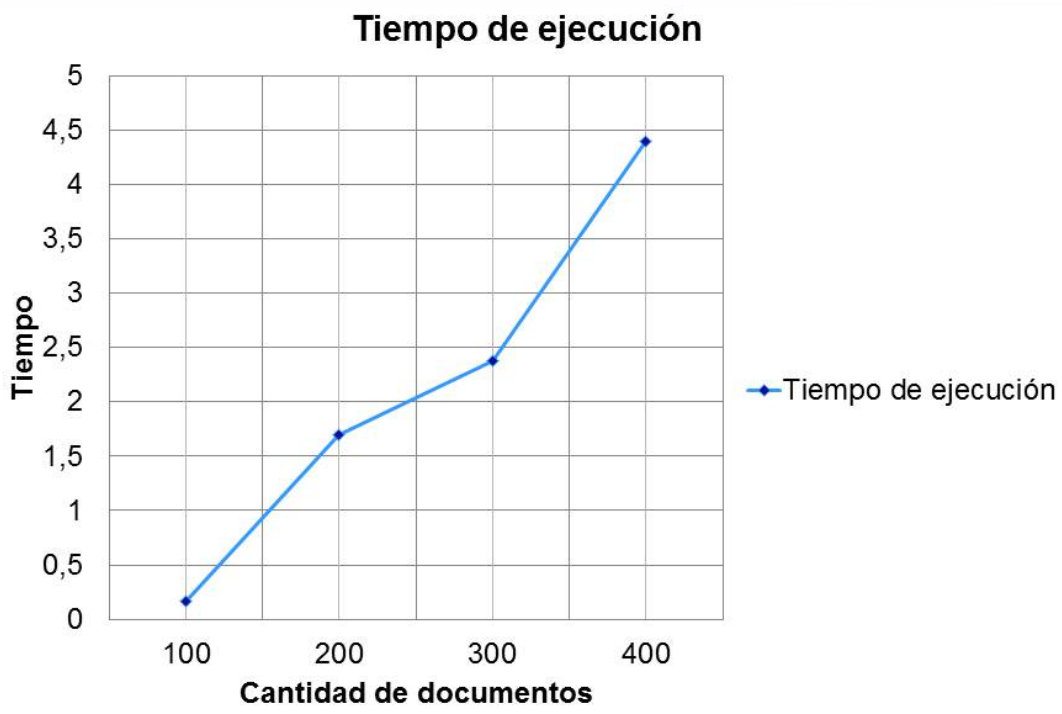


Gráfico 3. Tiempo de respuesta.

Los resultados obtenidos en el experimento realizado anteriormente muestran que, mientras mayor sea la cantidad de documentos a agrupar, el tiempo de respuesta del módulo aumenta. Esto se debe al grado de complejidad de la función de semejanza utilizada por el algoritmo, la cual debe calcular la semejanza existente entre cada par de documentos de la colección introducida.

3.5 Conclusiones parciales

La descripción de cada uno de los elementos y funciones necesarias para la construcción del algoritmo *Extended Star* facilitó que el mismo se implementara de forma correcta.

El diagrama de componentes refleja los archivos por los cuales está compuesto el módulo implementado, lo cual ayuda a comprender la estructura del mismo.

Las pruebas realizadas sobre la solución demostraron que los documentos de los grupos obtenidos son semejantes entre ellos.

El análisis del rendimiento del módulo muestra que la solución a medida que la cantidad de documentos a agrupar es mayor el tiempo de ejecución aumenta considerablemente.

CONCLUSIONES GENERALES

- Resultó el algoritmo *Extended Star* el candidato para implementar la solución propuesta.
- Las herramientas y lenguajes seleccionados permitieron una correcta implementación del módulo.
- Mediante el algoritmo *Extended Star* se logró la agrupación de los documentos de acuerdo a su similitud.
- El algoritmo se ejecuta correctamente con altos valores de precisión en los grupos formados.

RECOMENDACIONES

Con el objetivo de mejorar la calidad del módulo construido se proponen las siguientes recomendaciones:

- Incluir otros algoritmos de agrupamiento *Star* a la solución implementada.
- Implementar la función de semejanza separada del algoritmo de agrupamiento, guardándose en la base de datos el valor de similitud existente para cada documento.

BIBLIOGRAFÍA

1. TOLOSA, G. H. y BORDIGNON, F. R. A. *Introducción a la Recuperación de Información. .* Buenos Aires, Argentina: 2007

2. RODRÍGUEZ, F. G. *Experto en Drupal 7 Nivel Avanzado.* Editado por: S.L., F. 2012, ISBN 978-84-939410-5-5.

3. *Una introducción a APACHE.* 2010, Disponible en: http://linux.ciberaula.com/articulo/linux_apache_intro/.

4. AMÓN, I. y JIMÉNEZ, C. *Funciones de Similitud sobre Cadenas de Texto: Una Comparación Basada en la Naturaleza de los Datos.* 2010

5. ARCO, L.; BELLO, R.*, et al. Agrupamiento de Documentos Textuales mediante Métodos Concatenados.* Villa Clara: 2006,

6. ARCO, L.; BELLO, R.*, et al.* Agrupamiento de Documentos Textuales mediante Métodos Concatenados. 2009.

7. ASLAM, J. A.; PELEKHOV, E.*, et al. The Star Clustering Algorithm for Information Organization.* 2006.

8. BAEZA-YATES, R. y RIBEIRO-NETO, B. *Modern Information Retrieval.* 1999, ISBN 0-201-39829-X.

9. BORDIGNON, F. R. A. *RECUPERACIÓN DE INFORMACIÓN: UN ÁREA DE INVESTIGACIÓN EN CRECIMIENTO.* 1 ed. Luján, Argentina: 2007, vol. 6, ISBN 1856-4194.

10. FIGUEROLA, C. G.; BERROCAL, J. L. A.*, et al. Algunas Técnicas de Clasificación Automática de Documentos.* Universidad de Salamanca, 2007.

11. GARCÍA, L. A.; PÉREZ, R. B.*, et al. CorpusMiner 1.0: Herramienta para el agrupamiento de documentos.* 2007, vol. 1.

12. ROCHA, R. y COBO, Á. *Automatización de procesos de categorización jerárquica documental en las organizaciones.* 2010, Disponible en: http://revistas.concytec.gob.pe/scielo.php?pid=S2070-836X2010000100013&script=sci_arttext. ISBN 2070-836X

13. GARCÍA, R. J. G. *ALGORITMOS DE AGRUPAMIENTO SOBRE GRAFOS Y SU PARALELIZACIÓN.* Tesis Doctoral, Departamento de Ingeniería y Ciencia de los Computadores. Escuela Superior de Tecnología y Ciencias Experimentales, 2005.

14. GARCÍA, R. J. G. *ALGORITMOS DE AGRUPAMIENTO SOBRE GRAFOS Y SU PARALELIZACIÓN.* Tesis Doctoral, Departamento de Ingeniería y Ciencia de los Computadores. 2005.

15. GIL-GARCÍA, R. J.; BADÍA-CONTELLES, J. M.*, et al. Extended Star Clustering Algorithm.* Santiago de Cuba, Cuba: 2003,

16. GONZÁLEZ, L. C. y TORRES, E. R. P. *Extensión de Visual Paradigm for UML para el Desarrollo Dirigido por Modelos de aplicaciones de gestión de información.* Tesis de Diploma, Universidad de las Ciencias Informáticas, 2012.

17. GUERRA-GANDÓN, A.; VEGA-PONS, S.*, et al. Reconocimiento de Patrones.* 2012, ISBN 2072-6287.

18. MORENO, C. D. *CLUSTERING Y MAPAS AUTOORGANIZATIVOS (KOHONEN).* 2007.

19. PARÉ, R. C.; SANTILLÁN, L. A. C.*, et al. Bases de datos* 2010, Disponible en: http://www.dataprix.com/bases-datos-1.

20. PÉREZ, M. T. *Sistema para el Control de Asistencia integrado al CMS Drupal.* Tesis de diploma, Universidad de las Ciencias Informáticas, 2010.

21. POPESCU, M. y HRISTEA, F. State of the art versus classical clustering for unsupervised word sense disambiguation. *Springer Science & Business Media B.V.*, 2010, nº ISSN 10462-010-9193-7.

22. RAWTANI, M. R. y CHIDAMBARAM, S. S. *Drupal: The Open Source Content Management System Software Suit For Library With Library 2.0 Features.* Pondicherry University, 2009.

23. SANTILLÁN, L. A. C.; GINESTÀ, M. G.*, et al. Bases de datos en MySQL.* 2010, Disponible en: http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02151.pdf.

24. SEDEÑO, R. O. L. *Herramientas para un observatorio de información.* Granma: 2010, vol. 3, Disponible en: http://publicaciones.uci.cu/index.php/SC.

25. SEDEÑO, R. O. L. *Clasificación y agrupamiento de textos de noticias.* Granma: 2011, vol. 4, Disponible en: http://publicaciones.uci.cu/index.php/SC/article/view/373.

26. SEQUERA, J. L. C. *Nueva propuesta evolutiva para el agrupamiento de documentos en sistemas de recuperación de información.* Tesis Doctoral, Universidad de Alcalá, 2010.

27. SUÁREZ, A. P.; DELGADO, G. G.*, et al. Algoritmos de agrupamiento para colecciones de documentos.* Ciudad de La Habana: 2008, Disponible en: http://www.cenatav.co.cu/doc/RTecnicos/. ISBN 2072-6260.

28. SUÁREZ, A. P.; TRINIDAD, J. F. M.*, et al. Algoritmos dinámicos para el agrupamiento con traslape.* 2010

29. VALERO, A. T. *Extracción de Información con Algoritmos de Clasificación.* Tesis de maestría, Instituto Nacional de Astrofísica, Óptica y Electrónica., 2005.

30. YOLIS, E. *Algoritmos Genéticos aplicados a la categorización automática de documentos.* Tesis de Grado, Laboratorio de Sistemas Inteligentes. Universidad de Buenos Aires, 2003.

31. DÍEZ, J. R. F. D. C. *Sistemas de agrupamiento automático (clustering) en documentos mediante técnicas de softcomputing: Aplicaciones de algoritmos genéticos* 2010, Disponible en: http://www.uned.es/ca-guadalajara/actividades/09-10/Verano10/IA2010/SistemasAgrupamientoAutomatico.pdf.

REFERENCIAS BIBLIOGRÁFICAS

1. **BORDIGNON, F. R. A.** ***RECUPERACIÓN DE INFORMACIÓN: UN ÁREA DE INVESTIGACIÓN EN CRECIMIENTO.*** **1 ed. Luján, Argentina: 2007, vol. 6, ISBN 1856-4194.**

2. **BAEZA-YATES, R. y RIBEIRO-NETO, B.** ***Modern Information Retrieval.*** **1999, ISBN 0-201-39829-X.**

3. **YOLIS, E.** ***Algoritmos Genéticos aplicados a la categorización automática de documentos.*** **Tesis de Grado, Laboratorio de Sistemas Inteligentes. Universidad de Buenos Aires, 2003.**

4. **CASTELLANO, A. R.** ***Exactitud y precisión.*** **2009, Disponible en: http://www.upaep.cesat.com.mx/index.php?option=com_content&view=article&id=28:exactitud-y-precision&catid=11:metrologia&Itemid=14.**

5. **SEQUERA, J. L. C.** ***Nueva propuesta evolutiva para el agrupamiento de documentos en sistemas de recuperación de información.*** **Tesis Doctoral, Universidad de Alcalá, 2010.**

6. **GARCÍA, R. J. G.** ***ALGORITMOS DE AGRUPAMIENTO SOBRE GRAFOS Y SU PARALELIZACIÓN.*** **Tesis Doctoral, Departamento de Ingeniería y Ciencia de los Computadores. Escuela Superior de Tecnología y Ciencias Experimentales, 2005.**

7. **SUÁREZ, A. P.; DELGADO, G. G.*, et al.*** ***Algoritmos de agrupamiento para colecciones de documentos.*** **Ciudad de La Habana: 2008, Disponible en: http://www.cenatav.co.cu/doc/RTecnicos/. ISBN 2072-6260.**

8. **SEDEÑO, R. O. L.** ***Clasificación y agrupamiento de textos de noticias.*** **Granma: 2011, vol. 4, Disponible en: http://publicaciones.uci.cu/index.php/SC/article/view/373.**

9. **GIL-GARCÍA, R. J.; BADÍA-CONTELLES, J. M.*, et al.*** ***Extended Star Clustering Algorithm.*** **Santiago de Cuba, Cuba: 2003,**

10. **ASLAM, J. A.; PELEKHOV, E.*, et al.*** ***The Star Clustering Algorithm for Information Organization.*** **2006,**

11. **SEDEÑO, R. O. L.** ***Herramientas para un observatorio de información.*** **Granma: 2010, vol. 3, Disponible en: http://publicaciones.uci.cu/index.php/SC.**

12. **GARCÍA, L. A.; PÉREZ, R. B.*, et al.*** ***CorpusMiner 1.0: Herramienta para el agrupamiento de documentos.*** **2007, vol. 1,**

13. **COMMUNITY, N.** ***NetBeans IDE - The Smarter and Faster Way to Code*** **[Consultado el: enero de 2013]. Disponible en: http://netbeans.org.**

14. **ECURED.** ***Sistema Gestor de Base de Datos.*** **2011, Disponible en: http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos.**

15. **SANTILLÁN, L. A. C.; GINESTÀ, M. G.,** ***et al. Bases de datos en MySQL.*** **2010, Disponible en: http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02151.pdf.**

16. ***Una introducción a APACHE.*** **2010, Disponible en: http://linux.ciberaula.com/articulo/linux_apache_intro/.**

17. **GONZÁLEZ, L. C. y TORRES, E. R. P.** ***Extensión de Visual Paradigm for UML para el Desarrollo Dirigido por Modelos de aplicaciones de gestión de información.*** **Tesis de Diploma, Universidad de las Ciencias Informáticas, 2012.**

18. **GONZÁLEZ, E.** ***¿Qué es PHP? y ¿Para qué sirve? Un potente lenguaje de programación para crear páginas web.*** **Disponible en: http://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=492:ique-es-php-y-ipara-que-sirve-un-potente-lenguaje-de-programacion-para-crear-paginas-web-cu00803b&catid=70:tutorial-basico-programador-web-php-desde-cero&Itemid=193.**

19. **GARCIA, J.** ***Iniciación a PHP*** **[Consultado el: 11 enero de 2013]. Disponible en: http://www.webestilo.com/php/php00.phtml.**

20. **ROCHA, R. y COBO, Á.** ***Automatización de procesos de categorización jerárquica documental en las organizaciones.*** **2010, Disponible en: http://revistas.concytec.gob.pe/scielo.php?pid=S2070-836X2010000100013&script=sci_arttext. ISBN 2070-836X**

21. **SERINTO.** ***Arquitectura de Drupal 7*** **Disponible en: http://serinto.com/documentacion/programacion/drupal/introduccion/arquitectura-de-drupal-7.**

22. **RODRÍGUEZ, F. G.** ***Experto en Drupal 7 Nivel Avanzado.*** **Editado por: S.L., F. 2012, ISBN 978-84-939410-5-5.**

23. **ARCO, L.; BELLO, R.,** ***et al. Agrupamiento de Documentos Textuales mediante Métodos Concatenados.*** **Villa Clara: 2006,**

24. **AMÓN, I. y JIMÉNEZ, C.** ***Funciones de Similitud sobre Cadenas de Texto: Una Comparación Basada en la Naturaleza de los Datos.*** **2010**

ANEXOS

1. Grupos de documentos para la encuesta realizada:

Grupo 1

Documento 935:

Título: Pruebas de aceptación para un software con la presencia de una entidad certificadora de la calidad.

Descripción: En este artículo se define un flujo de trabajo de pruebas de aceptación del cliente con la participación de un tercero confiable, una empresa certificadora de calidad. En él se detallan: quiénes participan, qué hacen, cuándo y cómo deben hacerlo; así como, qué artefactos se generan. Es de destacar la participación a este nivel de los clientes, los usuarios finales, los desarrolladores y la empresa tercera garantizando la calidad y el buen desarrollo del proceso.

Documento 916:

Título: Análisis de modelos de calidad internacionales con respecto a su aplicación a la industria cubana del software.

Descripción: Se aborda la ausencia de aplicación de modelos de calidad para la producción de software en Cuba, haciendo un análisis crítico de la posible aplicación de modelos internacionales a las condiciones de nuestro país. Se proponen los objetivos a lograr en un sistema de calidad cubano.

Documento 967:

Título: Gestión de indicadores en proyectos de software. Perspectivas actuales y futuras.

Descripción: Se abordan las temáticas sobre la obtención de medidas e indicadores, así como modelos y herramientas surgidas a lo largo de las últimas décadas en torno a la medición de los procesos y construcción de productos de software, perspectivas actuales y futuras. El contenido servirá de apoyo para la conformación y despliegue del proceso de Medición y Análisis en el marco del Programa de Mejoras que lleva a cabo la Universidad de las Ciencias Informáticas (UCI).

Documento 977:

Título: Una experiencia novedosa para el testing desarrollada por un departamento de pruebas de software.

Descripción: La Producción de Software y Servicios Informáticos se basa en la integración de los procesos de formación, investigación y producción en torno a una temática para convertirla en una rama productiva. Las investigaciones en la Universidad de las Ciencias Informáticas (UCI)

potencian los resultados en la producción y la formación, con la participación importante del movimiento estudiantil. ¿Se convierte la UCI en un Centro mitad Universidad mitad Empresa? Definitivamente, la UCI es una Universidad cuya misión es producir software y brindar servicios informáticos a partir de la vinculación estudio – trabajo como modelo de formación. Con la creación del Laboratorio Industrial de Pruebas de Software (LIPS) se asegura que cada artefacto haga lo adecuado en todo momento. Va más allá de asegurar la idoneidad de un servicio o producto, ya que hace posible una gestión integral del valor añadido mediante el cumplimiento y la superación de las expectativas de los clientes. Se vincula la actividad productiva con la docente, impartiendo a los estudiantes de 2do año clases guiadas a obtener conocimientos relacionados con la gestión de la Calidad de Software desde la práctica, posibilitándole obtener una mejor y mayor preparación en los próximos años de la carrera.

Documento 978:

Título: Primeras ideas de un modelo cubano de referencia para el desarrollo de aplicaciones informáticas.

Descripción: Se examinan los modelos de referencia internacionales y nacionales para el desarrollo de aplicaciones informáticas y se fundamenta como ellos se encuentran frecuentemente ante la necesidad de demostrar calidad en su funcionamiento, en sus programas de formación y en la eficiente utilización de los recursos tecnológicos para lograr calidad en los productos. Los autores consideran es necesario el diseño e implantación en el país de un modelo propio que se adapte a nuestra cultura, a nuestro modelo económico y que incluya más elementos de gestión de conocimientos para la mejora continua del modelo y de las organizaciones. El estudio refleja los criterios iniciales para la definición del modelo.

Grupo 2

Documento 937:

Título: Propuesta metodológica del proceso de concepción y ejecución de contratación de software educativo

Descripción: La creciente necesidad de integración entre los procesos de enseñanza-aprendizaje y las Tecnologías de la Información y las Comunicaciones ha generado una alta demanda de productos de software educativo a nivel mundial. Numerosas son las afectaciones en la ejecución de los proyectos y múltiples los factores causales de esta situación. En este contexto se debe prestar especial atención a la gestión de compromisos en el desarrollo del proyecto. El presente trabajo tiene por objetivo establecer elementos a tener en cuenta en el

proceso de gestión de contratación en el proyecto que permita a los gerentes de proyectos la realización de un sano proceso de negociación y gestión de compromisos.

Documento 942:

Título: Gestión de recursos humanos por competencias en los proyectos de software

Descripción: Este trabajo tiene como objetivo contribuir a la implementación, en la industria de software, de la gestión de recursos humanos basado en el enfoque por competencias. Para ello, se realiza y fundamenta una propuesta de roles invariantes que participan tanto en proyectos de desarrollo como de implantación de software, y de competencias tanto técnicas como genéricas requeridas para desempeñar los roles propuestos. En el trabajo se fundamenta el uso del método Delphi y las variaciones requeridas, como vía para validar la propuesta de roles y competencias, y para identificar la importancia de cada competencia en el desempeño de cada rol. Además, se propone un procedimiento para determinar el índice de competencia de una persona para desempeñar un rol asignado en el proyecto.

Documento 953:

Título: Retos en la gestión de los riesgos de proyectos de software

Descripción: Se describen elementos acerca de los riesgos en proyectos de desarrollo de software como su definición y clasificación. Se analizan algunos estudios evolutivos y se establece una comparación de los modelos de Gestión de Riesgos. La profundidad con que los modelos aborden aspectos como la planificación, la comunicación y la utilización de las métricas no solo para caracterizar el riesgo sino para valorar y mejorar los resultados de la Gestión de Riesgos, junto con la facilidad de uso de las técnicas y herramientas, determinará la aplicación formal, sistemática, estructurada y coherente de estas prácticas en los proyectos de desarrollo de software.

Documento 967:

Título: Gestión de indicadores en proyectos de software. Perspectivas actuales y futuras.

Descripción: Se abordan las temáticas sobre la obtención de medidas e indicadores, así como modelos y herramientas surgidas a lo largo de las últimas décadas en torno a la medición de los procesos y construcción de productos de software, perspectivas actuales y futuras. El contenido servirá de apoyo para la conformación y despliegue del proceso de Medición y Análisis en el marco del Programa de Mejoras que lleva a cabo la Universidad de las Ciencias Informáticas (UCI).

Grupo 3

Documento 931:

Título: El sistema de calidad del Instituto Central de Investigación Digital, la documentación de software y RUP.

Descripción: Informática actualmente. Una metodología conocida internacionalmente para facilitar este proceso es Rational Unified Process (RUP) que cuenta con herramientas integradas. En el artículo se describen aspectos esenciales del RUP y del Sistema de Calidad del Instituto Central de Investigación Digital (ICID) basado en la norma ISO 9000:2000 (SGC ICID), proponiendo la forma de utilizar la información obtenida de RUP en la documentación exigida por el procedimiento de diseño del SGC ICID.

Documento 971:

Título: Proceso de capacitación que propone RUP según normas de calidad.

Descripción: En el presente trabajo se realizó una mejora del proceso de Capacitación de RUP utilizando elementos de las definiciones de la gestión de los recursos humanos de una organización que definen normas de calidad como CMMI, PCMM e ISO. Mediante el método analítico-sintético, se identificaron los elementos de las normas de calidad en los que se ganan criterios de qué se debería hacer para la optimización de la Capacitación de RUP. Partiendo de esta información, se rediseña el proceso de Capacitación de manera que queden definidas un conjunto de actividades que a partir de elementos de entrada, se generan elementos de salida, utilizando técnicas y herramientas factibles. Estas actividades son realizadas por el Jefe de Entrenamiento y Capacitación y el Desarrollador de Cursos.

Grupo 4

Documento 992:

Título: Propuesta de arquitectura de una herramienta web para la administración del gestor PostgreSQL.

Descripción: El presente artículo está orientado a proporcionar una breve descripción de la arquitectura de una herramienta Web para la administración del gestor de bases de datos PostgreSQL. En el mismo se exponen las principales funcionalidades que debe tener la herramienta para brindar a los usuarios una eficiente administración y uso del gestor, que a su vez lo haga atractivo a la vista de los usuarios y estos se sientan más identificados con él. También se realiza una descripción de las tecnologías libres a usar en su desarrollo y la integración de los módulos que conforman la herramienta.

Documento 993:

Título: Seguridad en bases de datos

Descripción: Los sistemas de base de datos son de gran uso, el cual va desde el uso de bases de datos ligeras, bases de datos en tiempo real (en algunas ocasiones obtenida a partir de la optimización de bases de datos relacionales) y bases de datos relacionales con potentes gestores como aplicación proveedora del servicio. En el mundo del software libre existen gran cantidad de gestores de bases de datos, entre los que se encuentran mysql, berkeley db, sqlite y postgresql. Este último con gran número de seguidores y de personas de gran nivel en el mundo del desarrollo libre que contribuyan a su propósito. El presente trabajo tiene como objetivo principal mostrar mediante una investigación los diferentes mecanismos para realizar la configuración de la seguridad para el gestor de base de datos Postgres.