Facultad 10

Detección de términos relevantes en páginas Web analizando las Hojas de Estilo en Cascada para la clasificación automática de documentos en el Motor de Clasificación Inteligente por Contenido.

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

Autores:

- ✓ Abimael Torres Valdés
- ✓ Raicel Espinosa Menéndez

Tutores:

- ✓ Karel Antonio Verdecia Ortiz
- ✓ Aleida Eva Sáez Aldana

Ciudad de La Habana, Junio de 2010

"Año del 51 Aniversario del Triunfo de la Revolución Cubana"

DECLARACIÓN DE AUTORÍA

Por este medio declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) para que haga el uso que estime pertinente con el mismo.

Para que así conste firmamos la presente a los ___ días del mes de _____ de 2010.

Autores:

_____ _____

Abimael Torres Valdés **Raicel Espinosa Menéndez**

Tutores:

_____ _____

Karel Antonio Verdecia Ortiz **Aleida Eva Sáez Aldana**

Oponente:

_____

Isachi Abreu Gil

AGRADECIMIENTOS

Agradezco primeramente a Dios por permitirme llegar hasta el final de esta carrera, una de las tantas que me quedan en la vida, y poder decir "hasta aquí me ha ayudado en todo y lo seguirá haciendo".

A mis padres, Ibrahim y Yolanda que son un regalo muy especial en mi vida, por su apoyo incondicional y su amor todo el tiempo, no saben cuánto los amo a ambos.

A mi hermana Sarai que siempre estábamos fajados como buenos hermanos. Siempre has sido para mí un pilar muy importante en mi vida, eres una parte esencial de mí, gracias por existir.

A mis abuelos Roberto, Martha y Linda, que son mis padres más veteranos, y sus consejos siempre son bien recibidos en todo momento.

A mis Tíos y Primos, son tantos que no me alcanzaría otra tesis para mencionarlos a todos, no saben cuánto me han apoyado en la vida, y lo importante que han sido cada minuto con sus consejos y atenciones.

A mis amigos y amigas que me han soportado todos estos años de universidad, y han estado conmigo en los momentos buenos y malos, los considero.

A todas mis amistades en general que siempre han estado a mi lado dándome apoyo y consejos.

A todas aquellas personas que han aportado sus conocimientos para la realización de este trabajo de diploma. Este triunfo es de todos ustedes, los llevo en el corazón, Dios los bendiga.

Gracias y mil gracias a todos. Los quiero.

Abimael Torres Valdés

En primer lugar le agradezco a la dirección de este país donde todos los jóvenes tienen la posibilidad de estudiar gratuitamente carreras universitarias siempre y cuando estén facultados para ello.

A la Universidad de las Ciencias Informáticas por la formación que me ha dado durante este largo periodo de estudio.

A mi familia, que siempre ha estado pendiente de todo lo que he necesitado y me ha apoyado en los antojos de toda la vida y que me sigue apoyando aunque ya me he hecho independiente, y no cito con detalles porque ellos saben perfectamente quienes son: los que siempre me han acompañado en todo momento, que parte de ellos están aquí hoy conmigo.

A los tutores por colaborar en la realización del trabajo de diploma. A todas las personas que han brindado su ayuda incondicional porque a veces quien menos uno espera, tiende su mano en el momento preciso.

A todos los profesores que he tenido.

A mis compañeros de grupo, de apartamento, en fin, sería una lista enorme y se hace imposible mencionarlos a todos, además creo que el estar agradecido se demuestra con el quehacer diario, la actitud que se toma para con los demás y no con las palabras que se puedan expresar en este corto momento donde se está nervioso y emocionado siendo obvio que se omiten muchas cosas.

A todas las personas que he conocido aquí en la universidad y fuera también, con quienes he compartido tanto malos como buenos momentos. A los que están hoy aquí presentes, gracias por asistir. A todos les agradezco por aceptarme con mis defectos y virtudes, muy en especial a mi compañero de tesis que me ha soportado durante todo este tiempo, por escogerme precisamente a mí para realizar en conjunto este trabajo tan importante para ambos que marca el fin de nuestra carrera universitaria. A todos muchísimas gracias por formar parte de la historia de esta etapa de mi vida.

Raicel Espinosa Menéndez

DEDICATORIA

Dedico este trabajo a mis padres y mi hermana, por ser un motor impulsor en mi vida, por su amor, comprensión y apoyo todo el tiempo.
A mi familia, abuelos, tíos y primos, esa maravillosa y única por la que me tildo de privilegiado al poder contar con su amor y cariño.
A todas aquellas personas que de cierto modo estuvieron muy cerca de mí en esta genial etapa y me permitieron contar con su apoyo y compañía.
Este trabajo de diploma está dedicado a ustedes, ya que sin ustedes no hubiera sido posible realizarlo.

Abimael Torres Valdés

Dedico este trabajo en especial a quien me dio la vida: mi mamá, porque gracias a ella estoy hoy aquí.
A mi abuela Gladis, mi tía Maricel, Lázaro mi padrastro, a Yuli mi prima, en fin a aquellas personas que siempre me brindaron su apoyo y me han dado aliento en todos los momentos, a todos esos que de una u otra forma han influido en mí siempre para bien.
A quienes me han brindado su cariño incondicionalmente.
A todos mis compañeros de grupo que han sido los que más cerca han estado y que han contribuido a que juntos hayamos logrado el objetivo por el cual abandonamos nuestros hogares para venir a esta universidad hace ya 5 años.
A todos los conocidos y amigos, que han formado parte de esta gran familia aquí en la Universidad de las Ciencias Informáticas.

Raicel Espinosa Menéndez

RESUMEN

En la Universidad de las Ciencias Informáticas se desarrolla un Motor de Clasificación Inteligente por Contenidos (MOCIC) con el fin de que clasifique de forma automática y por categorías, según el tema, la información que está en las páginas Web. Este motor posee entre otros módulos uno para la clasificación específicamente de texto. Para este módulo se desea implementar la función de ponderación ACC (Combinación Analítica de Criterios), que se basa en una combinación heurística de criterios analizando sólo el Lenguaje de Marcación de Hipertexto (HTML) de las páginas para detectar los términos relevantes y saber a qué categoría puede pertenecer. La tendencia actual en el mundo del diseño Web es el uso de las Hojas de Estilo en Cascada para dar estilo a las páginas y se propone en este trabajo un método que permite la obtención de términos relevantes en las páginas HTML a través del análisis de las CSS en función de uno de los criterios que combina la ACC, el “enfatizado”.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INTRODUCCIÓN

Internet es una infraestructura de redes que conecta a la vez todo tipo de ordenadores. Conocida como la "red de redes", es la autopista de la información por excelencia permitiendo la comunicación de millones de usuarios en todo el mundo ***(Noel L. Núñez Camallea & Ronald Coutin Abalo 2005)***. Es una combinación de hardware (ordenadores interconectados por vía telefónica o digital) y software (protocolos y lenguajes que hacen que todo funcione) que crece vertiginosamente a un ritmo sorprendente, pues cada día se publican en la "red de redes" miles de nuevos documentos y se conectan por primera vez miles de personas. Tras el impacto que ha tenido en el mundo el uso de esta potente herramienta, se trabaja constantemente con el fin de aumentar la rapidez de envío y recepción de datos para así mejorar la comunicación entre los usuarios desde cualquier parte del planeta.

La World Wide Web (WWW): conjunto de información multimedia ubicado en diferentes máquinas a lo largo del mundo y que están conectadas a Internet, es en la actualidad el punto más usado en este ámbito y como columna vertebral en la navegación ofrece gran cúmulo de información y diversos servicios. Se está hablando no más que del amplio mundo de la Web, páginas en formato electrónico que tratan temas específicos, poseen enlaces a otras páginas y tienen una estructura en general basada en el Lenguaje de Marcación de Hipertexto (HTML) definido por etiquetas que estructuran un texto para su visualización.

Se estima en nuestros días, que la información existente en formato digital o electrónico supere casi en número a la cantidad de libros escritos en la historia. Tanto cúmulo de contenido hace engorrosa la búsqueda de temas muy particulares y los usuarios desean consultar la información más precisa en el menor espacio de tiempo posible, existiendo una dificultad en este sentido y no precisamente en el acceso a la información, sino en la selección de entre toda la disponible, la que más satisfaga las necesidades de los mismos.

Pero ¿qué tipo de información desean encontrar los usuarios? Sencillamente de todo tipo, pues la amplitud en la red es sorprendente si de tópicos se trata, facilitando tantos de política, cultura, sociedad, preparación docente en cuanto a materias de distintas carreras universitarias, pornografía, violencia, terrorismo, aberración sexual, entre otros. No es conveniente entonces que las personas accedan a cualquier tipo de información, por lo que se precisa de una restricción al respecto.

Es fundamental que la información sea clasificada para tener el control de la categoría en que se encuentra según el tema y así pueda ser accedida en el momento deseado, la mejor,

más centrada, confiable, explícita, de toda aquella considerada pertinente para los usuarios de una institución en específico.

Cuba es un país bloqueado y el tema de Internet no está ajeno a las leyes que impone el recrudecido bloqueo, cuestión por la que no se ha permitido el enlace a un cable de fibra óptica internacional que permita tener un ancho de banda suficiente de Internet aún existiendo un número de estos cables que van hacia México y algunas islas del Caribe bordeando Cuba. Por esta razón la conexión se realiza de forma satelital, es decir, un canal que evidentemente es un servicio por el que se paga mucho más que si fuera una conexión de cable submarino y que lleva consigo demoras y retardos producto a los saltos que da la señal a través del satélite. El ancho de banda del país oscila entre los 200-300 megabits/s. Para una nación esto es muy poco y en busca de alguna alternativa a la problemática va en marcha un proyecto con la República Bolivariana de Venezuela con el propósito de enlazar a Cuba con este hermano país mediante un cable submarino y lograr una conexión mucho más robusta, fiable y barata a Internet. Este proyecto se está llevando a cabo actualmente y según se ha comentado en distintos medios de comunicación la fecha de culminación será aproximadamente para fines de este 2010 ó para el 2011.

Mientras tanto se precisa de un uso muy eficiente de Internet y por tanto ha de distribuirse entre todas las empresas e instituciones del país de la forma más homogénea posible de acuerdo a las necesidades. La UCI es privilegiada contando con un ancho de banda que aunque grande como institución si se repartiera entre la cantidad de usuarios que tiene la universidad (15 000) es en realidad muy poco. A medida que ha pasado el tiempo la institución se ha desarrollado y crecido, ampliándose también el ancho de banda. Pero no ha sido posible continuar creciendo en este sentido por la situación actual a nivel nacional y es necesario hacer una gestión eficiente de ese recurso para que no se agote, pues la universidad se conecta con las Facultades Regionales, los centros de desarrollo que están en otras provincias e instituciones que se alimentan de Internet a través del nodo central de la UCI.

En la facultad 10 de esta universidad precisamente, se desarrolla un Sistema de Filtrado de Paquetes por Contenido (FILPACON), el cual permite regular, aceptando o denegando, el acceso de usuarios a contenidos determinados de Internet y así brindar una navegación segura que se ajuste a las políticas de las instituciones en que se utilice. El mismo busca en primer lugar hacer un uso eficiente del canal de Internet educando a los usuarios en un buen uso del recurso para las necesidades básicas de la universidad: docencia, producción e investigación.

La tarea de ubicar la información por categorías, determinante dentro del proceso de

clasificación, se torna un poco compleja si se realiza de forma manual, por lo que se encuentra en desarrollo un “Motor de Clasificación Inteligente por Contenidos” (MOCIC) que se encarga de clasificar, gestionar y almacenar el contenido de las páginas Web capturadas de Internet con el objetivo de brindar un mejor acceso a la información que se determine, pueda ser accedida por los futuros usuarios. MOCIC está formado por un grupo de módulos que en su mayoría están dotados de Inteligencia Artificial (IA) permitiendo así la automatización del proceso de clasificación por contenidos para el cual jugará un papel protagónico alguna función de ponderación. Pues con la misma se determina por distintos mecanismos según sea, la relevancia de los rasgos en las páginas Web. A través del cálculo de la relevancia de los términos o rasgos de una página Web mediante factores determinantes que permiten hacerlo, se puede concebir de qué tema trata la página y por tanto la categoría a la que será asignada según su información.

En la tesis doctoral de Víctor Diego Fresno Fernández titulada Representación Autocontenida de Documentos HTML: una propuesta basada en Combinaciones Heurísticas de Criterios, de la Universidad “Rey Juan Carlos”, de España, se propone una función de ponderación denominada ACC (Combinación Analítica de Criterios) la cual se basa en una combinación lineal de criterios heurísticos extraídos de los procesos de escritura y lectura de textos. Esta función opera sobre el código HTML de la página en cuestión para realizar el proceso de ponderación.

La misma se implementa para MOCIC. Pero como el mundo actual está en constante cambio y actualización surgen nuevas tecnologías que influyen sobre las preexistentes y es el caso de las Hojas de Estilo en Cascada (CSS) que se asocian a las páginas HTML. La función ACC no las analiza.

De esta forma la **situación problémica** gira en torno a la pérdida de términos relevantes en los textos al usar la función ACC para MOCIC debido al tratamiento por parte de la misma a sólo el código HTML obviando que existan términos cuya relevancia esté dada por un estilo de tipo CSS. Surge así el **problema científico**: ¿Cómo lograr que la función ACC analice además de las páginas HTML a las CSS para que no sean obviados términos en el código HTML cuya relevancia estuviera dada por un estilo de tipo CSS? Se hace **objetivo general** de este trabajo de diploma proponer un método que permita realizar un análisis de las páginas CSS para obtener los términos que las mismas enfatizan en el código HTML. Para ello se **defiende la idea** de que es posible encontrar un método que permita analizar las páginas CSS para detectar términos en el código HTML cuya relevancia esté dada por un estilo de tipo CSS.

Se han definido los siguientes **objetivos específicos**:

- Determinar cuáles son los estilos más empleados por los diseñadores Web para resaltar determinadas frases en las páginas a través del uso de las CSS.
- Analizar qué criterios de los que combina la ACC puede utilizarse para el método que se ha de proponer como solución.
- Proponer heurísticas en función de obtener términos enfatizados en el código HTML a partir del análisis de las hojas de estilo en cascada.

Como **objeto de estudio** se toman las Funciones de Ponderación de Términos que pueden ser aplicadas a las páginas HTML para calcular la importancia o relevancia de un rasgo en el contenido de un texto. El campo de acción estará enmarcado en las Funciones de Ponderación de Términos de Carácter Local, que no son más que aquellas donde sólo se tiene en cuenta la información del propio documento para el cálculo de la relevancia, específicamente la función ACC que se implementa para MOCIC.

Como **tareas para la investigación** se definen:

- Entrevistar a personal especializado en programación Web e Inteligencia Artificial con el fin de adquirir conocimientos más profundos en el área.
- Estudiar el lenguaje HTML y las Hojas de Estilo en Cascada (CSS).
- Analizar las funciones que determinan la relevancia de los términos en una página Web, en específico la función ACC (Combinación Analítica de Criterios.)
- Investigar las diferentes técnicas de clasificación automática de textos y clustering de documentos a páginas Web.
- Analizar las diferentes técnicas de búsqueda basadas en heurística de criterios.

Métodos teóricos utilizados en la investigación:

Histórico – Lógico: para analizar la evolución de las funciones de ponderación de términos y hacer un estudio de las mismas para una mejor comprensión del tema, en especial las funciones de ponderación de términos de carácter local.

Analítico sintético: se revisa entre otros materiales la propuesta hecha en una tesis doctoral de la universidad “Rey Juan Carlos” donde se combinan heurísticamente algunos criterios para conformar una función de ponderación. Toda la atención va dirigida a los criterios que en particular se tienen en cuenta para denotar relevancia en los textos.

Inductivo – Deductivo: se analiza de manera muy detallada la función de ponderación ACC

para comprender su funcionamiento y luego determinar cuál o cuáles criterios se tienen en cuenta para la propuesta de este trabajo de diploma.

La tesis se divide en dos capítulos: El primero (Capítulo I) contiene todos los elementos teóricos suficientes para la comprensión de los temas que se tratan. El segundo (Capítulo II) propone y explica la solución a través de un ejemplo que ilustra el funcionamiento de la misma.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se exponen los principales conceptos y definiciones relacionados con el tema de investigación comenzando por una breve descripción de Internet muy ligada a la WWW como fuente principal de información en el mundo actual. El punto de partida lo constituye sin duda alguna el Lenguaje de Marcación de Hipertexto (HTML) debido a que es la base de las páginas Web y que para analizar las mismas es preciso abordar el tema de la representación de documentos. Inherentes al mundo de la Web se encuentran las Hojas de Estilo en Cascada (CSS), constituyendo un punto cumbre en este trabajo por su especial tratamiento para la solución que se propone en el Capítulo II. Las representaciones basadas en el modelo de espacio vectorial no se dejan de la mano por ser el primer paso para la posterior operación de las funciones de ponderación de términos en el proceso automatizado de clasificación de textos.

1.1 Internet y la WWW

Internet es conocida como la “red de redes”, pues conecta a miles de ordenadores en todo el mundo y brinda gran cantidad de información y servicios. Es una combinación de hardware (ordenadores interconectados por vía telefónica o digital) y software (protocolos y lenguajes que hacen que todo funcione).

Gran impacto ha causado esta potente herramienta en la sociedad, en sentido general tanto en el ocio como en el área del conocimiento por el fácil y rápido acceso que brinda a una enorme cantidad de información en forma de documentos.

Sin duda alguna lo más importante fue la aparición de la World Wide Web (WWW), también conocida como la telaraña, red o malla mundial. Básicamente consiste en un medio de comunicación en donde se dispone de documentos de hipertexto (o sea que se incluyen enlaces a otros sitios documentos) con textos, imágenes, videos, gráficos u otros objetos. La misma convierte el acceso a Internet en algo sencillo para los usuarios lo que ha propiciado a que acceder a diario a los servicios haya adquirido un crecimiento explosivo en cuanto al número de personas que la explotan con diferentes fines. Permite que la información viaje de un extremo del planeta a otro como es el caso del e-mail, así como revisar en distintos lugares en busca de ampliar información a través del conocido hipervínculo, que al hacer clic sobre él

nos comunica ya sea con otro sector del documento u otro documento en otro servidor de información. La misma es por excelencia la columna vertebral de Internet a través de la cual los usuarios acceden a los servicios y la información. Es el medio mediante el cual se viaja por la red.

1.2 Lenguaje de Marcación de Hipertexto.

1.2.1 ¿Qué es el Lenguaje de Marcación de Hipertexto?

Lenguaje de Marcación de Hipertexto (Hyper Text Markup Language) tiene como acrónimo HTML y es el lenguaje de marcas de texto utilizado por la WWW. Fue creado en 1986 por el físico nuclear Tim Berners-Lee. Es un lenguaje sencillo que permite describir hipertextos, es decir, textos presentados de una forma estructurada y agradable. El mismo se crea a partir de dos herramientas preexistentes: el concepto de hipertexto (conocido como link o ancla), el cual permite conectar dos elementos entre sí y el Lenguaje Estándar de Marcación General (SGML) que se emplea para colocar etiquetas o marcas en un texto para indicar como debe ser visto. No es un lenguaje de programación como es el caso de Visual Basic, C++, C#, Java, etc., sino un sistema de etiquetas.

HTML es un lenguaje de composición de documentos y especificación de ligas de hipertexto que define la sintaxis y coloca instrucciones especiales que no muestra el navegador, aunque sí le indica cómo desplegar el contenido del documento, incluyendo texto, imágenes y otros medios soportados. También indica cómo hacer un documento interactivo a través de esas ligas especiales de hipertexto, las cuales conectan diferentes documentos -ya sea en su computadora o en otras- así como otros recursos de Internet ***(Chuck Musciano y Bill Kennedy 1999).*** El objetivo principal es definir la estructura y la apariencia más básica de documentos de forma tal que puedan ser manejados de una manera rápida para los usuarios que estén en dispositivos diferentes.

Sencillamente HTML fue creado para darle estructura a documentos y hacerlos a su vez accesibles por lo que no da formato con perspectiva de llegar a un diseño espectacular.

1.2.2 Ventaja del Lenguaje de Marcación de Hipertexto.

El código HTML, no es más que texto y por esta razón es que solamente con un editor de texto como el que acompañan a todos los sistemas operativos: edit™ en MS-DOS, block de

notas en Windows, vi™ en UNIX, etc., basta para crear un documento de este tipo. Con uno de estos y un navegador perfectamente se puede ir apreciando como va quedando determinado trabajo. Por supuesto estos no son los únicos editores de texto que pueden ser usados, sino cualquier otro. También se puede usar procesadores de texto, que son editores con capacidades añadidas, como pueden ser Microsoft Word™ o WordPerfect™. No presenta ningún compilador, por lo tanto, algún error de sintaxis que se presente no será detectado y se visualizará en la forma como se entienda según como haya quedado estructurado.

1.2.3 Estructura del lenguaje de Marcación de Hipertexto

Un documento HTML presenta una estructura fácil y sencilla de aprender, el mismo consta de texto que define el contenido del documento y etiquetas que definen la estructura y apariencia de ese contenido.

Dicha estructura básicamente está dada por las etiqueta <HTML> y </HTML> que encierran al encabezado y cuerpo del documento. Al encabezado se le delimita con las etiquetas <head> y </head> las cuales permiten la asignación de un título e indican al navegador otros parámetros que puede utilizar al despliegue del documento. En el caso del cuerpo la delimitación la dan las etiquetas <body> y </body> donde reside toda la información. Las etiquetas o marcadores de control, como también se le conocen, sugieren al navegador como mostrar la información. La figura 1 presenta la estructura descrita anteriormente.

La generalidad de los elementos de un documento HTML constan de tres partes fundamentales en su estructura: una etiqueta de apertura, un bloque de texto y una etiqueta de cierre. Las etiquetas de apertura y cierre son denominadas contenedores, ya que contienen el ya nombrado bloque de texto, aunque también existen algunas etiquetas que sólo se componen de una de apertura no teniendo cierre como puede ser el caso de un separador por mencionar algún ejemplo.

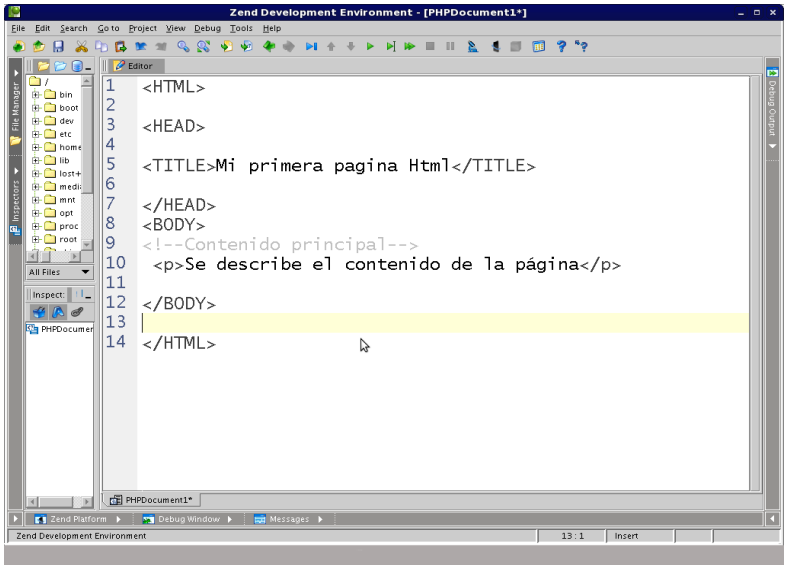

FIGURA 1 ESTRUCTURA DEL LENGUAJE HTML

1.2.4 Etiquetas del lenguaje de Marcación de Hipertextos

Cada etiqueta de HTML consta de un nombre, algunas veces seguido por diferentes opciones de atributos, todos colocados entre los símbolos mayor y menor que (< y >). La etiqueta más simple no es sino un nombre (abreviatura o comentario) encerrado entre tales signos, como <head> e <i>. Las etiquetas más complejas tienen uno o más atributos, los cuales especifican o modifican el comportamiento de la etiqueta ***(Chuck Musciano y Bill Kennedy 1999).***

Debido a que las etiquetas de documentos HTML son básicamente abreviaturas y anotaciones en el idioma inglés, que están definidas por palabras comunes, son fáciles de entender y utilizar. Tal es el caso de la etiqueta <TITLE> que es sin dudas uno de los pilares fundamentales a la hora de determinar el tema a tratar en un documento dado o en la búsqueda de un tema en específico.

Por ejemplo, cuando se va a una biblioteca en busca de algunas literaturas se toma como punto de partida de la búsqueda el título de las mismas, y después el índice sin necesidad de leer el contenido. En el caso de Google, ocurre algo parecido a lo antes expuesto, ya que uno de los primeros sitios donde busca es en el título del documento.

Esta etiqueta es verdaderamente importante debido a que:

1. Describe e identifica las páginas.

2. Le dan al lector una idea de la información que contienen los documentos.

3. La mayoría de los navegadores encabezan los resultados con los títulos de cada documento.

4. Cuando alguien añade una Web a sus favoritos el título aparece como la descripción del sitio en la carpeta de favoritos.

Por otra parte el hecho de que algunas frases o términos aparezcan en un documento de forma tal que resalten a la vista del usuario da una medida de la importancia que ha querido otorgarle el autor entre todos los términos del documento, logrando así la atención del usuario a esta parte del texto para dar a entender que en estas frases o términos se encierra gran parte del tema central que se aborda. Esta técnica presenta gran ventaja, ya que los lectores pondrán su atención en las frases o términos que les sean más llamativas en el texto, poniendo para sí más comprensión en las mismas. De esta forma se comprueba que el estilo dado a un documento juega un papel fundamental a la hora de otorgarle un significado a una parte específica del texto ***(Andrés Fernández 2010).***

Un estudio realizado en 1997 por John Morkes y Jacob Nielsen acerca de la forma en que las personas leen plantea que la lectura de textos en pantallas de ordenadores es diferente a la de un texto en papel. Para ello se tomó una muestra significativa como usuarios de prueba y sólo un 16% de ellos leyó las páginas Web mostradas de modo secuencial, frente a un 79% que al leer un documento HTML, lo hicieron saltando entre los temas más importantes, fijando su atención en diferentes partes de la páginas, y no palabra por palabra como ocurre en los textos impresos. Es cuando a partir de esta prueba surgen recomendaciones claves a la hora de escribir un documento HTML que sin duda alguna toma un lugar importante en el mundo actual dado el uso de la "red de redes" y que tendrá características totalmente distintas a las que tienen los textos en papel, por lo que se le debe prestar una gran atención a este aspecto ***(Morkes, J. y Nielsen 1997)***.

Avances en este sentido apuntan hacia el tratamiento del código HTML, porque como la función del etiquetado es resaltar algunas partes de otras lógicamente ahí está el detalle de establecer relevancias de partes del texto, de esta forma elementos de enfatizado como la cursiva, la negrita, el subrayado, etc, pueden utilizarse como indicadores de relevancia. También así la frecuencia de aparición de términos y la posición en la que se encuentra en el texto pueden ser verdaderamente significativas.

Existen varias etiquetas que tienen como función principal resaltar algunas frases del texto frente a otras. Es el caso de …, <u>...</u>, ..., ... o

..., que lo hacen de forma enfatizada y así llaman la atención del usuario con un efecto singular en cada caso. En la mayoría de las ocasiones bastaría solo con tomar estas partes enfatizadas del texto para tomar una idea de lo que trata el documento en general.

Existe otra forma de resaltar un determinado fragmento de texto. Es el caso de las etiquetas <h1>…</h1> que van con esta misma estructura hasta <h6>…</h6> y que se utilizan con los títulos en los encabezados. El uso de las mismas hace mucho más fácil la comprensión y análisis del texto, permitiendo percibir la información relevante del mismo en dependencia del tamaño que tomen los caracteres y buscar así la relación entre las diferentes partes del texto.

El estilo de un documento como se aprecia en lo antes explicado es un eslabón fundamental para la comprensión por parte de quien lo lea. El HTML se limita a la hora de dar estilos determinados o sencillamente dificulta en ocasiones el tener que darlos en varias páginas de un sitio Web. Como alternativa en la actualidad se desarrollan los sitios Web con la tecnología CSS para un trabajo más fácil y eficiente si es que a estilos hay que referirse.

1.3 Hojas de Estilos en Cascada

1.3.1 ¿Qué es CSS?

CSS son las siglas de Cascading Style Sheets, en español, Hojas de estilo en Cascada. Es una tecnología que permite crear páginas Web de una manera más exacta representando un avance importante para los diseñadores, quienes tienen un mayor rango de posibilidades para mejorar la apariencia de las páginas. Es la mejor forma de separar los contenidos y su presentación, imprescindible para que puedan ser creadas páginas Web complejas. El hecho de separar el contenido del diseño presenta entre otras ventajas la de brindar una mejor accesibilidad al documento así como que el mismo pueda ser visualizado en gran cantidad de dispositivos diferentes.

Por otra parte uno de los antecedentes más notables que ha conllevado al desarrollo de esta tecnología consiste en que las páginas Web tienen embebidas en su código HTML el contenido del documento con las etiquetas necesarias para darle forma, trayendo como inconveniente que la lectura del código se haga más pesada y difícil a la hora de buscar errores y depurar la página. El uso de las CSS reduce la complejidad del mantenimiento de la página, pues una vez creados los contenidos se utilizan para definir el aspecto de cada elemento. Más adelante se explica mejor cómo funciona esto.

En función de superar las limitaciones que tiene el lenguaje HTML cuando de dar forma a los documentos se trata, surgen las CSS, tecnología más actual y enfocada a objetivos más amplios resolviendo muchas problemáticas que con el HTML resultaran un tanto complejas o bien como se dijo ya limitadas.

Gracias a las CSS se puede ser mucho más dueño de los resultados finales de la página, logrando mejorar los aspectos que se dificultaban de alguna manera utilizando solamente HTML, como incluir márgenes, tipos de letra, fondos, colores, etc.

1.3.2 Estructura de CSS

Las hojas de estilo en cascada tienen una estructura muy simple, pero muy flexible y potente. Gracias a esto podemos definir la apariencia de cada elemento o grupo de ellos con total comodidad y facilidad. También distintas apariencias que sean precisas en función del medio por el que se mostrarán, y cambiarlas posteriormente, si es necesario, de una forma muy simple y rápida ***(www.psicobyte.com)***.

La sintaxis básica del CSS es muy simple. Presenta en su estructura tres elementos de uso obligatorio que son los selectores, las propiedades y los valores, es decir, consta de una serie de reglas que describen la forma en que se visualiza cada uno de los elementos. (**Figura 2**)

Selector {Propiedad: Valor ;}

Por ejemplo: h1 {color: black;}

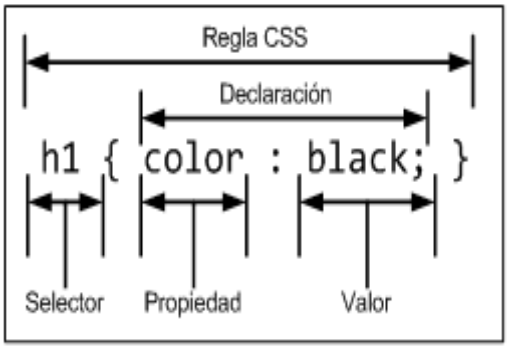

FIGURA 2 COMPONENTES DE UN ESTILO CSS BÁSICO

.

H1 es el selector que indica a que elemento HTML está apuntando. **Color** representa la propiedad a la que se le asigna un valor que sería **black**, o sea, lo que dice la regla de arriba es que las cabeceras aparezcan de color negro.

Los diferentes componentes de una regla CSS se definen a continuación:

▪ *Regla*: cada uno de los estilos que componen una hoja de estilos CSS. Cada regla está compuesta de una parte de *"selectores"*, un símbolo de *"llave de apertura"* ({), otra parte denominada *"declaración"* y por último, un símbolo de *"llave de cierre"* (}).

▪ *Selector*: indica el elemento o elementos HTML a los que se aplica la regla CSS.

▪ *Declaración*: especifica los estilos que se aplican a los elementos. Está compuesta por una o más propiedades CSS.

▪ *Propiedad*: permite modificar el aspecto de una característica del elemento.

▪ *Valor*: indica el nuevo valor de la característica modificada en el elemento.

Otro ejemplo de selectores lo tenemos en las clases **(class)** y los identificadores **(id).** La clase es el nombre que se le asigna a una etiqueta HTML para luego poder hacer referencia a ella. Gracias a las clases podemos asignar propiedades a una parte de los selectores de un mismo tipo. En el caso de los identificadores funcionan igual que las clases, pero con la diferencia de que se pueden emplear una vez en un documento HTML.

El siguiente ejemplo muestra una página HTML con estilos definidos sin utilizar CSS:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Ejemplo de estilos sin CSS</title>
</head>
<body>
<h1><font color="red" face="Arial" size="5">Titular de la página</font></h1>
<p><font color="gray" face="Verdana" size="2">Un párrafo de texto no muy largo.</font></p>
</body>
</html>
```

Aquí se utiliza la etiqueta <font> con sus atributos *color*, *face* y *size* para definir el color, la

tipografía y el tamaño del texto de cada elemento del documento. El principal problema de esta forma de definir el aspecto de los elementos se puede ver claramente con el siguiente ejemplo: si la página tuviera 50 elementos diferentes, habría que insertar 50 etiquetas <font>. Si el sitio web entero se compone de 10.000 páginas diferentes, habría que definir 500.000 etiquetas <font>. Como cada etiqueta <font> tiene 3 atributos, habría que definir 1.5 millones de atributos ***(Javier Eguíluz Pérez 2009).***

La solución que propone CSS es mucho mejor, como se puede ver en el siguiente ejemplo:

```
<! DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Ejemplo de estilos con CSS</title>
<style type="text/css">
h1 { color: red; font-family: Arial; font-size: large; }
p { color: gray; font-family: Verdana; font-size: medium; }
</style>
</head>
<body>
<h1>Titular de la página</h1>
<p>Un párrafo de texto no muy largo. </p>
</body>
</html>
```

Con el HTML se limitaba el maquetado de las páginas por lo que había que acudir al uso de algunos trucos para lograr dar efectos determinados. En cambio, con las CSS se tiene herramientas que permiten dar determinadas formas.

A continuación se mencionan alguna de ellas por citar algunas:

Definir la distancia entre líneas del documento.

Aplicar identado a las primeras líneas del párrafo.

Colocar elementos en la página con mayor precisión, y sin lugar a errores.

Definir la visibilidad de elementos como márgenes, subrayados, tachados, etc.

Existen tres caminos diferentes para aplicar las reglas de estilo CSS a una página Web:

1. **Una hoja de estilo externa**, es una hoja de estilo que está almacenada en un archivo diferente de archivo donde se almacena el código HTML de la página Web. Esta forma es la más potente de programar, porque separa completamente las reglas de formateo para la página HTML de la estructura básica de la página.

2. **Una hoja de estilo interna**, que es una hoja de estilo que está incrustada dentro de un documento HTML. De esta manera, se obtiene el beneficio de separar la información del estilo, del código HTML propiamente dicho. En general, la única vez que se usa una hoja de estilo interna, es cuando se quiere proporcionar alguna característica a una página Web en un simple fichero.

3. **Un estilo en línea**, que es un método para insertar el lenguaje de estilo de página, directamente, dentro de una etiqueta HTML. Este modo de proceder no es totalmente adecuado. El incrustar la descripción del formateo dentro del documento de la página Web, a nivel de código se convierte en una tarea larga, tediosa y poco elegante de resolver el problema de la programación de la página. Este modo de trabajo se podría usar de manera ocasional si se pretende aplicar un formateo con prisa, al vuelo. No es todo lo claro o estructurado que debería ser pero no obstante funciona.

En el primero de los casos todos los estilos se incluyen en un archivo de tipo CSS (archivo simple de texto con extensión .css) que luego las páginas de tipo HTML conectan a través de la etiqueta <link>. Es preciso hacer notorio que pueden ser creados tantos archivos .css como sean necesarios y que una página HTML puede enlazar la cantidad de archivos de este tipo que sea pertinente para lograr el efecto deseado.

Cuando un navegador carga una página HTML, en conjunto descarga los archivos .css externos enlazados mediante la etiqueta <link >, antes de que sea mostrado el contenido de la página para así dar los estilos a los contenidos. La principal ventaja es que se puede incluir un mismo archivo CSS en multitud de páginas HTML, por lo que se garantiza la aplicación homogénea de los mismos estilos a todas las páginas que forman un sitio web.

Generalmente, la etiqueta <link> posee algunos atributos cuando se realiza el enlace hacia un archivo .css los cuales son:

- *rel*: indica el tipo de relación que tiene el recurso enlazado (en este caso, el archivo .css) y la página HTML. Para los archivos CSS, siempre se utiliza el valor stylesheet.
- *type*: indica el tipo de recurso enlazado. Sus valores están estandarizados y para los archivos CSS su valor siempre es text/css.
- *href*: indica la URL del archivo CSS que contiene los estilos. La URL indicada puede ser relativa o absoluta y puede apuntar a un recurso interno o externo al sitio web.
- *media*: indica el medio en el que se van a aplicar los estilos del archivo CSS.

Pero la única forma de realizar este proceso no es mediante la etiqueta <link>; se puede hacer referencia a archivos CSS externos usando la etiqueta <style>. Un ejemplo de ellos se tiene a continuación:

```
<! DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Ejemplo de estilos CSS en un archivo externo</title>
<style type="text/css" media="screen">
@import '/css/estilos.css';
</style>
</head>
<body>
<p>Un párrafo de texto. </p>
</body>
</html>
```

En este caso, para incluir en la página HTML los estilos definidos en archivos CSS externos se utiliza una regla especial de tipo @import. Las reglas de tipo @import siempre preceden a cualquier otra regla CSS (con la única excepción de la regla @charset). La URL del archivo CSS externo se indica mediante una cadena de texto encerrada con comillas simples o dobles o mediante la palabra reservada url() ***(Javier Eguíluz Pérez 2009).***

De esta forma, las siguientes reglas @import son equivalentes:

@import '/css/estilos.css';

@import "/css/estilos.css";

```
@import url('/css/estilos.css');
@import url("/css/estilos.css");
```

1.4 Lenguaje de programación Python

Es preciso hacer referencia al lenguaje de programación Python, debido a que el Motor de Clasificación Inteligente por Contenidos (MOCIC) se desarrolla usando el mismo. Por lo tanto, el conocimiento de sus orígenes y algunas características no estría demás a modo de información.

1.4.1 ¿Qué es Python?

Siendo un lenguaje de programación un idioma artificial para expresar procesos algorítmicos que creen, describan y transformen información y que puedan ser ejecutados por computadoras, se usa para la creación de programas que controlen tanto el funcionamiento físico como lógico en una máquina. Tal es el caso de Python, considerado limpio y elegante a la hora de programar. Es fácil de aprender y aunque sencillo, también potente, orientado a objetos y de scrpits, es decir, instrucciones que se ejecutan paso a paso, instrucción por instrucción. Debido a esta última cualidad no genera ejecutables sino que es el mismo Python el encargado de ejecutar el código. Esta propiedad hace que sea un lenguaje interpretado y no compilado. La ventaja que trae es que se pueda escribir un programa, ser salvado y ejecutado mientras que en un lenguaje compilado hay que pasar por los pasos de compilar y ligar el software haciendo el proceso más lento. El objetivo principal de este lenguaje es brindar facilidad tanto de lectura como de diseño.

El creador del lenguaje es un europeo llamado Guido Van Rossum. El nombre del lenguaje proviene de la afición de este hombre por los geniales humoristas británicos Monty Python. Elobjetivo principal de su creador era cubrir la necesidad de un lenguaje orientado a objetos de uso sencillo que se utilizara para dar soluciones a múltiples tareas de la programación que habitualmente se hacían en Unix utilizando C. El desarrollo del mismo duró varios años, lo que no fue un trabajo de la noche a la mañana. Ya en el año 2000 disponía de un producto bastante completo y contaba con un equipo de desarrollo. Python ha sido usado para crear programas tan famosos como el gestor de listas de correo Mailman y los gestores de contenido Zope y Plone.

1.4.2 Características del Lenguaje de Programación Python

Propósito general

Se pueden crear todo tipo de programas. No es un lenguaje creado específicamente para la Web, aunque entre sus posibilidades sí se encuentra el desarrollo de páginas.

Multiplataforma

El código puede funcionar en cualquier arquitectura sólo con la condición de que se disponga del intérprete del lenguaje lo que hace que sea multiplataforma. Dispone de estructuras de datos de alto nivel y una solución de programación orientada a objetos aunque es un lenguaje multiparadigma y no hay por qué acogerse a un solo estilo sino que también se pude explotar la programación estructurada y funcional.

Interpretado

No se debe compilar el código antes de su ejecución. En realidad sí que se realiza una compilación, pero la misma ocurre de manera transparente para el programador. En ciertos casos, cuando se ejecuta por primera vez un código, se producen unos bytecodes que se guardan en el sistema y que sirven para acelerar la compilación implícita que realiza el intérprete cada vez que se ejecuta el mismo código.

Interactivo

Dispone de un intérprete por línea de comandos en el que se pueden introducir sentencias. Cada una se ejecuta y produce un resultado visible, que puede ayudarnos a entender mejor el lenguaje y probar los resultados de la ejecución de porciones de código rápidamente.

Orientado a Objetos

La programación orientada a objetos está soportada en Python y ofrece en muchos casos una manera sencilla de crear programas con componentes reutilizables.

Funciones y librerías

Dispone de muchas funciones incorporadas en el propio lenguaje, para el tratamiento de strings, números, archivos, etc. Además, existen muchas librerías que podemos importar en los programas para tratar temas específicos como la programación de ventanas o sistemas en red o crear archivos comprimidos en .zip.

Sintaxis clara

Tiene una sintaxis muy visual, gracias a una notación identada (con márgenes) de obligado cumplimiento. En muchos lenguajes, para separar porciones de código, se utilizan elementos como las llaves o las palabras clave **begin** y **end**. Para separar las porciones de código en Python se debe tabular hacia dentro, colocando un margen al código que iría dentro de una

función o un bucle. Esto ayuda a que todos los programadores adopten las mismas notaciones y que los programas de cualquier persona tengan un aspecto muy similar.

1.5 Representación de Documentos

La representación de documentos es el primer paso a realizar a la hora de buscar la información deseada por un usuario, por tanto deberá ser fiel al contenido del documento incluyendo la información útil que se espera obtener. Es la forma de presentar un documento de manera tal que sirva de entrada para los algoritmos que posteriormente los MB emplean para acceder a la información. Los documentos pueden ser representados de forma autocontenida, trayendo consigo que no se utilice ninguna información proveniente de otras páginas, sino de sí misma, evitando así tener que controlar alguna información que generalmente suele ser necesaria en la mayoría de las representaciones que se utilizan en el ámbito de la RI como las frecuencias de los diferentes términos de un vocabulario en los documentos de una colección.

En el caso de páginas Web, y en el límite, esta dependencia externa implicaría considerar el total de los documentos contenidos en la Web o, al menos, un subconjunto suficientemente significativo del ámbito de aplicación de la representación. De este modo, cualquier representación autocontenida resultará completamente independiente del tamaño actual y futuro de la Web, así como de su estructura, ya que no se tendrá que analizar ningún otro documento para representar uno dado. Además, podría aplicarse en sistemas sin necesidad de contar con enormes medios de almacenamiento ni de procesamiento, ni tampoco requerirían una exploración intensiva de colecciones de documentos correlacionados.

Históricamente, los primeros pasos que se dieron en representación de páginas Web trasladaron directamente técnicas que se habían aplicado hasta ese momento en la representación de textos, empleándose, por tanto, representaciones basadas en contenido. De este modo, las frecuencias de aparición de los términos en la propia página, así como las frecuencias inversas del documento (el número de documentos dentro de la colección en los que aparece un determinado rasgo), constituían la base de estas primeras representaciones. También se aplicaron algunas técnicas basadas en el análisis de algunas etiquetas HTML, pero pronto se introdujeron en las representaciones elementos propios de los documentos Web (información de hiperenlaces inter-documento), aplicándose técnicas basadas principalmente en un análisis de correlaciones y desarrollándose, de este modo representaciones basadas en

estructura. Este tipo de representaciones, fundamentalmente aplicadas en el campo de la RI, pasaron a ser predominantes frente a las representaciones por contenido no sólo en este campo, sino también en problemas de Clasificación automática de textos o Agrupación de documentos ***(Víctor Fresno Fernández 2005).***

1.5.1 Modelo de Espacio Vectorial

Espacio Vectorial (EV), es sinónimo de Espacio Lineal (objeto de estudio del Álgebra Lineal) donde a cada uno de los elementos que lo conforman se le llaman vectores, sobre los cuales pueden realizarse dos operaciones básicas que son: escalarse (multiplicar por un escalar) y sumarse.

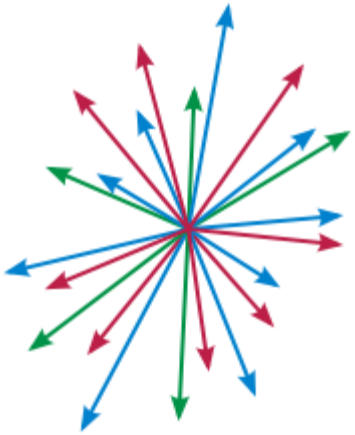

FIGURA 3 REPRESENTACIÓN DE UN ESPACIO VECTORIAL

Existen algunas técnicas de representación de documentos, de distintos tipos, que han sido aplicadas al campo de la RI y es precisamente el Modelo de Espacio Vectorial una dentro de todo el grupo de técnicas que además han sido muy empleadas en clasificación automática de textos y agrupación de documentos en los últimos años.

El Modelo de espacio vectorial no es más que un modelo algebraico utilizado para filtrado, recuperación, indexado y cálculo de relevancia de información. Representa documentos en lenguaje natural de una manera formal mediante el uso de vectores en un espacio lineal multidimensional. Así llámensele entonces modelos de representación vectorial a estas representaciones.

Las representaciones vectoriales resultan muy sencillas y descansan sobre la premisa de

que el significado de un documento puede derivarse del conjunto de términos presentes en el mismo. Pueden considerarse “basados en términos” o características, los cuales serán, de un modo u otro, los vectores generadores de un espacio vectorial.

Los primeros trabajos de representación vectorial surgen en el entorno de la clasificación de documentos. Aquellas representaciones, muy sencillas, se basaban en que el significado de un documento podía atribuirse del conjunto de términos presentes en él, así que realizando la representación de una forma manual como se hacía entonces, un documento pertenecía a una clase si presentaba términos específicos que anteriormente se etiquetaban como pertenecientes a esa misma clase. Así estos modelos podían considerarse basados en términos que serían los vectores generados de un espacio vectorial.

La idea básica de este modelo reside en la construcción de una matriz, de términos y documentos, donde las columnas representan los términos incluidos en los documentos y las filas los documentos. Así a las filas se les llaman en términos algebraicos vectores, que serían equivalentes a los documentos que se expresarían en función de las apariciones (frecuencia) de cada término. De esta forma, un documento podría expresarse mirándolo a través de las filas de la siguiente forma como se observa en la Tabla 1. D1 (1, 1, 1, 1, 0, 0, 0, 0), siendo cada uno de estos valores el número de veces que se repite el término en el documento expuesto. La longitud del vector estaría dada por la cantidad de columnas que tenga la matriz.

El cálculo de la similitud entre una consulta y los documentos disponibles se realiza calculando cuál de los documentos tiene más elementos en común con la consulta que se introdujo. Existen diversas maneras de efectuar ese cálculo.

TABLA 1 REPRESENTACIÓN DE LA MATRIZ DE ESPACIO VECTORIAL

	Río	**Danubio**	**Viena**	**Color**	**Azul**	**Caudal**	**Invierno**	**Rhin**	**Navegable**
D1	1	1	1	1	1	0	0	0	0
D2	1	0	0	0	0	1	1	0	0
D3	2	1	0	0	0	1	0	1	0
D4	1	0	0	0	0	1	0	0	1

Una de las más sencillas consiste en el sumatorio de los productos. En resumen, se multiplican los valores de los términos de la consulta por cada uno de los valores que

contengan los vectores en cada columna, sumando el resultado final. Después de realizar esta operación se toman los resultados que presenten un mayor valor numérico ***(Luís Codina 2005).***

En tareas de clasificación automática de textos y agrupación de documentos, pasan a ser representados como vectores dentro de un espacio euclídeo (espacio vectorial normado de dimensión finita en que la norma es heredada de un producto escalar), de forma que midiendo la distancia entre dos vectores se trata de estimar su similitud como indicador de cercanía semántica. Los documentos se modelan como conjuntos de términos que pueden ser individualmente tratados y pesados. De este modo, en el caso de la RI basta con representar las consultas (queries) del mismo modo que se representaría cualquier documento que contuviera los términos presentes en dicha query. Después se calcula la distancia entre el vector de consulta y el conjunto de documentos presentes en la colección que se esté considerando. En este contexto, los modelos vectoriales permiten un emparejamiento parcial entre los documentos y el vector de consulta.

El modelo de espacio vectorial, se caracteriza, fundamentalmente, porque asume el "principio de independencia", por el que se considera que las cadenas aparecidas en un mismo texto no tienen relación entre sí, pudiendo ser cuantificadas individualmente; además, no tiene en cuenta el orden en el que aparecen en el texto. De este modo, la semántica de un documento queda reducida a la suma de los significados de los términos que contiene.

En la mayoría de los casos, estos modelos no tratan de reducir las dimensiones del espacio, colapsándolas en un subconjunto más reducido, y consideran cada término como un objeto independiente. A pesar de esto, no son simples ficheros invertidos que guardan información de relación entre términos y documentos, sino que representan modelos más flexibles, al permitir realizar el pesado de cada término individualmente, de forma que este pueda considerarse más o menos importante dentro de un documento o del global de la colección. Por ejemplo, el producto escalar entre dos vectores mide la similitud entre dos documentos como la distancia euclídea entre los vectores de representación. En algunos problemas que tratan de encontrar la similitud semántica entre documentos, las direcciones de los vectores dentro de un espacio son indicadores más eficaces que la distancia euclídea entre ambos ***(Baeza-Yates y Ribeiro-Neto).***

El Modelo de Espacio Vectorial es la representación abstracta de documentos y consultas, que permite definir las características de ambos, así como calcular el grado de semejanza o similitud entre unos y otros. Según este modelo se pueden representar como un vector cada

una de las expresiones de lenguaje natural que pueda contener un documento dado. Representa cada documento en una colección mediante un vector de n elementos, donde n es el número de términos indizables susceptibles de ocurrir o aparecer en cualquier elemento de la colección. Mediante diferentes sistemas de cálculo, a cada término o cada elemento del vector de cada uno de los documentos se le asigna un valor numérico o peso, que pretende significar la importancia o valor informativo de ese término en ese documento. Una consulta dada, formulada en lenguaje natural, puede representarse también mediante el mismo sistema, es decir, mediante otro vector de los mismos n elementos, cada uno de los cuales contiene el peso de cada uno de los términos de dicha consulta. En consecuencia, es fácil calcular cualquiera de las funciones utilizadas habitualmente para establecer la similitud entre dos vectores, aplicándola entre el vector de la consulta y los de cada uno de los documentos. El resultado es un valor numérico que pretende indicar el grado de ajuste o semejanza entre la consulta y cada uno de los documentos; de forma que, aquellos documentos que arrojen una cifra más alta, serán los que más se ajusten a la consulta formulada (Figura 4).

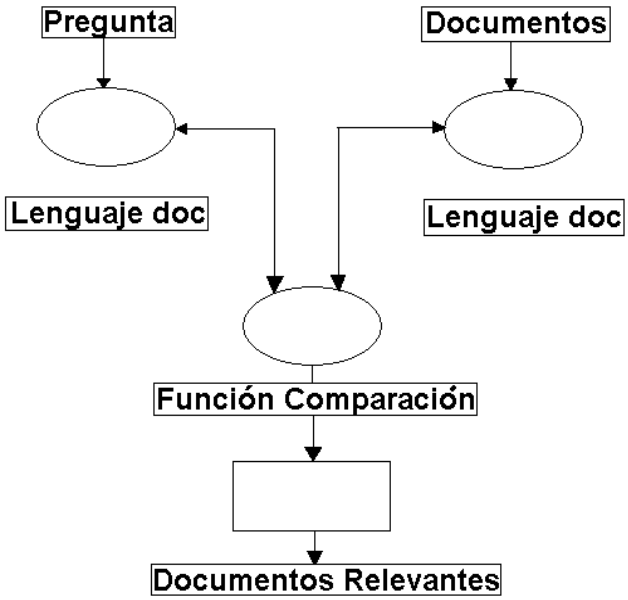

FIGURA 4 PROCESO DE DETECCIÓN DE DOCUMENTOS RELEVANTES

.

1.6 Clasificación Automática de Documentos

El gran cúmulo de información textual que está disponible en Internet en la actualidad, conlleva a que a diario las personas en busca de conocimiento naveguen a través de la Web, por lo que es el recurso más importante para el desarrollo de investigaciones .Sin embargo, al no existir una entidad central que la administre, y debido al continuo incremento de información, se hace cada vez más difícil la búsqueda de documentos con información relevante para un usuario.

Existen potentes herramientas llamadas motores de búsqueda que son utilizadas por los usuarios para acceder a la información realizando consultas que permiten que la misma sea organizada en función de categorías, títulos o contenido. Aparece así la necesidad de aludir al término “clasificar”, muy conocido por todas las personas y utilizado con mayor frecuencia por quienes operan con documentos, para quienes es indispensable organizarlos de forma tal que puedan ser localizados en su debido momento de la forma más amena posible. Con la creciente disponibilidad de documentos en formato electrónico, susceptibles, por consiguiente, de ser procesados de manera automática, surge la posibilidad de abordar la clasificación de documentos de manera automática.

La clasificación automática de documentos puede interpretarse como un proceso de "aprendizaje matemático-estadístico", durante el cual un algoritmo implementado computacionalmente capta las características que distinguen cada categoría o clase de documentos de las demás, es decir, aquellas que deben poseer los documentos para pertenecer a esa categoría. Estas características no tienen por qué indicar de forma absoluta e inequívoca la pertenencia a una clase o categoría, sino que más bien lo hacen en función de una escala o graduación. De esta forma, por ejemplo, documentos que posean una cierta característica tendrán un factor de posibilidades de pertenecer a determinada clase, de modo que la acumulación de dichas características arrojará un resultado que consiste en un coeficiente asociado a cada una de las clases ya conocidas. Este coeficiente lo que expresa en realidad es el grado de confianza o certeza de que el documento en cuestión pertenezca a la clase asociada al coeficiente resultante ***(René Venegas 2007).***

La clasificación automática de documentos no es más que la tarea en que un documento, o una parte del mismo, son etiquetados como pertenecientes a un determinado conjunto, grupo o categoría predeterminada. Por lo general, cuando se habla de clasificación automática se distingue entre dos escenarios diferentes, que obviamente, requieren soluciones distintas. Estos escenarios reciben diversos nombres, pero básicamente consisten en lo siguiente:

Clasificación supervisada o categorización

Situación en la que se parte de una serie de clases o categorías conceptuales prediseñadas a priori, y en la que labor del clasificador (manual o automático) es asignar cada documento a la clase o categoría que le corresponda conociéndose así como clasificación supervisada, no sólo porque requiere la elaboración manual o intelectual del cuadro o esquema de categorías, sino también, porque requiere un proceso de aprendizaje o entrenamiento por parte del clasificador, que debe ser supervisado manualmente en mayor o menor medida(Figura 5).

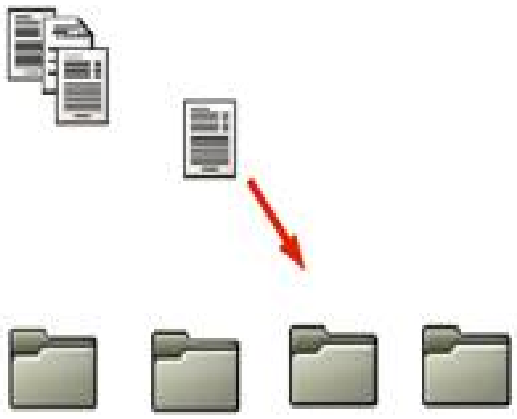

FIGURA 5 CLASIFICACIÓN SUPERVISADA

Clasificación no supervisada

En el segundo escenario posible, no hay categorías previas ni esquemas o cuadros de clasificación establecidos a priori. Los documentos se agrupan en función de sí mismos, de su contenido; de alguna manera, podemos decir que se autorganizan. Es lo que se conoce como clasificación (automática) no supervisada o clustering; no supervisada porque se efectúa de forma totalmente automática, sin supervisión o asistencia manual.

En muchas ocasiones el resultado obtenido después de realizadas las consultas no se corresponden con la información de más relevancia aunque se haya cumplido con los criterios de búsqueda especificados y en torno a ello se han elaborado diversas técnicas, que se han aplicado con mayores o menores resultados. En la tesis doctoral Representación Autocontenida de Documentos HTML: una propuesta basada en Combinaciones Heurísticas

de Criterios de Víctor Fresno, se plantea que representar documentos HTML supone el punto de partida en la aplicación de cualquier técnica de clasificación automática de documentos, siendo necesario así que se transforme un documento desde su formato inicial a una forma que sea la más propicia a la entrada de los algoritmos que emplean los MB para realizar posteriormente todas las tareas necesarias en el proceso de clasificación automática.

Existe gran cantidad de algoritmos que son utilizados para la clasificación automática de documentos. La mayor parte de ellos no fueron creados sólo para clasificar documentos, sino que se han propuesto para clasificar todo tipo de objetos ***(Carlos G. Figuerola, José L. Alonso Berrocal, Angel F. Zazo Rodríguez, Emilio Rodríguez 2005)***.

1.6.1 Clasificación Probabilística Naive Bayes

Los algoritmos probabilísticos son aquellos que se basan en el teorema de Bayes donde el mismo permite estimar la probabilidad de un suceso a partir de que ocurra otro suceso, del cual depende el primero, basando así sus resultados en decisiones aleatorias, de esta forma como promedio se obtiene una buena solución al problema planteado ***(J. Campos).***

El algoritmo más simple conocido de este tipo es el *Naive Bayes* que básicamente da la probabilidad de que un documento pertenezca a una clase o categoría determinada, dependiendo de una serie de características de cada una de las cuales conocemos la probabilidad de que aparezcan en la categoría en cuestión. Estas características son los términos que contienen los documentos, por tanto, la medida de aparición de los términos en un documento como la probabilidad de que aparezcan en una determinada categoría puede obtenerse a partir de los documentos de entrenamiento, para ello se utiliza la repetición del mismo en la colección de entrenamiento.

1.6.2 Clasificación Basada en el Modelo de Espacio Vectorial

En la clasificación automática de documentos es muy utilizado el modelo de espacio vectorial dada las grandes ventajas que proporciona que se explican en el epígrafe anterior. A continuación se mencionan dos algoritmos de los más utilizados basados en el modelo de espacio vectorial.

Algoritmo de Rocchio:

Este algoritmo es bien conocido y aplicado en la realimentación de consultas, es decir, el

usuario determina cuales documentos son importantes y cuáles no, después de que realiza una consulta. Con el resultado de la selección el sistema genera una nueva consulta en función de cuáles fueron relevantes y cuáles no. Así el algoritmo proporciona un sistema para construir el vector de la nueva consulta, recalculando los pesos de los términos de la misma y aplicando un coeficiente a los pesos de la consulta inicial, otro a los de los documentos relevantes y otro distinto de los no relevantes.

En el ámbito de la categorización, el mismo algoritmo de Rocchio proporciona un sistema para construir los patrones de cada una de las clases o categorías de documentos. Así, partiendo de una colección de entrenamiento, categorizada manualmente de antemano, y aplicando el modelo vectorial, podemos construir vectores patrón para cada una de las clases, considerando como ejemplos positivos los documentos de entrenamiento de esa categoría, y como ejemplos negativos los de las demás categorías (Figura 6).

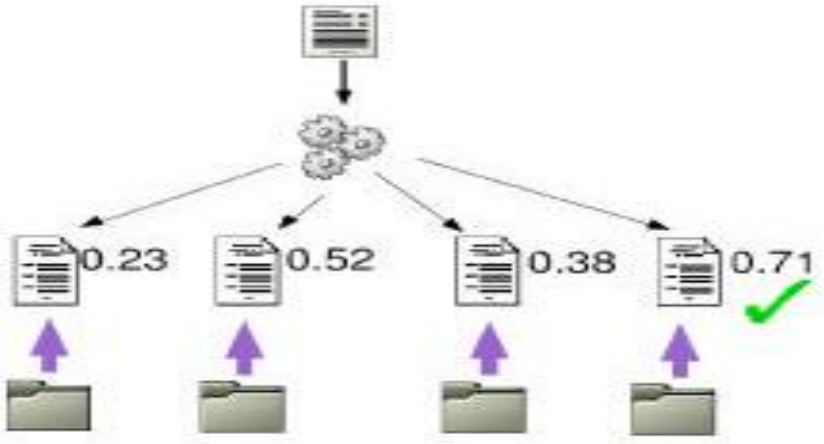

FIGURA 6 FORMAS DE CONSTRUIR PATRONES DE CADA CLASE

Una vez que se tienen los patrones de cada una de las clases, el proceso de entrenamiento o aprendizaje está concluido. Para categorizar nuevos documentos, simplemente se estima la similitud entre el nuevo documento y cada uno de los patrones. El que arroja un índice mayor nos indica la categoría a la que se debe asignar ese documento ***(Carlos G. Figuerola, José L. Alonso Berrocal, Angel F. Zazo Rodríguez, Emilio Rodríguez 2005).***

Algoritmo del vecino más cercano (Nearest Neighbour, NN) y variantes:

La idea básica consiste en que si se calcula la similitud entre el documento a clasificar y cada uno de los documentos de entrenamiento, al cual sea más parecido estará indicando a

qué clase o categoría se asigna el documento que se desea clasificar. Una vez localizado el documento de entrenamiento más similar, dado que han sido previamente categorizados manualmente, se sabe a qué categoría pertenece y, por ende, a qué categoría se asigna el documento que se clasifica.

Una de las variantes más conocidas de este algoritmo es la del k-nearest neighbour o *KNN* que consiste en tomar los *k* documentos más parecidos, en lugar de sólo el primero. Como en esos *k* documentos habrá, seguro, de varias categorías, se suman los coeficientes de los de cada una de ellas. La que más puntos acumule, será la candidata idónea. El *KNN* une a su sencillez una eficacia notable. Obsérvese que el proceso de entrenamiento no es más que la indización o descripción automática de los documentos, y que tanto dicho entrenamiento como la propia categorización pueden llevarse a cabo con instrumentos bien conocidos y disponibles para cualquiera. KNN parece especialmente eficaz cuando el número de categorías posibles es alto, y cuando los documentos son heterogéneos y difusos ***(Yuan Jiang and Zhi-hua Zhou 2004)***.

1.6.3 Clasificación Basada en Árboles de Decisión

Un árbol de decisión tiene unas entradas que pueden ser un objeto o una situación descrita por medio de un conjunto de atributos y a partir de aquí devuelve una respuesta que en últimas es una decisión que es tomada a partir de las entradas.

Los valores que pueden tomar las entradas y las salidas pueden ser discretos o continuos. Se utilizan más los valores discretos por simplicidad, cuando se utilizan en las funciones de una aplicación se denomina clasificación y cuando se utilizan los continuos se denomina regresión. Un árbol de decisión lleva a cabo un test a medida que este se recorre hacia las hojas para alcanzar así una decisión.

El árbol de decisión suele contener nodos internos, nodos de probabilidad, nodos hojas y arcos. Un nodo interno contiene un test sobre algún valor de una de las propiedades. Un nodo de probabilidad indica que debe ocurrir un evento aleatorio de acuerdo con la naturaleza del problema, este tipo de nodos es redondo, los demás son cuadrados. Un nodo hoja representa el valor que devolverá el árbol de decisión. Y finalmente las ramas brindan los posibles caminos que se tienen de acuerdo con la decisión tomada ***(Álvaro González Fernández 2007).***

Algoritmo ID3

Este algoritmo es muy utilizado en el campo de la inteligencia artificial. El mismo genera

árboles de decisión a partir de ejemplos de partida. Su uso se basa en la búsqueda de hipótesis o reglas en dado un conjunto de ejemplos. El conjunto de ejemplos deberá estar conformado por una serie de tuplas (filas) de valores, cada uno de ellos denominados atributos, en el que uno (el atributo a clasificar) es el objetivo, el cual es de tipo binario (positivo o negativo, si o no, válido o inválido, etc.). De esta forma, el algoritmo trata de obtener las hipótesis que clasifiquen ante nuevas instancias, si dicho ejemplo va a ser positivo o negativo.

ID3 realiza esta labor mediante la construcción de un árbol de decisión. Los elementos que lo conforman son los nodos, los cuales van a contener los atributos y los arcos que contendrán los posibles valores del nodo padre y las hojas que son los nodos que clasifican el ejemplo como positivo o negativo.

1.6.4 Clasificación Basada en Redes Neuronales

Las redes neuronales tienen gran utilización en la clasificación automática de documentos ya que una de sus principales aplicaciones es en el reconocimiento de patrones. Las mismas constan de varias capas de neuronas interconectadas entre ellas. La capa de entrada recibe términos mientras que las neuronas de la capa de salida mapean clases o categorías (Figura 7). Las interconexiones entre las mismas tienen pesos, lo que expresa la mayor o menor fuerza de la conexión. Es posible entrenar una red para que dada una entrada determinada, produzca la salida deseada. El proceso de entrenamiento consta de un ajuste de los pesos de las interconexiones, a fin de que la escogida sea la que en verdad se desea ***(Carlos G. Figuerola, José L. Alonso Berrocal, Angel F. Zazo Rodríguez, Emilio Rodríguez 2005)***.

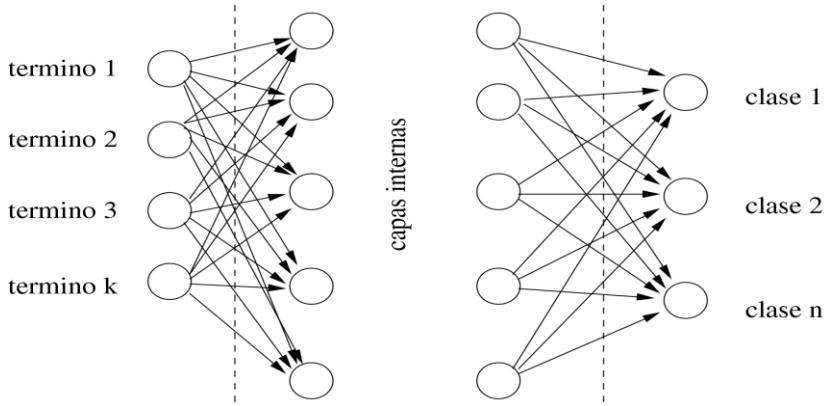

FIGURA 7 EJEMPLO DE RED NEURONAL PARA LA CLASIFICACIÓN AUTOMÁTICA

1.7 Funciones de Ponderación de Términos

Como parte del proceso de clasificación automática de documentos, necesario para que el acceso a la información en formato electrónico disponible en Internet sea más ajustable a las necesidades de los usuarios, es preciso hacer referencia a las funciones de ponderación de términos, las cuales se basan fundamentalmente en un conteo de frecuencias, ya sea en el conjunto de documentos de una colección o dentro del documento a representar. La mayoría dan la posibilidad de en conjunto organizar un grupo de términos que luego se reducen a un vocabulario mediante el cual se representa un documento. El término ponderar significa: dar valor, dar peso, por lo que obviamente estas funciones son utilizadas para determinar la relevancia que tienen los términos o rasgos en un documento.

Si se consideran las palabras como características definitorias del documento, el proceso que debe seguir el sistema de clasificación se inicia con la selección de aquellas palabras útiles que permitan discriminar unos documentos de otros. En este punto, debemos señalar que no todas las palabras contribuyen con la misma importancia en la caracterización del documento. Desde el punto de vista lingüístico aplicado a la recuperación o clasificación de documentos, existen lexemas casi vacíos de contenido semántico, como los artículos, las preposiciones o las conjunciones. Estos lexemas son conocidos como palabras funcionales en la tradición lingüística y como *stop words* en el procesamiento de lenguaje natural. Estas palabras, que en español comúnmente son entre 100 y 200, son poco útiles para el proceso de clasificación. También son poco importantes aquellas palabras que por su frecuencia de aparición en toda la colección de documentos pierden su poder de discriminación, es por ello que o son eliminadas o son ponderadas con muy bajo peso estadístico ***(René Venegas 2007).***

Pero como una palabra puede aparecer en más de una ocasión en un documento y por supuesto unas tendrán más peso estadístico que otras, el valor numérico de cada uno de los componentes del vector analizando con el modelo vectorial, obedece a cálculos más sofisticados que la simple asignación binaria. Por otra parte, entonces es importante normalizar los vectores para que no haya documentos privilegiados.

Indagando en literatura relacionada al tema se encuentran gran cantidad de estas funciones, las cuales básicamente se pueden distinguir con claridad en dos grupos que se ejemplifican en los epígrafes siguientes.

1.7.1 Funciones de Ponderación de Carácter Local

Son conocidas como funciones de ponderación de carácter local aquellas que toman únicamente información del propio documento para obtener una representación, sin necesidad de ninguna información externa. Es importante resaltar que como consecuencia de esta definición, cuando una función local se aplica a la representación de un documento HTML se obtiene, necesariamente, una representación autocontenida.

Función de ponderación ACC (Analytical Combination of Criteria)

Es un método de representación el cual se basa en combinaciones analíticas de criterios a tomar en cuenta son, la frecuencia de aparición de un término, si se encuentra en el título, si está enfatizado y la posición que ocupa en el texto ***(Víctor Fresno Fernández 2005).***

Función Binaria (Binary, Bin)

Es el método de representación más sencillo, dentro de los modelos de representación vectorial, es el conocido como conjunto de palabras o espacio vectorial binario ***(Víctor Fresno Fernández 2005)***.

Frecuencia de Aparición (Term Frecuency, TF)

Es la representación más sencilla dentro de los modelos no binarios, conocida como Bolsa de palabras (bag of words). La relevancia se representa por la frecuencia de aparición del rasgo en el documento ***(Víctor Fresno Fernández 2005)***.

Frecuencia aumentada y normalizada (Augmented Normalized Term Frecuency, ANTF)

Esta función representa una frecuencia normalizada de un rasgo en un documento. La normalización se realiza con la mayor de las frecuencias presentes en el documento ***(Víctor Fresno Fernández 2005)***.

1.7.2 Funciones de Ponderación de Carácter Global

Son conocidas como funciones de ponderación de carácter global aquellas que toman información de una colección para generar las representaciones. Si una función de ponderación consta de una parte local y una global también entraría en esta clasificación.

Frecuencia inversa del documento (Inverse Document Frecency, BINIDF)

Esta función trata de enriquecer la representación binaria suponiendo que los términos que aparecen en muchos documentos de la colección no son tan descriptivos como aquellos que aparecen en unos pocos ***(Víctor Fresno Fernández 2005)***.

Función Normal (N)

Esta función corrige los términos con una frecuencia alta en el conjunto de términos de la colección.

Algo que no se puede dejar de mencionar es que las funciones de ponderación no se orientan a las tareas que se realicen posteriormente, así que una representación que se genere con cualquier función de ponderación va a ser exactamente igual sin importar la tarea que se vaya a realizar a continuación.

1.7.3 Función de Ponderación ACC

Resulta interesante estudiar la función de ponderación de términos propuesta en la tesis doctoral de Víctor Diego Fresno Fernández, Representación Autocontenida de Documentos HTML: una propuesta basada en Combinaciones Heurísticas de Criterios de la universidad Rey Juan Carlos de España. En ese trabajo se propone el método para establecer la relevancia de un término en el contenido de una página Web en función de una combinación lineal de criterios heurísticos (ACC). Se plantea que la forma más sencilla y directa de realizar una combinación de criterios es a través de una función de combinación lineal. Esta función ACC combina linealmente los criterios tomado en cuenta cuatro que son de suma importancia. Esta función de combinación lineal se expresa de la siguiente forma:

$$F(\overrightarrow{t_i}, \overrightarrow{d_j}) = C_1 f_{criterio1}(\overrightarrow{t_i}, \overrightarrow{d_j}) + .. + C_n f_{criterio\ n}(\overrightarrow{t_i}, \overrightarrow{d_j})$$

Donde $F(\overrightarrow{t_i}, \overrightarrow{d_j})$ no es más que la relevancia que presenta un término t_i en un documento d_j.

De esta forma, se puede establecer un peso diferente para cada criterio, así logrando que unos términos pudieran aportar más que otros. Las asignaciones de los pesos para cada criterio fueron las siguientes; frecuencia $C_{fre} = 0.3$, título $C_{tit} = 0.15$, posición $C_{pos} = 0.3$ y enfatizado $C_{enf} = 0.25$. Se definen cuatro funciones de ponderación con respecto a cada uno de los criterios a tomar en cuenta. Todos ellos se describen a continuación:

Frecuencia:

Es la frecuencia de aparición de un término en el documento un factor determinante a la hora de determinar su relevancia. Pero este criterio no se debe tomar de forma aislada, es decir, no se debe medir la relevancia de un término tomando solamente la frecuencia de aparición, ya que esto podría potenciar palabras de uso común, palabras muy utilizadas pero que no permiten distinguir claramente contenidos de documentos con temáticas diferentes. Según la Ley de Zipf los términos de un documento en los que el valor del producto calculado entre su frecuencia de aparición y su posición ordinal estuviera próximo al valor de la constante que determina dicha Ley, serían los candidatos idóneos para la indización del documento. En la práctica resultaba que los términos que aparecen muchas veces en un documento no son candidatos (porque son muy generales) y tampoco aquellos que aparecen pocas veces (porque son muy específicos y su presencia en el documento puede ser anecdótica).

Por lo tanto, se plantea que la función de ponderación para el criterio frecuencia de aparición de un término en un documento es:

$$f_{\text{frecuencia}}(\overrightarrow{t_i}, \overrightarrow{d_j}) = \frac{f_{ij}}{N_j}$$

Donde f_{ij} es la frecuencia de aparición de un término t_i en un documento d_j y N_j la suma de las frecuencias del total de términos presentes en el documento d_j.

Título:

El título de un documento es un aspecto a tomar en cuenta debido a que presenta una relevancia elevada dentro de una página Web, teniendo consigo un resumen del contenido del documento en cuestión. De esta manera, se considera que el criterio título es un factor que se debe tomar en cuenta a la hora de calcular la relevancia de un término en el contenido de un documento.

Por lo tanto, se plantea que la función de ponderación para el criterio título de un término en un documento es:

$$f_{\text{titulo}}(\overrightarrow{t_i}, \overrightarrow{d_j}) = \frac{t_{ij}}{N_{titulo}(j)}$$

Donde t_{ij} es la frecuencia del término t_i en el título de un documento d_j y $N_{titulo}(j)$ la cantidad total de términos en el título del documento d_j.

Posición:

Para el cálculo de relevancia de un término teniendo en cuenta la posición que ocupa dentro de un documento, es un punto a tomar en cuenta, debido a que un término en la introducción de un texto de una medida de la idea a desarrollar en el documento tomando así una relevancia mayor. Mientras que un término tiene una frecuencia total en el documento, el título o las partes enfatizadas, su valor respecto a la posición vendrá dado por el conjunto de posiciones en las que aparezca. Por tanto, se asumió que las páginas Web se consideran como nodos dentro de un grafo de hipertextos, donde como hipótesis se toma que las páginas Web tienen un carácter expositivo. Las páginas Web de carácter expositivo presentan una estructura muy común dividida en tres partes que son, introducción, desarrollo y conclusión.

Independientemente de la estructura antes expuesta, el texto de la misma se divide en cuatro partes, donde los términos que aparezcan en la primera y la cuarta parte del texto tendrán una relevancia mucho mayor que los aparecidos en la segunda y tercera parte del texto en cuestión. Para la experimentación se asignó un peso mayor a los términos que aparezcan en la primera y cuarta con $\frac{3}{4}$, y de $\frac{1}{4}$ a las restantes.

Por lo tanto, se plantea que la función de ponderación para el criterio posición de un término en un documento es:

$$\mathrm{f_{posicion}}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) = \frac{\frac{3}{4}f_{1,4}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) + \frac{1}{4}f_{2,3}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right)}{\sum_{t=1\ldots k}\left(\frac{3}{4}f_{1,4}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) + \frac{1}{4}f_{2,3}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right)\right)}$$

Donde $f_{1,4}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right)$ es la frecuencia del término t_i en posiciones preferentes, es decir, la primera y cuarta parte del texto del documento d_j y $f_{2,3}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right)$ la frecuencia de términos en las posiciones estándar, es decir, segunda y tercera del texto del documento d_j. Donde k presenta la cantidad total de términos diferentes en el documento. Por tanto, la función está normalizada ya que la frecuencia total del término t_i en el documento d_j es $f_{ij} = f_{1,4}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) + f_{2,3}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right)$ por lo tanto, la expresión anterior queda de la siguiente forma:

$$\mathrm{f_{posicion}}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) = \frac{2f_{1,4}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) + f_{ij}}{\sum_{t=1\ldots k}\left(2f_{1,4}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) + f_{ij}\right)} = \frac{2f_{1,4}\left(\overrightarrow{t_i}, \overrightarrow{d_j}\right) + f_{ij}}{2f_{j\,[1,4]} + N_j}$$

Donde $f_{j[1,4]}$ es la suma de las frecuencias totales de los términos t_i presentes en posiciones preferentes del documento d_j y N_j la suma de las frecuencias del total de términos presentes en el documento d_j.

Entonces para la futura aplicación de esta función en cualquier problema de ponderación a las posiciones referentes o estándares, la expresión se generaliza de la siguiente forma:

$$f_{posicion}(\overrightarrow{t_i}, \overrightarrow{d_j}) = \frac{(a+b) f_{1,4}(\overrightarrow{t_i}, \overrightarrow{d_j}) + b f_{ij}}{(a-b) f_{j[1,4]} + b N_j}$$

Siendo **a** el peso aplicado a las posiciones referentes y **b** el peso a las posiciones estándar, poniendo la condición **a** + **b** = 1

Enfatizado:

Esta técnica de enfatizado brinda grandes ventajas al autor debido a que permite destacar determinadas frases o términos. Logrando que el lector se enfoque en estas partes enfatizadas. El texto en esta forma permite una mejor comprensión por parte del lector, logrando que el mismo se lleve la idea de que temas aborda el documento en cuestión. También permite que el texto sea mucho más accesible visualmente, facilitando así una lectura más rápida.

Por lo tanto, se plantea que la función de ponderación para el criterio título de un determinado término en un documento es:

$$f_{enfatizado}(\overrightarrow{t_i}, \overrightarrow{d_j}) = \frac{e_{ij}}{N_{enfatizado}(j)}$$

Donde e_{ij} es la frecuencia del término t_i en el conjunto de términos enfatizados de un documento d_j y $N_{enfatizado}(j)$ la cantidad de términos enfatizados del documento d_j.

De esta función de ponderación se tomará sólo el criterio enfatizado, ya que es el que permite resaltar términos y frases en el cual se quiere hacer énfasis en el texto de un documento dado.

1.8 Conclusiones

En este capítulo han sido expuestos los principales puntos de interés de la investigación. Los conceptos abordados ayudan a la comprensión del papel que juega el proceso de clasificación automática de documentos para la búsqueda y extracción de información así como las funciones de ponderación de términos para con este proceso ligado a la aplicación de los espacios vectoriales como una forma matemática de representación de los documentos.

CAPÍTULO 2 PROPUESTA DE SOLUCIÓN

Introducción

La propuesta a solución de este trabajo va muy ligada a la función de ponderación ACC porque se hará uso de ella para MOCIC. Consiste en un método y es preciso aclarar que el significado que se toma para esta palabra es: serie de pasos sucesivos que conducen a un objetivo. Se toma en cuenta específicamente el criterio "enfatizado" de la ACC porque es sobre el cual influyen las CSS para con el efecto dado a los caracteres de un texto. Se detalla el análisis del texto HTML que se realiza con el uso de la librería LXML, la cual, devuelve las etiquetas Style y Links para ser utilizadas en el posterior parseo a las CSS utilizando la librería CSSutils y así obtener los estilos empleados a través de las CSS a términos en el texto del documento. Durante el desarrollo de este capítulo se explica todo el proceso.

2.1 Arquitectura de MOCIC

El Sistema de Filtrado de Paquetes por Contenido denominado FILPACON permite regular, aceptando o denegando, el acceso de usuarios a contenidos determinados de Internet y así brindar una navegación segura que se ajuste a las políticas de las instituciones en que se utilice. El mismo busca en primer lugar hacer un uso eficiente del canal de internet educando a los usuarios en un buen uso del recurso que debe de ser utilizado para las necesidades básicas de la Universidad de las Ciencias Informáticas que son la docencia, producción e investigación.

Dado el gran volumen que posee Internet actualizar manualmente esta base de datos resulta imposible; para automatizar esta tarea surge el Motor de Clasificación Inteligente por Contenidos, que es el encargado de automatizar el proceso de clasificación de documentos HTML provenientes de Internet. Este motor está conformado por ocho módulos diferentes como se muestra en la figura 2.1. que cumplen una función determinada. Estos módulos son los siguientes (Figura 8):

- Módulo-Clasificador de Texto
- Módulo-Clasificador de Rostros

- Módulo-Clasificador de Desnudez
- Módulo-Clasificador de Objetos
- Módulo-Reconocimiento Óptico de Caracteres
- Módulo-Clasificador de Enlaces
- Módulo-Controlador
- Módulo-Decisor

Módulo-Clasificador de Texto

Tiene como función recibir a través de la consulta de un fichero de solicitudes, un mensaje proveniente del Módulo-Controlador el cual indica la clasificación del texto de una URL. Para dicha clasificación accederá al directorio de localización de la URL, cargará el fichero .html correspondiente, le realizará un pre-procesamiento para identificar el idioma al que pertenece y seguidamente determinará las categorías de contenido previamente definidas (Ciencias, Computadoras, Deporte, Juegos, Pornografía, Violencia,…) asociadas a esta URL, elaborando un mensaje que colocará en un fichero de respuestas. Este mensaje será posteriormente utilizado por el Módulo Decisor, para decidir a qué categoría finalmente pertenece la URL. Pero teniendo en cuenta –como se ha mencionado anteriormente- la introducción de nuevas tecnologías como el CSS se hace de vital importancia el análisis del mismo para obtener una clasificación mucho más confiable.

Módulo-Clasificador de Rostros

Este módulo tiene como función recibir a través de la consulta de un fichero de solicitudes, un mensaje proveniente del MOD-Controlador el cual indica la clasificación de las imágenes de un documento HTML. Para clasificarlo accederá al directorio de localización del documento HTML, cargará el directorio de sus imágenes y le realizará un procesamiento para determinar el número de rostros de personas encontrados en cada imagen. Seguidamente se elabora un mensaje que se colocará en un fichero de respuestas. Este mensaje será posteriormente utilizado por el Módulo Decisor, para decidir a qué categoría finalmente pertenece el documento HTML.

Módulo-Clasificador de Desnudez

Este módulo tiene como función recibir a través de la consulta de un fichero de solicitudes, un mensaje proveniente del MOD-Controlador el cual indica la clasificación de las imágenes de un documento HTML. Para clasificarlo accederá al directorio de localización del documento HTML, cargará el directorio de sus imágenes y le realizará un procesamiento para determinar

por cada imagen la existencia o no de desnudez (presencia de piel humana). Seguidamente se elabora un mensaje que se colocará en un fichero de respuestas. Este mensaje será posteriormente utilizado por el Módulo Decisor, para decidir a qué categoría finalmente pertenece el documento HTML.

Módulo-Clasificador de Objetos

Este módulo tiene como función recibir a través de la consulta de un fichero de solicitudes, un mensaje proveniente del MOD-Controlador el cual indica la clasificación de las imágenes de un documento HTML. Para clasificarlo accederá al directorio de localización del documento HTML, cargará el directorio de sus imágenes y le realizará un procesamiento para determinar por cada imagen el número de símbolos encontrados de cada categoría (Ciencias, Computadoras, Deporte, Juegos, Pornografía, Violencia, Sustancias dañinas, Salud). Seguidamente se elabora un mensaje que se colocará en un fichero de respuestas. Este mensaje será posteriormente utilizado por el Módulo Decisor, para decidir a qué categoría finalmente pertenece el documento HTML.

El Módulo-Reconocimiento Óptico de Caracteres

Tiene como función recibir a través de la consulta de un fichero de solicitudes, un mensaje proveniente del Módulo-Controlador el cual indica la clasificación de las imágenes de una URL. Para dicha clasificación accederá al directorio de localización de la URL, cargará el directorio de sus imágenes y le realizará un procesamiento para extraer por cada imagen todo el texto asociado y elaborar un único fichero de texto a partir del cual, y utilizando los mismos algoritmos del clasificador de textos determinará las categorías de contenido (Ciencias, Computadoras, Deporte, Juegos, Pornografía, Violencia…) asociadas a esta URL, elaborando un mensaje que colocará en un fichero de respuestas. Este mensaje será posteriormente utilizado por el Módulo Decisor, para decidir a qué categoría finalmente pertenece la URL.

El Módulo-Clasificador de Enlaces

Tiene como función recibir a través de la consulta de un fichero de solicitudes, un mensaje proveniente del Módulo-Controlador el cual indica la clasificación de los enlaces relacionados a una URL. Para dicha clasificación accederá al directorio de localización de la URL, cargará el fichero donde están depositados todos los enlaces correspondientes, le realizará un procesamiento para determinar las categorías de contenido presentes (Ciencias, Computadoras, Deporte, Juegos, Pornografía, Violencia…) asociadas a esta URL, elaborando un mensaje que colocará en un fichero de respuestas. Este mensaje será posteriormente utilizado por el Módulo Decisor, para decidir a qué categoría finalmente pertenece la URL.

El Módulo-Controlador

Es el que controla y sincroniza todo el funcionamiento del motor. Posee un fichero de configuración central en el cual se puede:

1. Activar o desactivar Módulos.
2. Configuración general de los módulos (forma de comunicación con los restantes módulos, localización de ficheros de configuración específico, localización de directorio DEPÓSITO)
3. Posee una interfaz Web para la configuración de los módulos activos y para el monitoreo del funcionamiento de los mismos.

El Módulo-Decisor

Tiene como función recibir por parte del Módulo-Controlador, toda la información proveniente de los módulos clasificadores y devolverle la categoría más probable a la que pertenece la URL.

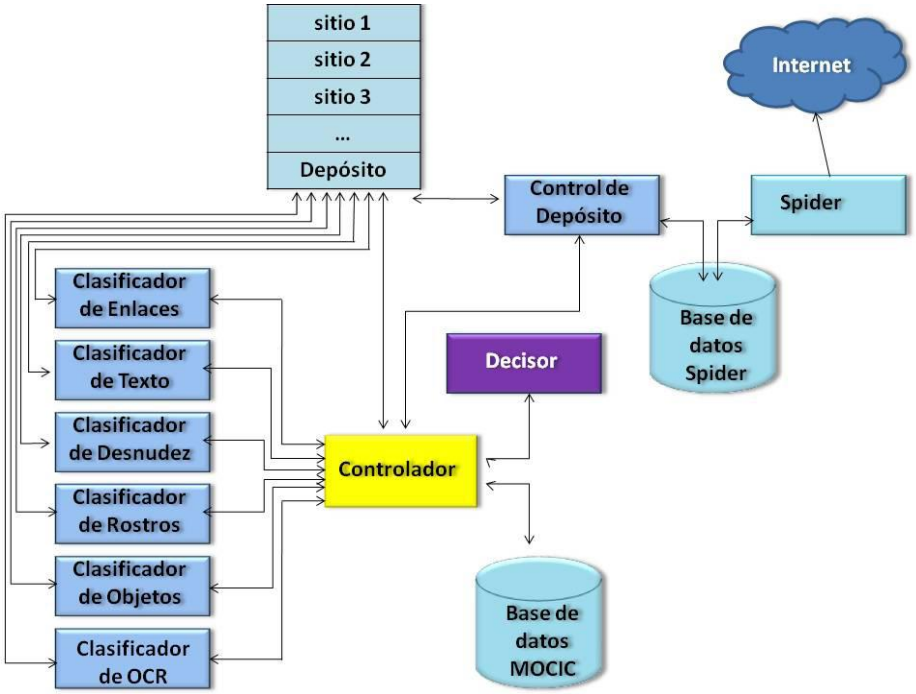

FIGURA 8 ARQUITECTURA DEL MOCIC

El trabajo se desarrolla utilizando el Módulo-Clasificador de Texto el cual tiene como función

recibir a través de la consulta de un fichero de solicitudes, un mensaje proveniente del Módulo-Controlador el cual indica la clasificación del texto de una URL. Para dicha clasificación accederá al directorio de localización de la URL, cargará el fichero .html correspondiente, le realizará un pre-procesamiento para identificar el idioma al que pertenece y seguidamente determinará las categorías de contenido previamente definidas (Ciencias, Computadoras, Deporte, Juegos, Pornografía, Violencia…) asociadas a esta URL, elaborando un mensaje que colocará en un fichero de respuestas. Este mensaje será posteriormente utilizado por el Módulo Decisor, para decidir a qué categoría finalmente pertenece la URL.

Para este motor se desea implementar la función de ponderación de términos ACC descrita anteriormente en el primer Capítulo I. Como la misma sólo realiza un análisis del código HTML es de vital importancia que se extienda a un análisis también de las CSS, nueva tecnología que rompe paradigmas para el diseño de páginas Web. Se hace preciso definir qué términos o frases están enfatizados en el texto o sencillamente qué propiedades CSS determinarán el enfatizado, debido a que será el único criterio que se tendrá en cuenta para esta propuesta. Así se obtienen las partes en el texto en las que se encierra la información en la que el autor desea que el lector haga más énfasis y por consiguiente se puede conocer a qué categoría pertenecerá a la hora de la clasificación automática.

2.2 Propiedades y Valores que Permiten Enfatizar los Elementos

Existen diferentes formas para resaltar determinadas frases en un documento. Así se debe de tomar en cuenta cada una de estas ya que encierran una significación visual, por lo que el lector se enfoca en estas partes enfatizadas. También permiten que el texto sea mucho más accesible visualmente, facilitando así una lectura más rápida y efectiva. En la actualidad los selectores que más se utilizan para enfatizar determinados elementos utilizando HTML pueden ser los mostrados en la siguiente tabla:

Teniendo en cuenta que con el uso de las CSS se puede enfatizar un determinado elemento se realizó la Tabla 3 con algunas de las propiedades y valores que se toman para enfatizar elementos en un documento dado.

Con la utilización de las propiedades y valores expuestos en la Tabla 3, se podrá dar un carácter enfatizado a un término o elemento determinado en el texto de un documento haciendo uso de las hojas de estilo en cascada (CSS). Por lo tanto parseando el CSS se puede

obtener a su vez los selectores a los cuales se les están aplicando estas propiedades y valores. Una vez determinado el selector en el código HTML se obtendrá el término que se enfatizó.

TABLA 2 SELECTORES QUE SE UTILIZAN PARA ENFATIZAR ELEMENTOS CON EL LENGUAJE HTML

Apertura	Acción	Atributos	Cierre
<H1...H6>	Tamaño de letras del 1 al 6.	HTML 3.0: left, center, right	</H1.../H6>
	Formato enfatizado más fuerte que <EM>.	Ninguno	
	Pone el texto en negrita	ninguno	
	Formato enfatizado en itálica.	ninguno	
<u>	Para subrayar el texto indicado	ninguno	</u>
	Itálica (Cursiva).	Ninguno	

TABLA 3 ALGUNAS PROPIEDADES Y VALORES QUE SE EMPLEAN CON CSS PARA ENFATIZAR

Descripción	Propiedad	Valor
Tamaño	'font-size'	12px…npx
Ancho de fuente	'font-weight'	Bold
Decoración	'text-decoration'	Underline
Mayúsculas	'text-transform'	Uppercase

2.3 Librerías a Utilizar

Acerca de librerías se pude destacar que en el ámbito de las ciencias de la computación se les denomina a una serie de programas que son utilizados para desarrollar software. Las mismas contienen código y datos que brindan servicios a programas que funcionan de forma independiente. Es decir, como que son un pedacito más de cada programa para funciones

particulares que estos necesiten en momentos determinados. En ocasiones se tiene el caso de programas ejecutables que pueden ser simultáneamente programas independientes o bibliotecas, término por el cual se le conoce también a librería. Es preciso señalar que la mayoría de las bibliotecas no son ejecutables pero que tanto ejecutables como librerías llevan a cabo referencias entre ellas mediante un proceso que se conoce con el nombre de enlace. El lenguaje de programación Python no está exento de poseer librerías y cuenta con un número de ellas para determinadas funciones.

2.3.1 Librería LXML

La propuesta a solución de este trabajo parte del uso de la librería LXML, que no es más que un nuevo enlace para Libxml2 y Libxslt totalmente independiente de los enlaces existentes Python. LXML viene siendo como urllib2 (librería para recuperar el contenido de una página web) y BeautifulSoup (parser HTML/XML para analizar texto, técnica que se conoce como screen-scraping) en una sola librería excepto que LXML permite utilizar XSLT y Selectores CSS para realizar el scraping. Esta librería LXML brinda facilidades a la hora del análisis de XML y HTML existiendo una potente API nativa Python para ello, siendo de esta forma la más rica específicamente para el trabajo con XML y HTML en este lenguaje al combinar la velocidad e integridad de libxml2 y libxslt.

2.3.2 Librería CSSUTILS

CSSUTILS se desarrolla en el estándar de Python. La misma es utilizada para analizar y construir hojas de estilo CSS. Debe ser capaz de leer y escribir el mayor número posible de estilos CSS, intenta utilizar las características de CSS 2.1 y CSS 3. Esta librería se publica bajo la licencia LGPL 3 o posterior a esta versión.

En nuestra tesis es de vital importancia la utilización de la librería, ya que la misma nos permite analizar las hojas de estilo en cascada (CSS) y de esta forma determinar las propiedades que den un efecto de enfatizado en el texto.

2.4 Esquema de Solución

La figura 9 muestra el esquema que expone de forma clara los pasos a seguir para lograr la obtención de los términos que denoten cierta relevancia en un texto, teniendo en cuenta las CSS asociadas a las páginas.

Se parte desde la base de los documentos: el código HTML, haciendo uso de la librería LXML para a través de un parseo obtener las etiquetas Style y Link, las cuales contienen la URL del archivo .css a analizar. Específicamente con las sentencias del lenguaje Python `link = xml.cssselect.CSSSelector('link')` y `style = lxml.cssselect.CSSSelector('style')` se obtienen estas etiquetas. Se analiza posteriormente para el caso de la etiqueta Style la dirección que contiene en el @import y para la etiqueta link que el atributo *type* tenga el valor "text/css", porque para el caso de los archivos .css siempre ha de ser el mismo. Se obtendría la dirección que tendría como valor el atributo href, logrando así localizar cada una de las direcciones de los archivos .css que dan estilo al documento.

Luego de la obtención del archivo .css o los archivos en caso de ser más de uno, con la librería CSSUTILS, se procede al posterior parseo del mismo o los mismos para capturar los selectores con su propiedad y valor correspondientes, donde estos últimos son los que enfatizarían el término, rasgo, frase, etc., en el texto del documento.

Conociendo dicho selector con su propiedad y valor quedaría obtenerlo en el código HTML y de esta forma se encontrará el término enfatizado dado por el estilo que da el archivo .css.

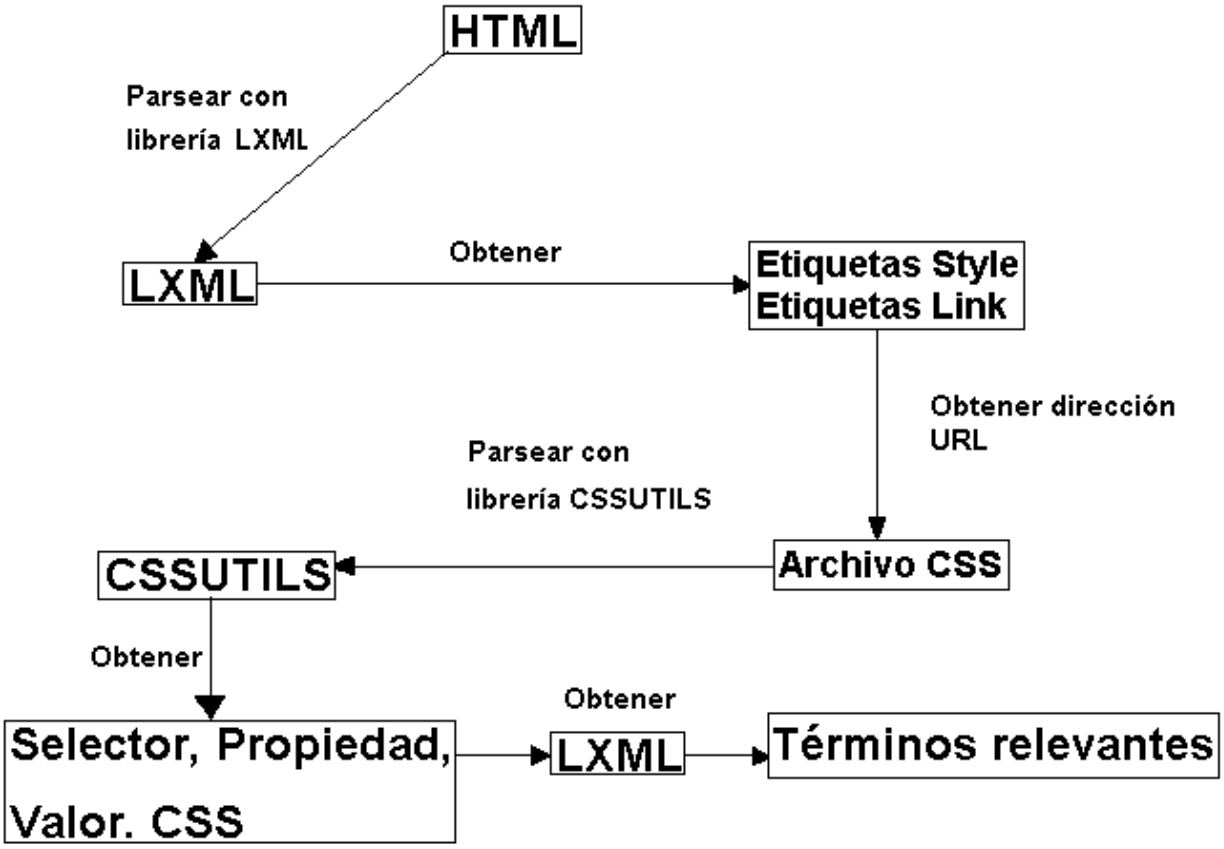

FIGURA 9 ESQUEMA PARA LA PROPUESTA DE SOLUCIÓN

2.5 Estudio de Caso

A continuación se muestra las imágenes de los resultados obtenidos de una aplicación de apoyo a baja escala que se realizó con el fin de comprobar el correcto funcionamiento del método propuesto para la detección de términos relevantes a través de las Hojas de Estilo en Cascada.

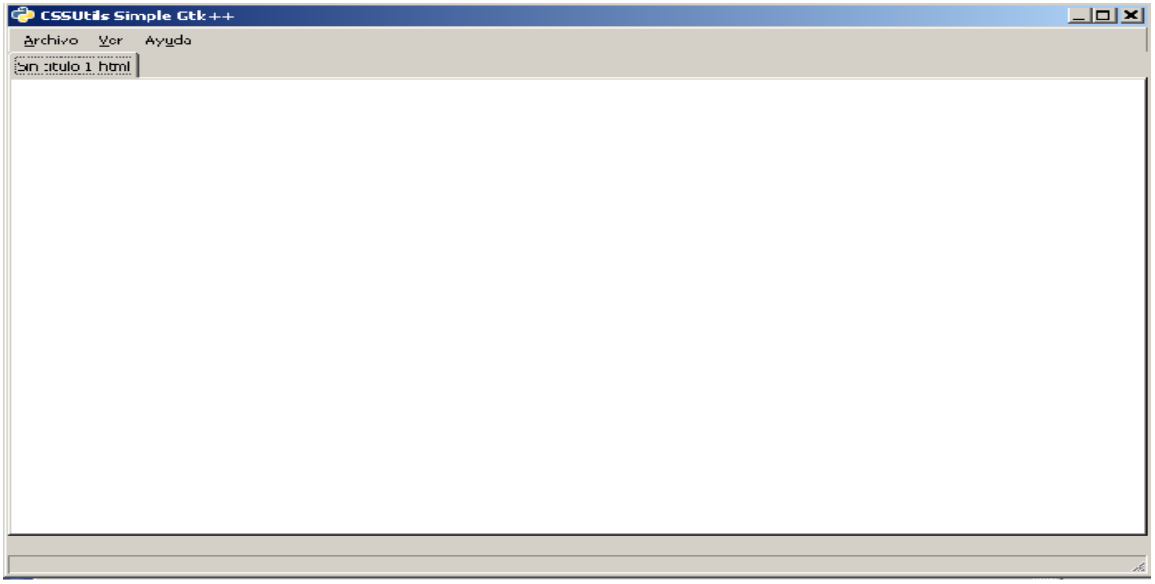

FIGURA 10 INTERFAZ DE APLICACIÓN DE APOYO

En la figura 10 se muestra la interfaz de la aplicación de apoyo sin ningún archivo HTML cargado para ser parseado. En la parte superior se muestra el menú que contiene las diferentes funcionalidades de la misma.

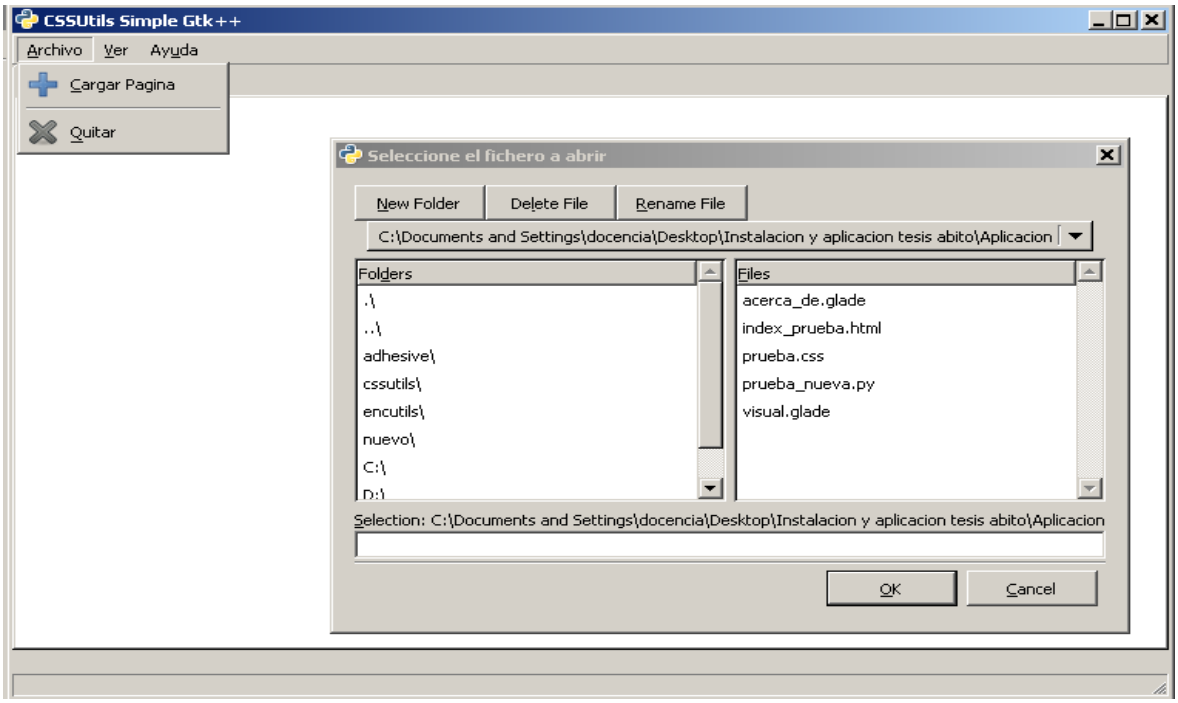

FIGURA 11 INTERFAZ DE CARAR ARCHIVO

En la figura 11 se muestra la ventana que permite seleccionar el fichero HTML que se desea

cargar para su posterior análisis.

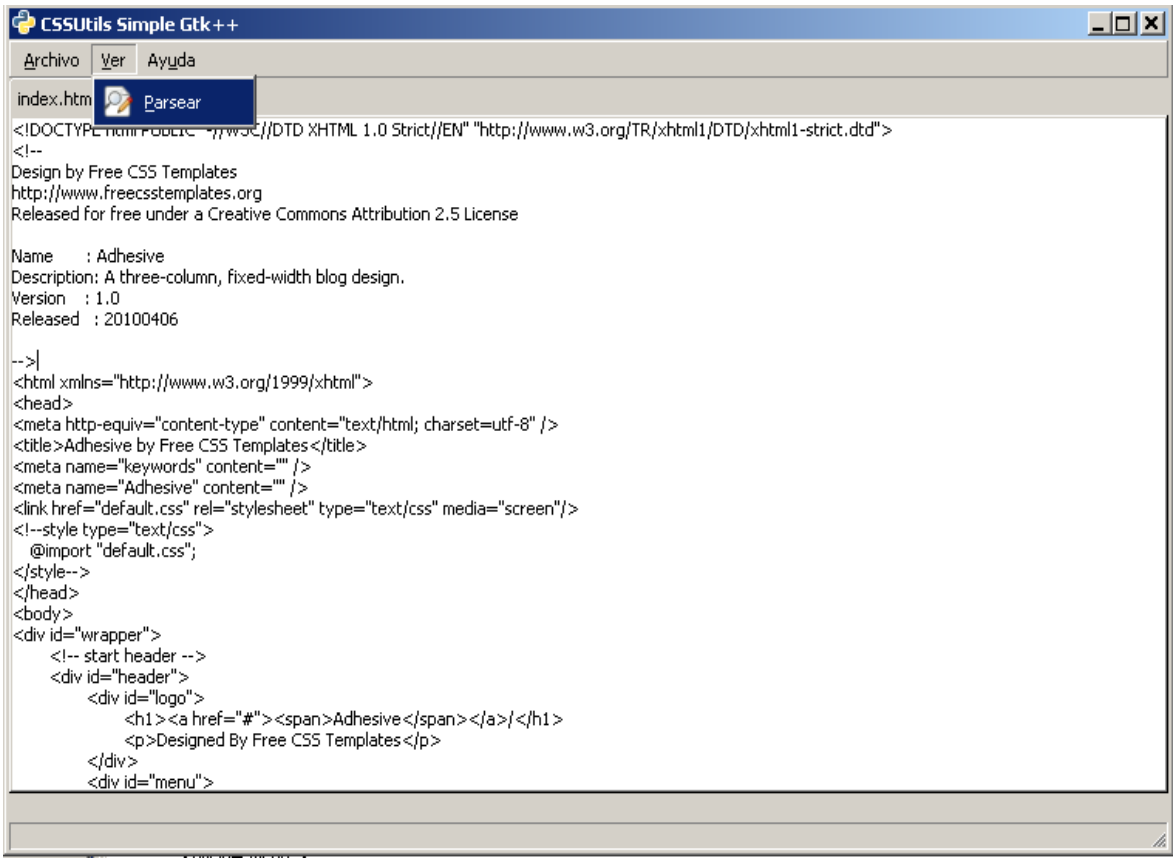

FIGURA 12 PARSEO CON LA LIBRERÍA LXML Y CSSUTILS PARA EL CSS

En la figura 12 se muestra el código HTML y la opción activa para parsear el HTML con la librería LXML que permite obtener las etiquetas link y Style y de esta forma luego la dirección CSS para acceder a ella.

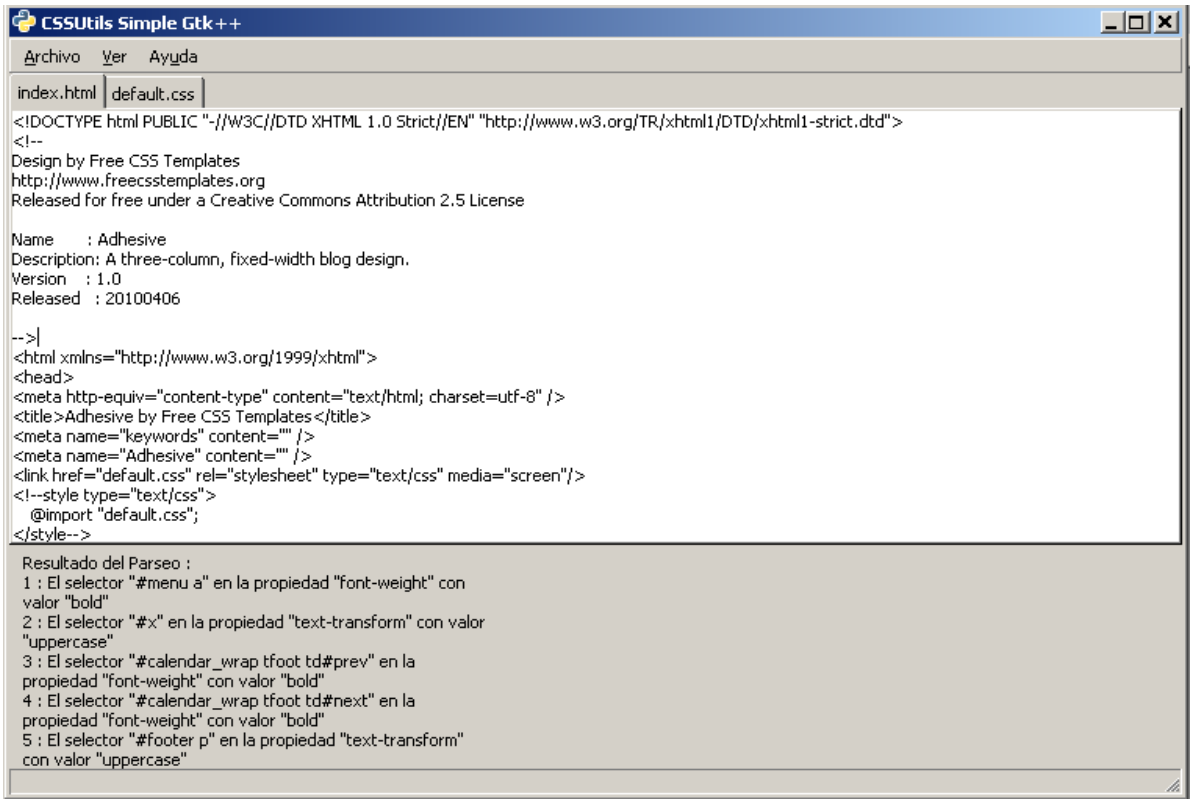

FIGURA 13 RESULTADO DEL PARSEO

En la figura 13 se muestra el resultado final para el documento. Aparecen las propiedades CSS y los valores que dan enfatizado a partes determinadas del texto analizado, junto con los selectores a los que se aplican las reglas en el .html.

2.6 Conclusiones

Durante el desarrollo de este capítulo se describió brevemente el funcionamiento de MOCIC para quien será aplicada la propuesta a solución, en específico para el Módulo-Clasificador de texto. Dicha propuesta se explicó destacando los aspectos o conceptos fundamentales que se relacionan con la misma. Se formuló un ejemplo que se toma a modo de prueba para observar paso a paso el método propuesto y hacerlo más entendible.

CONCLUSIONES GENERALES

- Se determinaron los estilos más empleados por los diseñadores Web para resaltar diferentes términos o frases en el texto de un documento a través de los archivos CSS.
- El criterio “enfatizado” es el único de los cuatro criterios combinados en ACC que resalta el texto de un documento.
- Se propuso un método como solución al problema para obtener términos relevantes en las páginas HTML a partir del análisis de las CSS asociadas..

RECOMENDACIONES

Realizar un estudio del "color del texto" como indicador de enfatizado debido a que existe gran variedad de diseños web en ese sentido por lo que no se hace posible tomar un estándar para ello y la tarea se torna compleja.

Integrar el método propuesto a la función ACC y por consiguiente al Motor de Clasificación Inteligente por Contenido (MOCIC).

REFERENCIAS BIBLIOGRÁFICAS

Álvaro González Fernández. *Extracción y recuperación de la información II*. 2007.

Andrés Fernández. *Estructura de páginas web: Marcado semántico de contenidos*. 2010 Available from world wide web: <http://www.lawebera.es/manual-diseno-web/estructura-paginas-web-marcado-semantico-contenidos.php>.

Baeza-Yates y Ribeiro-Neto. *Modelo de Espacio Vectorial*.

Carlos G. Figuerola, José L. Alonso Berrocal, Angel F. Zazo Rodríguez, Emilio Rodríguez. Algunas Técnicas de Clasificación Automática de Documentos. 2005.

Chuck Musciano y Bill Kennedy. *HTML la Guía Completa*. [México DF], 1999.

J. Campos. *Esquemas algorítmicos - Algoritmos probabilistas*.

Javier Eguíluz Pérez. Introducción a CSS. 2009.

Luís Codina. *Teoría de recuperación de información: modelos fundamentales y aplicaciones a la gestión documental*. 2005.

Morkes, J. y Nielsen. *Concise, scannable, and objective: How to write for the web.* 1997.

Noel L. Núñez Camallea & Ronald Coutin Abalo. *Diccionario de Informática*. 2005.

René Venegas. *Clasificación de textos académicos en función de su contenido léxico-semántico*. [Revista Signos], 2007a.

René Venegas. Clasificación de textos académicos en función de su contenido léxico-semántico. *Revista Signos* 2007b.

Víctor Fresno Fernández. *Representación Autocontenida de Documentos HTML: una*

propuesta basada en Combinaciones Heurísticas de Criterios. 2005.

www.psicobyte.com. Estructura de una hoja de estilo CSS. [cited 19 Mayo 2010]. Available from world wide web: <http://www.psicobyte.com/html/css/css2.html>.

Yuan Jiang and Zhi-hua Zhou. *Editing Training Data for kNN Classifiers with Neural Network Ensemble*. [National Laboratory for Novel Software Technology, Nanjing University, China], 2004.

BIBLIOGRAFÍA

Alberto P. García-Plaza, Víctor Fresno, Raquel Martínez. *Una Representación Basada en Lógica Borrosa para el Clustering de páginas web con Mapas Auto-Organizativos*. [Universidad Nacional de Educación a Distancia C/Juan del Rosal, España], 2009.

Alberto Téllez Valero. Extracción de Información con Algoritmos de Clasificación. 2005.

Álvaro González Fernández. *Extracción y recuperación de la información II*. 2007.

Anarta Ghosh and Michael Biehl and Barbara Hammer. *Dynamical analysis of LVQ type learning rules*. 2005.

Anca Doloc-mihu and Vijay Raghavan and Peter Bollmann-sdorra. *Color Retrieval in Vector Space Model*. 2003.

Andrés Fernández. *Estructura de páginas web: Marcado semántico de contenidos*. 2010 Available from world wide web: <http://www.lawebera.es/manual-diseno-web/estructura-paginas-web-marcado-semantico-contenidos.php>.

Arantza Casillas Rubio, Raquel Martínez Unanue, Víctor Fresno Fernández, Soto Montalvo Herranz. *Evaluación del clustering de páginas web mediante funciones de peso y combinación heurística de criterios*. 2005.

Baeza-Yates y Ribeiro-Neto. *Modelo de Espacio Vectorial*.

Barbara Hammer Marc and Marc Strickert and Thomas Villmann. *Relevance LVQ versus SVM*. [Department of Mathematics/Computer Science, University of Osnabr¨uck, D-49069 Osnabr¨uck, Germany & Clinic for Psychotherapy and Psychosomatic Medicine, University of Leipzig], 2004.

Carlos G. Figuerola, José L. Alonso Berrocal, Angel F. Zazo Rodríguez, Emilio Rodríguez. Algunas Técnicas de Clasificación Automática de Documentos. 2005.

César Cuba Rodríguez, Julio César Morejón Ríos. *Almacenamiento y Gestión de Contenido en el Motor de Clasificación Inteligente de Contenidos (MOCIC)*. [Universidad de las Ciencias Informáticas], 2009.

Chenyi Xia Hongjun and Hongjun Lu and Beng Chin and Ooi Jing Hu. *GORDER: An Efficient Method for KNN Join Processing*. 2004.

Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze. *An Introduction to Information Retrieval*. [Cambridge University PressCambridge, England], 2008.

Cristina Díaz Moreno. Clasificación no supervisada. Recuperación y organización de la información. 2007. Available from world wide web: <http://clustering.50webs.com/>.

Cui Yu and Beng Chin Ooi and Kian-lee Tan and H. V. Jagadish. *Indexing the Distance: An Efficient Method to KNN Processing*. [Department of Computer Science National University of Singapore, Department of Electrical Engineering & Computer Science University of Michigan], 2001.

Dr. Roberto Hernández Sampieri, Dr. Carlos Fernández Collado, Dra. Pilar Baptista Lucio. *Metodología de la investigación*. 2006.

Edelsys Hernández Meléndrez. Cómo escribir una tesis. 2006.

Ellen Voorhees National and Ellen M. Voorhees. *Overview of TREC 2004*. 2001.

Ellen Voorhees National and Ellen M. Voorhees. *Question Answering in TREC*. [National Institute of Standards and Technology Gaithersburg], 2005.

HTML.net. Aprende HTML. 2005. Available from world wide web: <http://es.html.net/tutorials/html/lesson2.asp>.

Indira Tamarit Muñoz, Ana Miranda Bermúdez. Propuesta del Módulo Decidor del Motor de Clasificación Inteligente de Contenidos (MOCIC). 2009.

J. Campos. *Esquemas algorítmicos - Algoritmos probabilistas*.

Javier Eguíluz Pérez. Introducción a CSS. 2009.

Javier Eguíluz Pérez. Introducción a XHTML. 2009b. Available from world wide web: <http://librosweb.es/xhtml/capitulo1/breve_historia_de_html.html>.

John M. Conroy and Daniel M. Dunlavy and Dianne P. O'leary. *From trec to duc to trec again*. [University of Maryland].

Luís Codina. *Teoría de recuperación de información: modelos fundamentales y aplicaciones a la gestión documental*. 2005.

María Victoria Fornaguera Rodríguez. *Propuesta de un sistema de reconocimiento de rostros para MOCIC*. [Universidad de las Ciencias Informáticas], 2009.

Michael Biehl and Anarta Ghosh and Barbara Hammer and Yoshua Bengio. *Dynamics and generalization ability of LVQ algorithms*. 2006.

Noel L. Núñez Camallea & Ronald Coutin Abalo. *Diccionario de Informática*. 2005.

Pascal Soucy. *Beyond TFIDF weighting for text categorization in the vector space model*. [Université Laval Québec, Canada], 2005.

René Venegas. *Clasificación de textos académicos en función de su contenido léxico-semántico*. [Revista Signos], 2007.

Ricardo Cárdenas Medina. *Inmersión en Python*. 2005.

Robert A. Day. Cómo escribir y publicar trabajos científicos. 2005.

Rolando Alfredo Hernández León & Sayda Coello González. *EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTIFICA*. [Universidad de las Ciencias Informáticas], 2002.

Ronan Cummins. *Determining general term weighting schemes for the vector space model of information retrieval using genetic programming*. [Dept. of Information Technology, National University of Ireland, Galway, Ireland.], 2004.

Rup Nielsen Informatics and Rup Nielsen and Informatics and Finn A and Finn A and Finn A. *Bornholm Text analysis*. [Informatics and Mathematical Modelling Technical University of Denmark], 2002.

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. [Printed in the United States of America.], 2009.

Victor Diego Fresno Fernández. Representación Autocontenida de Documentos HTML: una propuesta basada en Combinaciones Heurísticas de Criterios. 2005.

Víctor Fresno Fernández. Clasificación Naïve Bayes de páginas web para representaciones enriquecidas con información de marcado. 2005.

Víctor Fresno Fernández. *Representación Autocontenida de Documentos HTML: una propuesta basada en Combinaciones Heurísticas de Criterios*. 2005.

www.psicobyte.com. Estructura de una hoja de estilo CSS. Available from world wide web: <http://www.psicobyte.com/html/css/css2.html>.

Yuan Jiang and Zhi-hua Zhou. *Editing Training Data for kNN Classifiers with Neural Network Ensemble*. [National Laboratory for Novel Software Technology, Nanjing University, China], 2004.

Yusef Hassan Montero y Francisco Jesús Martín Fernández. *DHTML: Capas*. 2003.

GLOSARIO DE TÉRMINOS

ACC: Combinación Analítica de Criterios.

CSS: Cascading Style Sheets: Hojas de estilo en cascada. Es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML

FILPACON: Filtrado de Paquetes por Contenido: Es un sistema que pretende ser flexible y fiable para regular los contenidos nocivos de Internet.

Hardware: Conjunto de componentes físicos internos y externos que integran una computadora.

Hipertexto: Tecnología que organiza una base de información en bloques distintos de contenidos, conectados a través de una serie de enlaces cuya activación o elección provoca la RI.

Hipervínculo (hyperlink): Una conexión entre un elemento de un documento de hipertexto como una palabra, frase, símbolo o imagen y un elemento diferente del documento, otro documento de hipertexto, un archivo o un guión.

HTML: HyperText Mark Language: Lenguaje de Marcas de Hipertexto. Es el lenguaje de marcado predominante para la construcción de páginas Web.

HTTP: El protocolo de transferencia de hipertexto o *HyperText Transfer Protocol* es el protocolo usado en cada transacción de la Web. Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor.

IA Inteligencia artificial.

Internet: Red internacional que conecta miles de redes más pequeñas. “Internet” con mayúscula se refiere a la red que actualmente se usa, mientras que “internet” con minúscula es el concepto de interconectar varias redes.

K-NN: k Nearest Neighbour: Proviene del idioma inglés y es traducido como los k vecinos más cercanos.

Libxml2: Es una librería que facilita el manejo de los XMLs. La librería permite el acceso tanto para lectura como para escritura.

Libxslt: Es una librería que contiene las librerías XSLT, útiles para añadir a las librerías libxml2 soporte de ficheros XSLT.

MB: Motores de Búsqueda: Es un sistema informático que busca archivos almacenados en servidores web gracias a su «*spider*».

MOCIC: Motor de Categorización Inteligente de Contenido.

Página: Toda entidad en la Web que tiene asociada una *URL*. En este documento usamos una definición un poco más restrictiva que no considera como páginas a imágenes, video, música y otros archivos multimedia o comprimidos.

Protocolo TCP-IP: Es un conjunto de protocolos de red en la que se basa Internet y que permiten la transmisión de datos entre redes de computadoras. Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP).

RI: Recuperación de información.

Software: se refiere al equipamiento lógico o soporte lógico de una computadora digital, y comprende el conjunto de los componentes lógicos necesarios para hacer posible la realización de tareas específicas; en contraposición a los componentes físicos del sistema, llamados Hardware.

UCI: Universidad de las Ciencias Informáticas.

URL: Estándar para referirse a una dirección en la Web.

Web: Red: La traducción literal de esta palabra inglesa es tela de araña, pero en términos informáticos significa mucho más que eso.

WWW: World Wide Web: Telaraña o malla mundial. Sistema de información distribuido con mecanismos de hipertexto. Es el universo de servidores HTTP, que permiten mezclar texto, gráficos y archivos de sonido.