



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

"Clasificación de noticias digitales a través de
redes neuronales ART'2"

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
INGENIERA EN COMPUTACIÓN

PRESENTA:
MARÍA DEL ROCÍO ORTIZ ALBARRÁN



DIRECTOR DE TESIS:
M.C. EDUARDO ESPINOSA AVILA

2010

...

«Recibid la enseñanza, y no plata; Y ciencia antes que el oro escogido... El hombre sabrá alzar los ojos al cielo y agradecer con humildad la ayuda de su hermano y de su padre...»

♡x**Jah**

Al viento, al fuego y a la energía de n dimensiones que me han llevado por un camino lleno de desniveles y horizontes desconocidos, por que sé que estás ahí, te siento día con día. «No seré confundida después de tanto confiar en ti».

*A mi fuerza, el más dulce aliento, **mi mami Estela**, la mano firme llena de cuidados, trabajo y esfuerzo, mi inspiración a cada rayo de sol para construir los cimientos de los edificios más impresionantes que rondan mi mente.*

*A mi corazón, el mayor templo de la verdad y del sentir, **mi papi Luis**, por que cada segundo a su lado forma parte del pilar de mi vida, y estar a su lado es mi más valioso tesoro.*

*A mi esperanza, la magia de saber que nunca estoy sola, que siempre va conmigo un ángel que guía mis pasos, mis pensamientos, mis palabras, **María Estela** gracias a ti no conozco la soledad.*

*A mi entusiasmo por vivir, por disfrutar, **José**, la voz de impulso que me alentó a dar un paso siempre firme y a levantar el corazón cada vez que el alma cae.*

*A mi alegría, el más grande cómplice, **Luis**, el ejemplo de orden y disciplina, por crear conmigo un árbol irracional de lágrimas y risas que vivirá en mi recuerdo y en cada nuevo mirar, Siempre estaré contigo.*

*A mi amor, la rareza de los lados del corazón, **Lalo**, el compañero que me muestra como se cultiva el alma y como el infinito no basta cuando se cuantifica el sentir de una caricia.*

*A mi nobleza, el toque de ternura, **mi tía Laura**, la relatora de mundos fantásticos que se recorren con la vista y se crean solo con tinta y papel... y los recuerdos; por mostrarme que la vida es el retumbar de un fuerte latido, la lucha por un deseo.*

*A mi valentía, muestra de lucha incansable, **Mary Carmen**, el adalid que me regala una mirada, una palabra, una vida, su confianza, por ser una de las 4 estrellas que iluminaron mis noches más largas.*

*A mi juventud, **Paco, Alonso y Emi**, las chispas de locura que me muestran con 3 sonrisas que el universo nunca termina de crear obras de arte en el alma de aquellos que valoran el cariño, el trabajo y la lucha.*

*A mis sentimientos, las estrellas fugaces, que en cada resplandor dejan una estela impresionante de polvos cósmicos en la vida de quien les rodea, mis maestros y amigos: **Fabian Orozco, Carmen Chico, Haydée Álvarez, Tere Miranda, Sonia Leños, Miguel Reyes**, acompañantes fieles de mis pensamientos, de mi formación académica y humana, fundadores de las ideas en mi mente y de los valores que rigen mi corazón.*

*A mi sonrisa, esa que se forma con los recuerdos, como una galaxia interminable que se admira al elevar los ojos al cielo sólo en las noches que el buen amigo acompaña el corazón... Gracias **Lalo, Adrian, Alfredo, Anita, Eloisa, Jassiel, Vale, Aide, Karina, Gabriel**...*



Gracias UNAM!!! Por tu riqueza infinita y por llenarme de tus ideales de sabiduría, historia y nobleza, por saber llenar mi alma con sangre azul y cubrir mi rostro con piel dorada. Yo sé que la Universidad Nacional Autónoma de México es parte de mi.

Gracias a mi director de tesis!

M.C. Eduardo Espinosa Avila, por su dedicación, sus consejos, su tiempo y por mostrarme la importancia de la honestidad, la responsabilidad y la puntualidad en la vida profesional. Por enseñarme que la cualidad más valiosa del conocimiento es que se puede compartir con los demás y por revelarme la infinidad del saber, de las teorías del hombre y del universo.

Gracias a mis sinodales!

*Ing. Stalin Muñoz
Ing. Román Osorio
M.I. Juan Manuel Gómez
Dra. Karina Mendoza*

Por que no solo fueron una guía en la finalización de este trabajo sino un ejemplo de vocación en la docencia, gracias por sus enseñanzas pero sobre todo por mostrarme los frutos del trabajo bien realizado.

Gracias Facultad de Ingeniería!!!

El hogar de mis sueños, de mi carácter, de mis valores... El mundo donde aprendí que el saber solamente me hará crecer cuando sepa cómo utilizarlo para mejorar la vida del hombre.



Índice general

1. Introducción	9
1.1. Planteamiento inicial	9
1.2. Problemática	10
1.2.1. Situación Actual	10
1.2.2. Herramientas existentes	11
1.3. Solución	11
1.3.1. Objetivo	11
1.3.2. Propuesta	11
1.3.3. Justificación	12
1.3.4. Metodología	15
1.3.4.1. Desarrollo del proyecto	15
1.3.4.2. Desarrollo de la ART	16
1.3.4.3. Desarrollo del Sistema	16
1.4. Organización de la tesis	16
2. Marco Teórico	17
2.1. Herramientas para el diseño e implementación	17
2.1.1. PHP	17
2.1.2. CSS	17
2.1.3. Ajax	18
2.1.4. Apache	19
2.1.5. MySQL	19
2.2. Conceptos de Ingeniería de <i>Software</i>	20
2.2.1. Programación Estructurada	20
2.2.2. Ciclo de vida de un sistema de informático	20
2.2.2.1. Modelo en cascada	21
2.2.2.2. Modelo Transformacional	22
2.2.2.3. Modelo Espiral.	23
2.2.2.4. Modelo evolutivo ó incremental	23
2.3. Algoritmos	25
2.3.1. Complejidad	26
2.3.2. Correctez	26
2.4. Redes neuronales artificiales	26
2.4.1. Elementos	27
2.4.2. Funcionamiento	28
2.4.3. Aprendizaje	30
2.4.3.1. Aprendizaje no supervisado	30

2.4.4.	Tipos de Red	30
2.4.5.	ART	34
2.4.6.	ART2	34
3.	Red Neuronal	37
3.1.	Arquitectura	37
3.2.	Conexiones	38
3.3.	Establecimiento de constantes	38
3.4.	Entradas de la red	40
3.5.	Activación de neuronas	40
3.6.	Proceso de entrenamiento	41
4.	Desarrollo del sistema	43
4.1.	Análisis	44
4.1.1.	Requerimientos del sistema	44
4.1.2.	Requerimientos de interfaces	49
4.2.	Diseño	50
4.2.1.	Diseño Gráfico	51
4.2.2.	Diseño modular	51
4.2.2.1.	Módulo PP (M-PP)	52
4.2.2.2.	Módulo de Red Neuronal(M-RN)	52
4.2.2.3.	Módulo de Proceso(M-Proceso)	53
4.2.2.4.	Módulo de Administración (M-Administración).	54
4.2.2.5.	Módulo de Diseño de la Base de Datos(M-BD).	54
4.2.2.6.	Módulo de Presentación (M-Presentación).	55
4.2.3.	Algoritmos	56
4.2.3.1.	Complejidad	57
4.2.4.	Diagramas de flujo(Ordinogramas)	58
4.2.5.	Diagrama de bloques	60
4.3.	Codificación del sistema	60
4.4.	Implantación	66
4.5.	Pruebas	67
4.5.1.	Plantillas de prueba(pruebas de sistema)	68
4.5.2.	Pruebas de Integración	75
4.5.3.	Pruebas de Rendimiento	76
4.6.	Mantenimiento	76
5.	Seguridad	77
5.1.	Seguridad del sistema	77
5.1.1.	Niveles de usuario	78
5.2.	Seguridad de la base de datos	79
5.2.1.	Niveles de usuario	80
6.	Resultados	83
7.	Conclusiones	87
	Referencias	89
	Glosario	92

Apéndices	97
Apéndice A. Documentos entregables	97
A-1. Especificación de requisitos de <i>Software</i>	97
A-2. Diagrama de Flujo de Datos	100
A-3. Base de Datos	102
A-4. Interfaz base	105
A-5. Manual de Usuario	106
Apéndice B. Ejemplo de patrones por diario	136
Apéndice C. Código de la red ART2	138

Capítulo 1

Introducción

Una de las metas de la Ingeniería en Computación es la automatización de procesos mediante el análisis de tareas cotidianas y un buen diseño de *software*, esto con el fin de lograr que los resultados que brinda un sistema superen los resultados obtenidos por los seres humanos. Las visiones futuristas de escritores como Isaac Asimov se ven realizadas conforme avanza el tiempo y la tecnología, es por esto que el ser humano en la búsqueda de facilitar sus tareas se adentra en el mundo de la automatización, ésta busca que el ser humano deje de realizar las tareas triviales, peligrosas o desconocidas y que a través de la computación la humanidad logre realizar tareas cada vez más complejas y resolver problemas en los cuales, actualmente, el hombre no ha encontrado una solución.

1.1. Planteamiento inicial

Ante la necesidad de la manipulación masiva de información y ante la disponibilidad de esta en internet surge la necesidad de automatizar el proceso de análisis, clasificación, presentación y búsqueda de información de interés. Una de las nuevas fuentes más grandes de información en la red es el surgimiento de los diarios en versión electrónica, en estas versiones se presentan entre 30 y 1500 notas periodísticas digitalizadas que se realizan a través de administradores de contenido muy específicos para esta tarea, estos que poseen la versatilidad de agregar programas que presenten la información de la manera en la que la institución interesada lo desea o lo necesita.

Cuando se quiere realizar un análisis a través de estos recursos informativos sobre un tema determinado uno puede establecer una búsqueda en los diarios electrónicos de interés, a través de motores específicos como *google* o utilizar las opciones de búsqueda que ofrecen las páginas *web* de algunos periódicos, sin embargo, si se trata de buscar NO un solo tema sino cientos y NO en la página de un solo diario sino en decenas de diarios, la tarea se vuelve exhaustiva, interminable, poco eficiente y contiene muchos errores humanos ya que la correcta navegación del sitio completo cuando no hay buscador interno depende en gran medida de la buena estructura visual que el diario presenta a sus usuarios.

Además cabe destacar que la búsqueda de *google* no es eficaz para noticias ya que la actualización de las bases de datos no se realiza todos los días, de tal manera que pudiendo existir una noticia de interés en la red, *google* no la detectará a menos que su desarrollo implique el indexamiento para el buscador de *google* lo que en México no ocurre en ningún diario antes de las 12 del día y únicamente si la nota es de relevancia nacional o si se efectúan miles de búsquedas sobre el tema.

1.2. Problemática

Partiendo del marco anterior si uno tiene alrededor de 80 temas de interés y necesita buscar estos en más de 50 diarios, todos los días sin excepción, incluyendo días festivos y con un límite de tiempo cada día, podría pensar que la tarea es incumplible o que requiere de muchos recursos computacionales y humanos siendo una tarea larga y que conlleva pérdida de información por su magnitud y por la naturaleza de la presentación de la información en los portales *web*.

Es aquí donde entra la computación y las técnicas de inteligencia artificial que permiten automatizar las tareas más complejas con resultados que mejoran incluso los análisis del ser humano en algunos sentidos como la rapidez y confiabilidad, evitando al máximo los errores humanos.

Buscando una solución a tal problema se integraron áreas como teoría de la información, inteligencia artificial, programación, ingeniería de *software* y se unieron en la búsqueda de un programa que «lea» la información contenida en los diarios electrónicos de México y permita obtener información básica como título de la nota, fecha y cuerpo, además de indicar si alguno de los temas de interés se encuentra en el cuerpo de la nota y si es así se presente el contenido en una interfaz amigable a los usuarios finales.

1.2.1. Situación Actual

Los casos en los cuales se necesita información de las noticias que publican los diarios en línea son cada vez más frecuentes, la búsqueda automática de tópicos determinados es un tema de estudio amplio y que cuenta con herramientas que están en constante cambio, el Instituto de Ingeniería siendo hogar de la investigación y la aplicación del conocimiento englobado por las ciencias fue una de las instituciones que tuvo la necesidad de encontrar en la *web* información de interés con las características descritas en los apartados anteriores y en la búsqueda de la solución a este problema se estudiaron los casos de las agencias de noticias, se estudiará el caso de una empresa de monitoreo de medios cuyos temas de análisis van desde el ámbito comercial al ámbito político y social.

Dado que es de vital importancia el análisis de tal cantidad de información en un rango de tiempo ya estipulado, actualmente se cuenta con 3 personas que revisan los 9 diarios principales de la ciudad de México: Reforma, Universal, La Jornada, Milenio, La Crónica, El Financiero, El Economista, Excelsior, Uno mas Uno.

Se tienen 81 temas de interés, estos temas son los que se buscan en las noticias web, cada tema es buscado a través de palabras clave, por ejemplo: si el tema es Agrobio las palabras clave manejadas son: Biotecnología, Transgénicos, Agroalimentos, Agrobio, Codex Alimentario, etc.

Con los recursos mencionados se recopila la información necesaria entre las 6:00 am y las 9:00 am, además a lo largo de la mañana y hasta las 11:00 am se revisan de manera general alrededor de 40 diarios más.

El procedimiento es el siguiente:

1. Se abre el portal *web*
2. Se utiliza el buscador del diario para ubicar la aparición de una palabra clave en alguna nota dentro del diario.
3. Si la búsqueda arroja apariciones se guarda la liga en un documento junto con el tema del cual provenía la palabra clave ubicada.

4. Si el diario no tiene buscador se utiliza la búsqueda avanzada de google.
5. Se repite el proceso con todas las palabras clave que se tienen.
6. El documento se envía a un coordinador, este se encargará de enviar las notas a los analistas y estos decidirán si la nota es de interés para el tema tratado.
7. Las notas que son de interés se capturan y se muestran en las páginas correspondientes al tema.

1.2.2. Herramientas existentes

El grupo de Ingeniería Lingüística de la UNAM, realizó una aplicación en java que permite clasificar contenido de cualquier archivo de texto electrónico, está dirigido a la semántica de la información, y a la clasificación de contenidos a través del algoritmo de agrupamiento por contextos definitorios planteado en una tesis de maestría con resultados satisfactorios, sin embargo no es una aplicación *web*, la administración no fue planeada para múltiples usuarios y no se ha probado con las cantidades de información que se ambiciona manipular.

Existe un *software* libre que copia un sitio web a una maquina local, WebHTTrack, es un programa desarrollado en 2007 que aún posee deficiencias como lentitud y un nivel medio de automatización.

En cuanto a las herramientas comerciales existe un *software* que permite obtener el sitio *web* completo en cuestión de minutos, Teleport Pro[©], cada archivo alojado en el servidor que contiene una liga o un camino a la página principal (al nodo raíz); trabaja en base a lo que en informática se conoce como araña *web*, y tiene la opción de discriminar páginas según una lista de contenido. Se puede pensar en que esta es la opción para disminuir la carga de trabajo y sólo llegar al punto de análisis, pero se presenta el problema de que la lista de temas y de palabras clave es grande y se sigue incrementando conforme a las necesidades de los temas analizados, además una vez que la lista es establecida no hay manera de diferenciar cual archivo pertenece a cada tema de manera que sigue siendo necesario el procesamiento de información en los archivos de texto digitales.

1.3. Solución

1.3.1. Objetivo

El objetivo es desarrollar un sistema de procesamiento automático de información digital, que busque temas de interés para el usuario final en archivos de texto electronicos y que presente los resultados de la manera más rápida posible, que haga la clasificación pertinente para saber si pertenece a un grupo de noticias y si no es así que ya no se procese el archivo, es decir si no es una noticia queda descartado del procesamiento principal.

1.3.2. Propuesta

Se propone realizar la programación de una red neuronal artificial basada en la teoría de resonancia adaptativa (ART por sus siglas en inglés *Adaptative Resonance Theory*) analógica que permita clasificar cuáles archivos de texto electrónicos son noticias y cuáles son sólo anuncios o páginas principales de secciones. La obtención de los archivos en línea se realizará a través del programa Teleport[©] del que se habló en la definición de la problemática, estos

archivos quedarán separados en carpetas diferentes según el diario al que pertenezcan y lo que se desea es clasificarlas por el contenido de la información que posee el cuerpo de la nota. Para lograr identificar la estructura del archivo html y obtener la información de interés se utilizará el concepto de reconocimiento de patrones, el patrón es una secuencia de características que se repiten en los html de cada diario *online*, de manera que reconociendo a qué patrón pertenece el archivo se identifican rápidamente características importantes de la noticia como son: el contenido de la nota, el autor, el título, etc.

Una vez obtenidos los archivos de cada diario se debe ejecutar un programa que elimine aquellos que no se deben procesar, es decir ejecutar una red neuronal ART que permita descartar aquellos archivos que no cumplen con ninguna clasificación de las que ya tiene almacenada en sus patrones. Una vez descartados los archivos que no son noticias se debe procesar la información para encontrar las notas que son de interés y guardarlas en una tabla donde se guarde la relación entre la nota y el tema encontrado. Las que no contienen ningún tema de interés deberán guardarse haciendo la distinción de que no poseen temas de interés.

1.3.3. Justificación

Se eligió trabajar con una red neuronal artificial por que el procesamiento computacional que requiere este método es menor que el necesario cuando se comparan patrones con el método de programación tradicional, el esquema de las redes neuronales permite almacenar información implícita en la estructura de la red tal como se hace en la estructura cerebral, de manera que obtener un resultado de esta forma es muy eficiente. Las redes neuronales son compresores naturales de información y su estructura y algoritmo evita las comparaciones excesivas.

Se eligió el modelo ART en primera instancia por que es aprendizaje no supervisado, de esta manera no importa el número de patrones existentes, y el usuario no tiene obligación de preclasificar el entrenamiento. Además las redes ART tienen la ventaja de que su esquema trabaja buscando un equilibrio en el dilema estabilidad-plasticidad, ofreciendo memoria y aprendizaje al mismo tiempo[4].

El desarrollo de un *software* a partir de los motores de búsqueda es un proyecto de investigación largo y que involucra conceptos de estructuras de datos, inteligencia artificial, semántica, etc; por lo que el *software* propuesto tiene un punto de inicio: se utiliza el programa Teleport© que a través de hilos recorre los archivos *web* en búsqueda de más enlaces a otras páginas descargando cada archivo que puede analizar facilitando el procesamiento de la información de interés.

Los lenguajes de programación *web* pueden ser ejecutados desde la máquina cliente o desde el servidor[15], son interpretados y tienen una menor velocidad de ejecución que los lenguajes que no se utilizan para *web* como C[6]. Actualmente existen muchos lenguajes *web* que son variantes de los lenguajes más conocidos. En el mundo laboral y académico los lenguajes más conocidos son:

Java, Javascript, PHP, ASP, JSP, Python y Ruby[32].

Lenguaje	Ventajas	Desventajas
Javascript • Se ejecuta del lado del cliente	• Rápido • Poderoso	• Códigos visibles • Su ejecución requiere completa descarga

Lenguaje	Ventajas	Desventajas
<ul style="list-style-type: none"> • Orientado a eventos, tiene objetos pero no herencias 	<ul style="list-style-type: none"> • Compatible con los principales navegadores • No requiere servidores instalados • Es libre 	<ul style="list-style-type: none"> • No tiene documentación oficial • Varía entre diferentes navegadores
PHP	<ul style="list-style-type: none"> • Lenguaje multiplataforma • Conexión con la mayoría de los manejadores de base de datos • Capacidad de expandir su potencial utilizando módulos • Posee documentación • Es libre • Incluye gran cantidad de funciones • No requiere definición de tipos de variables ni manejo detallado de bajo nivel 	<ul style="list-style-type: none"> • Necesita tener instalado Apache • La programación orientada a objetos es aún muy deficiente para aplicaciones grandes. • Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número. • La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.
ASP	<ul style="list-style-type: none"> • Usa Visual Basic Script, siendo fácil para los usuarios. • Comunicación óptima con SQL Server. 	<ul style="list-style-type: none"> • Código desorganizado. • Se necesita escribir mucho código para realizar funciones sencillas.

Lenguaje	Ventajas	Desventajas
<ul style="list-style-type: none"> • ASP no necesita ser compilado para ejecutarse 	<ul style="list-style-type: none"> • Soporta el lenguaje JScript (Javascript de Microsoft). 	<ul style="list-style-type: none"> • Tecnología propietaria. • Hospedaje de sitios <i>web</i> costosos.
<p data-bbox="359 568 485 595">ASP.NET</p> <ul style="list-style-type: none"> • ASP no necesita ser compilado para ejecutarse 	<ul style="list-style-type: none"> • Completamente orientado a objetos. • Incremento de velocidad de respuesta del servidor. • División entre la capa de aplicación o diseño y el código. • Facilita el mantenimiento de grandes aplicaciones. 	<ul style="list-style-type: none"> • Mayor consumo de recursos. • Tecnología propietaria. • Es necesario Internet Information Server (IIS).
<p data-bbox="392 954 448 981">JSP</p> <ul style="list-style-type: none"> • Se ejecuta del lado del servidor. • Desarrollado para la creación de aplicaciones <i>web</i> potentes 	<ul style="list-style-type: none"> • Código separado de la lógica del programa. • Las páginas son compiladas en la primera petición. • El código JSP puede ser incrustado en código HTML. • Ejecución rápida de servlets. • Crear páginas del lado del servidor. • Multiplataforma. • Código bien estructurado. • Integridad con los módulos de Java. • La parte dinámica está escrita en Java. 	<ul style="list-style-type: none"> • Para su funcionamiento se necesita tener instalado un servidor Tomcat. • La legibilidad del código puede verse afectada al mezclar sentencias HTML y Java. • La complejidad del lenguaje inherente al lenguaje java

Lenguaje	Ventajas	Desventajas
<p>Python</p> <ul style="list-style-type: none"> • Lenguaje de programación multiparadigma 	<ul style="list-style-type: none"> • Permite la utilización se servlets. • Lenguaje de propósito general. • Gran cantidad de funciones y librerías. • Sencillo y rápido de programar. • Multiplataforma. 	<ul style="list-style-type: none"> • Curva de aprendizaje elevada.
<p>Ruby</p> <ul style="list-style-type: none"> • Lenguaje interpretado de muy alto nivel y orientado a objetos 	<ul style="list-style-type: none"> • <i>Software</i> libre. • Multiplataforma. 	<ul style="list-style-type: none"> • Poca documentación existente

Para el proyecto es importante que el lenguaje de programación sea libre ya que el uso de Teleport© ya involucra un costo. De los lenguajes de programación disponibles como *software* libre los que tienen mayor potencial son Java, Ruby, Python y PHP, por lo que la elección de lenguaje se realizó entre estos. Por un lado la estructura robusta de java ofrece confiabilidad, escalabilidad y un nivel alto de seguridad; la similitud de python con el lenguaje natural proporciona una capacidad particular para resolver problemas de manera más intuitiva y en menos líneas de código y PHP brinda mayor capacidad de abstracción, potente manejo de datos y una gran cantidad de funciones disponibles que facilitan la programación de grandes aplicaciones.

La naturaleza del problema exige un manejo exhaustivo de archivos y cadenas, un servidor seguro, la mayor velocidad de procesamiento y el menor costo de recursos computacionales por lo que el lenguaje elegido es PHP, ya que gran parte de su sintaxis es muy parecida a C lo que facilita su aprendizaje y hereda de perl su buen manejo de cadenas. Además de ser *software* libre su servidor apache se encuentra en diversas aplicaciones, tiene un mayor nivel de seguridad que Tomcat y es un lenguaje ya instalado en el servidor de producción y en los servidores de aplicación.

1.3.4. Metodología

1.3.4.1. Desarrollo del proyecto

- Se implementará la red neuronal ART.
- Se probará la red otorgando diferentes valores en los parámetros para determinar con cuales se obtienen los mejores resultados.
- Se implementará el sistema de codificación de datos de entrada.
- Se integrará el sistema de la red ART con el sistema de manipulación y presentación de datos.

- e) Se realizarán pruebas globales.
- f) Se evaluarán los resultados obtenidos con el sistema.

1.3.4.2. Desarrollo de la ART

- a) Se estudiará la estructura de los archivos a analizar para determinar las características del vector de entradas.
- b) Se elaborará un método de codificación.
- c) Se determinará el número de capas de la red neuronal.
- d) Se programará el método de aprendizaje.
- e) Se probarán las salidas que brinda la red de resonancia adaptativa con un grupo de entradas de prueba.
- f) Se analizará la veracidad de las salidas arrojadas.
- g) Se elaborarán las conclusiones.

1.3.4.3. Desarrollo del Sistema

- a) Se listarán y analizarán los requerimientos del usuario final.
- b) Se elaborará la base de datos necesaria para cubrir los requerimientos, la base cumplirá con los principios básicos de diseño de base de datos.
- c) Se organizará el sistema en módulos, se programará la realización de los módulos y se determinará la plantilla de apariencia del sistema.
- d) Se analizarán y elaborarán los algoritmos base del sistema.
- e) Se probarán los algoritmos con datos reales para determinar su eficiencia y determinar si los resultados arrojados cumplen con los objetivos planteados.
- f) Se elaborará la parte de administración del sistema.
- g) Se liberará el *software* para las primeras pruebas con el usuario final.
- h) Si el usuario final lo pide, se harán las correcciones necesarias.
- i) Se evaluará qué tanto mejora el proceso analizado con la implementación del *software*.
- j) Se elaborarán las conclusiones del trabajo realizado.

1.4. Organización de la tesis

La tesis consta de siete capítulos y tres apéndices que abarcan el proyecto en su totalidad. A continuación se describe brevemente el contenido de cada uno de ellos.

Capítulo 1. Se presenta el contexto del problema y se propone una solución de manera general.

Capítulo 2. Se describen los conceptos teóricos y prácticos necesarios en la elaboración del proyecto.

Capítulo 3. Se describe el módulo de implementación de ART2.

Capítulo 4. Se describe el proceso de desarrollo del sistema.

Capítulo 5. Se describen las características de seguridad del sistema inherentes a las herramientas utilizadas y las características de seguridad implementadas.

Capítulo 6. Se analizan los resultados y se obtienen estadísticas que avalan el buen funcionamiento del sistema.

Capítulo 7. Se listan las conclusiones de la elaboración del trabajo mostrando ventajas y desventajas del uso del sistema y las mejoras que este aporta.

Capítulo 2

Marco Teórico

2.1. Herramientas para el diseño e implementación

El desarrollo de una aplicación como la que se planea está basada en muchos conocimientos y aplicaciones cuya utilidad y buen funcionamiento han sido demostrados, la base del sistema es la programación y el lenguaje de alto nivel elegido es PHP.

2.1.1. PHP

A partir de la revolución informática surge la necesidad de la creación de páginas *web* dinámicas no sólo para obtener la simpatía de los usuarios finales y con esto mayores ganancias sino también para un mejor intercambio de información a favor de facilitar el proceso de captura y consulta del conocimiento. La arquitectura cliente-servidor es la que alberga el dinamismo en las páginas *web*, funciona a través de peticiones que son atendidas en la medida de la capacidad y los recursos disponibles[1]. En la comunicación dinámica *web* el cliente hace una petición a través de una página y espera una respuesta del servidor o simplemente manda información al servidor y este responde la petición o recibe la información enviada para procesarla o almacenarla. Para cumplir con el objetivo de comunicar cliente con servidor a nivel aplicación se necesita un lenguaje de programación que tenga la capacidad de crear contenidos *web*, PHP cumple con esta característica, es un lenguaje interpretado usado para la creación de aplicaciones, desde formularios simples hasta sistemas robustos[6].

Como PHP es un lenguaje enfocado a la funcionalidad de las páginas y no a la apariencia de estas es necesario utilizar los estilos, a manera de presentar un sistema atractivo al usuario no solo en la funcionalidad sino también en lo visual y familiar, esto se logra con CSS.

2.1.2. CSS

Es bien sabido que las páginas *web* se construyen en base a un lenguaje de marcado de hipertexto (*HTML*), que estructura y presenta información en un formato estándar. No obstante y a pesar de ser un lenguaje de presentación visual, no fue creado para diseñar publicaciones porque limita gráficamente los sitios.

En 1996 apareció CSS (Cascading Style Sheets), un lenguaje de estilo en cascada que se puede utilizar conjuntamente con *HTML* para definir el formato visual de una página *web*, sin forzar el código para adaptar los documentos al diseño de presentación y sin colocar elementos en la página que no favorecen el contenido.

Las publicaciones digitales de 1990 eran diferentes a las que se realizan hoy en día, ya que ahora podemos observar imágenes, animaciones y archivos multimedia. Por ello, el siguiente reto de la red radica en la adopción de tecnologías para controlar la presentación de las páginas, sin involucrar el contenido.

En 1994 Hakon Lie publicó el primer borrador sobre las hojas de estilo en cascada y *HTML*, con la idea de que el lenguaje de estilo de la *web* debe combinar, de algún modo, las preferencias del autor y el usuario. Su propuesta la llevaba a cabo con una cascada que tenía en cuenta múltiples conjuntos de reglas de estilo.

De este modo, en 1994 se creó la *World Wide Web Consortium* (w3c.com) para establecer los estándares técnicos del crecimiento y el desarrollo de la *web*.

Los beneficios más importantes del uso de las hojas de estilo externas son el aumento en la accesibilidad y la disminución del trabajo de producción, así como la fácil labor de mantenimiento de los sitios, evitando la búsqueda entre líneas de código[31].

Asimismo, al proporcionar un nivel de abstracción, las CSS incorporadas en la cabecera de un documento mejoran el código del mismo, el tiempo de respuesta y la velocidad de presentación de la página, siendo posible declarar una regla de estilo una sola vez y hacer que las condiciones de presentación se apliquen a todos los elementos de ese tipo[22].

La importancia de utilizar hojas de estilo en cascada para la presentación de documentos en *internet*, además de separar contenido de presentación, es que ofrecen facilidad en la producción y rapidez en la accesibilidad.

Una vez que se toca el tema de la apariencia en las páginas web que usará el usuario final se llega al punto en el que se busca equilibrar la eficiencia con la buena presentación, además los sitios en los cuales se requiere hacer peticiones a una base de datos deben ser ligeros para evitar aumentar la lentitud de las peticiones al servidor, una herramienta que permite actualizar solo segmentos de una página web sin necesidad de cargar de nuevo hojas de estilo ni imágenes es AJAX.

2.1.3. Ajax

Existen muchas herramientas que los desarrolladores de aplicaciones *web* utilizan al realizar su trabajo, por ejemplo, el lenguaje JavaScript permite controlar tareas, como el presionar un botón o una liga, que se generan en el navegador de los usuarios que visitan una página *web*; mientras que otros lenguajes o plataformas como PHP (PHP Hypertext Pre-processor), ASP (Active *Software* Professionals), JSP (Java Server Pages), entre otros, procesan la información en el servidor y regresan el resultado del proceso a la máquina cliente para desplegarlo en el navegador del usuario.

En este sentido, cada vez que en el navegador del usuario se genera una petición al servidor y se despliega la información resultante, se debe redibujar toda la página. Si tomamos en cuenta las imágenes, tablas, listas, menús y, en general, todo lo que conforma la página *web* de inicio antes de efectuar una petición, y la que aparece al actualizar la página, nos podemos dar cuenta de que es “inútil” borrarla cuando en un 90 % se redibujará tal y como estaba antes de hacer cualquier petición. Dicho procedimiento se puede optimizar a través de AJAX (Asynchronous JavaScript And XML, por sus siglas en inglés).

A la letra ‘A’ del acrónimo AJAX le corresponde la palabra *Asynchronous* (asíncrono). Al hacer las peticiones de esta manera, es decir de forma asíncrona, podemos mostrar la información que el servidor retorna, sólo en las secciones de la página en donde nos interesa mostrarla, sin necesidad de actualizarla por completo.

AJAX es una combinación de JavaScript, que trabaja del lado del cliente, y de lenguajes que procesan la información en el servidor y la entregan como una cadena de texto o en un

archivo XML, lo que proporciona una gran posibilidad de manejo de los datos, en conjunto con las peticiones asíncronas arriba mencionadas.

Podemos decir entonces que la combinación de estas tecnologías y técnicas de programación es lo que define a AJAX.

AJAX permite desarrollar aplicaciones *web* mucho más atractivas para el usuario, puesto que son más ágiles y de respuesta inmediata. El número de eventos que un usuario debe realizar para terminar por ejemplo, un registro en línea, disminuye de manera significativa, aunado a que el número de páginas que se despliegan para concluir un proceso también se reduce, lo que permite optimizar el tiempo que el usuario invierte en realizar cualquier actividad.

Sin embargo, las desventajas que podría tener AJAX son:

- La parte medular de su funcionamiento está basada en código JavaScript, lo que no garantiza que el navegador del usuario de la página *web* lo soporte.
- El uso común de los botones **adelante** y **atrás** del navegador *web* se modifica de alguna manera, lo que puede confundir al usuario cuando no tiene presente esta situación al desarrollar los sistemas.
- Aunque es difícil encontrar información bibliográfica relacionada con este tema, en Internet existen sitios como el de Mozilla Developer:
 - http://developer.mozilla.org/en/docs/AJAX: Getting_Started que pueden ser de mucha ayuda para comenzar a entender de forma práctica el funcionamiento de AJAX[13].

Las herramientas de desarrollo son vitales para la funcionalidad del sistema, pero el programa debe encontrarse alojado en un ambiente adecuado, se requiere un servidor que permita la comunicación entre el usuario final y las aplicaciones del sistema, uno de los servidores más comunes es Apache.

2.1.4. Apache

Es un servidor *web*, recibe y contesta las peticiones de parte de un navegador o cliente *web*, usualmente cuando se habla de un servidor *web* nos referimos a una máquina en especial que está dedicada únicamente a esta función, sin embargo, se debe tener en cuenta que cualquier computadora conectada a Internet puede ser un servidor siempre que se le instale el *software* adecuado. Apache es un *software* gratuito, originalmente desarrollado para UNIX que permite que una máquina pueda atender peticiones de manera que cualquier máquina con Apache se comporta como un servidor[18].

El servidor aunque es sumamente potente y útil siempre requiere el complemento de las bases de datos para poder realizar aplicaciones robustas, es necesario un administrador de estas bases de datos para poder gestionar una gran cantidad de datos, una herramienta que permite realizar todas las transacciones necesarias para el proyecto actual es MySQL.

2.1.5. MySQL

Es un sistema de administración de bases de datos relacionales, rápido, sólido y flexible. Es ideal para crear bases de datos con acceso desde páginas *web* dinámicas, para la creación de sistemas de transacciones en línea o para cualquier otra solución profesional que implique almacenar datos, teniendo la posibilidad de realizar múltiples y rápidas consultas. Es de

licencia pública, fue realizado en c++ y se descarga de manera gratuita en la red. Permite administrar múltiples bases de datos, maneja alrededor de 30 tipos de datos en los atributos de las tablas; permite restringir el tamaño de los contenidos de cada atributo y clasificarlo como atributo normal, llave primaria, índice de tabla, etc.

Además MySQL es compatible con diferentes interfaces de administradores, una de ellas es phpmyadmin, que funciona en equipos con sistema operativo basado en unix y windows. La interfaz permite una sencilla administración de usuarios, de los datos de la base, importación y exportación de tablas y también se utiliza para corregir errores en las consultas sql realizadas en los sistemas[6].

2.2. Conceptos de Ingeniería de *Software*

El desarrollo del programa no puede realizarse sin planeación, es un proyecto grande y la mala planeación ocasiona trabajo extra y costos como pérdida de tiempo y de información. Una manera de realizar un buen sistema es aplicar los conceptos de ingeniería de *software*, estos ayudan a aprovechar los recursos de la manera adecuada y realizar programas óptimos para resolver el problema que se presenta.

2.2.1. Programación Estructurada

La programación estructurada surge para clarificar los códigos y para simplificar la depuración y el mantenimiento de un programa, se origina de la idea de descomponer la tarea del usuario en partes elementales que son más sencillas de resolver que el problema en su totalidad. Cada una de las partes se define en un módulo que es llamado desde un programa principal.

Una de sus principales ventajas es que a lo largo de su desarrollo se crean bibliotecas de módulos o subrutinas que pueden ser utilizados en otras aplicaciones de manera muy sencilla. Esto es muy importante para el tipo de desarrollo que se realizará en este proyecto, además la documentación resultante es más abundante y fácil de leer, lo que también es muy importante ya que esta aplicación será modificada o servirá de base para automatizar el proceso de interés en mayor medida. Por otro lado, el planteamiento de un problema como programa estructurado resulta más sencillo por la experiencia que se tiene y el tiempo disponible para entregar el *software* completo. De tal manera que este paradigma es el que se utilizará para realizar la aplicación necesaria en el proceso de notas digitales[29].

2.2.2. Ciclo de vida de un sistema de informático

Proceso de *software*

Es un conjunto estructurado de actividades (de programación y administrativas) para desarrollar un sistema de *software*, se compone de varias actividades que varían según la organización y tipo de sistema[26]. La gestión exige disponer de un modelo definido, debido a los grandes avances y difusión de los sistemas computacionales y a las aportaciones que se presentan día con día existen muchos modelos que convienen a diferentes aplicaciones, paradigmas y sistemas de organización.

Las características de un proceso de *software* son las siguientes:

- Facilidad de mantenimiento
- Rapidez (relativa)

- Claridad
- Visibilidad
- Facilidad de soporte
- Aceptación
- Fiabilidad
- Robustez

El proceso de *software* se caracteriza también por una secuencia llamada ciclo de vida, que presenta el orden o estructura en la que los procesos que lo conforman son ejecutados, este ciclo de vida también se conoce como las etapas para el desarrollo del *software*[26]. En general, el ciclo de vida contiene las siguientes etapas:

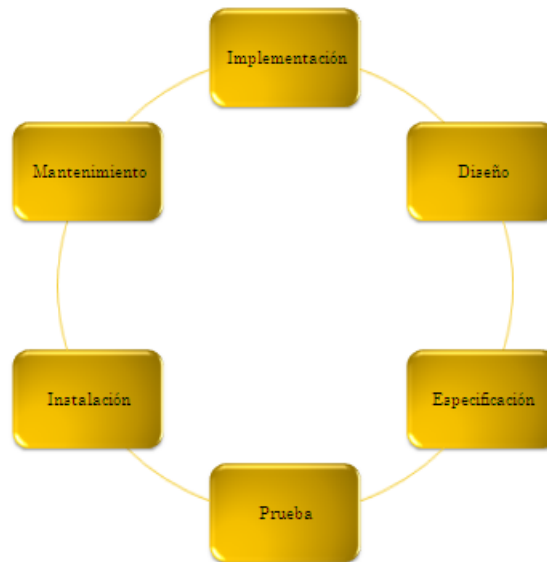


Figura 2.1: Proceso de *software*

2.2.2.1. Modelo en cascada

Este modelo es muy secuencial, en el diagrama se observa que cada etapa depende en gran medida de la anterior y a pesar de esta dependencia es el modelo más simple, más conocido y usado, se caracteriza por que cada fase emite un producto para la siguiente[9].

Ventajas: requiere de disciplina, planeación y administración, la puesta en práctica se pospone hasta que se entienda bien el objetivo.

Desventajas: lineal, rígido, monolítico, no realista. Difícil de introducir modificaciones sobre todo en la fase de mantenimiento, la cual en 20 % es correctiva, otro 20 % adaptativa pero en más del 50 % está dirigida al perfeccionamiento del sistema.

Características: enfocado a la producción de documentación.

En general las etapas de diseño en este modelo son las siguientes:

1. Definición de requisitos. Con la participación del cliente, se definen la funcionalidad, restricciones y objetivos del sistema.
2. Diseño del sistema. El sistema se divide y se establece la funcionalidad del *software*. El resultado es la arquitectura del sistema.
3. Implementación y prueba unitaria.
4. Integración y prueba del sistema
5. Operación y mantenimiento. El sistema se entrega al cliente, se pone en funcionamiento. Se modifica para depurar errores y mejorar su funcionamiento.

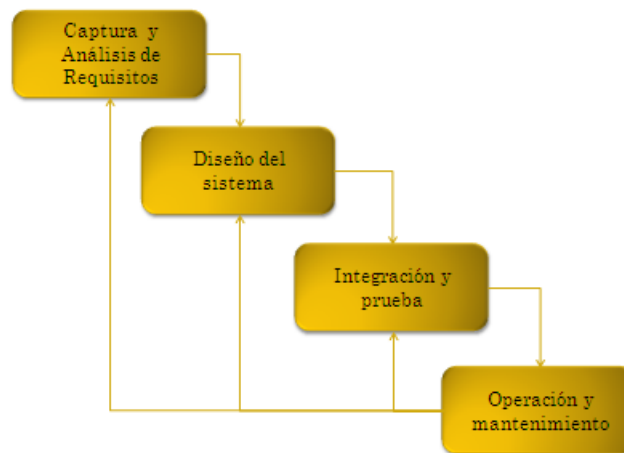


Figura 2.2: Modelo en cascada

2.2.2.2. Modelo Transformacional

Este modelo actualmente no es ni básico ni popular, pero sí es el sueño dorado de los desarrolladores, puesto que propone la posibilidad de construir código a partir de las especificaciones dadas. Esto constituye la base para las herramientas CASE (*Computer-Aided Software Engineering*), donde, utilizando un lenguaje abstracto o gráfico para describir las especificaciones de manera más agradable, la herramienta automatizada se encarga de realizar el código. En el modelo transformador, se realiza la especificación formal del sistema y la herramienta convierte automáticamente esta especificación en código.

Rational Rose puede ser un ejemplo de este modelo porque de una especificación con UML (*Unified Modeling Language*) puede generar código en Java, C++ o VisualBasic. En realidad sólo genera esqueletos, especificaciones de clases y encabezados de métodos pero no sus códigos.

La desventaja es que es bueno para productores relativamente pequeños de áreas limitadas. Las hojas de cálculo y lenguajes de cuarta generación (4GL) –desarrollados en la práctica para el uso de bases de datos– son ejemplos de lenguajes de especificaciones más abstractas que permite generar el código, pero hasta la fecha, estos lenguajes están restringidos a ciertas aplicaciones (como los constructores de la familia Visual) y aún no existen para propósito general.

La característica de este modelo es que se trabaja en el nivel de especificación y no se llega al nivel del lenguaje de programación[27].

2.2.2.3. Modelo Espiral.

La misma idea de prototipos la encontramos en este modelo. En la propuesta de Boehm, la espiral representa muy bien la idea del desarrollo iterativo e incremental, así como la del desarrollo por prototipos, en donde estos al principio abarcan una pequeña parte del sistema, (la evaluación de riesgos) y empiezan en el centro de la espiral.

Es un modelo evolutivo pero en cada vuelta, antes de generar un nuevo prototipo, se evalúa el riesgo que se corre si se continúa con la siguiente iteración, es decir, si aún por parte del cliente y del programador existe tiempo, recursos y decisión para seguir mejorando este proceso de desarrollo o por el contrario, si se encuentra que los riesgos son demasiado grandes, tomar las decisiones adecuadas como la suspensión del proyecto o la solicitud de más recursos. La idea es que antes de generar un prototipo, se evalúe la factibilidad y se analicen los riesgos. Si el resultado es positivo, entonces se genera el primer prototipo (el cual abarca sólo una parte del sistema), se continúa con las fases siguientes (simulación, comparación, validación de requerimientos) hasta que deje de ser prototipo y se convierta en el sistema completo, cuya puesta en operación, ya integrada y aceptada por el cliente, constituye la salida del proceso. Este modelo es iterativo pues repite los pasos incrementándolos cada vez al ir robusteciendo el prototipo en su funcionalidad[20].

2.2.2.4. Modelo evolutivo ó incremental

El modelo Evolutivo conocido también como incremental e iterativo, consiste en hacer la documentación de las fases, realizando un prototipo del sistema, se evalúa el qué tan lejos el prototipo está de la solución final esperada por el cliente; se toman en cuenta las observaciones de esta evaluación, y se crea un nuevo prototipo que las incluya. Esto se realiza en una vuelta repetitiva donde se incrementa el alcance del prototipo en pequeñas proporciones hasta cumplir los requerimientos totales.

El objetivo es vencer las limitaciones del modelo de ciclo de vida en cascada motivadas por la deficiente realimentación entre sus fases[26].

Dentro del mismo modelo evolutivo se tienen dos posibilidades:

■ Prototipo evolutivo.

Objetivo: Poner un sistema a disposición de los usuarios finales.

El proceso comienza con unos requisitos claros. Debe usarse en sistemas en los que no es posible, inicialmente, desarrollar la especificación, por ejemplo: sistemas de inteligencia artificial, interfaces de usuario, etc. Se basa en técnicas que permiten obtener de forma rápida versiones del sistema. La verificación es imposible por no existir especificación. La validación demuestra de forma subjetiva la adecuación del sistema

■ Problemas

- Cambios continuos provocan la destrucción de la estructura del sistema, encareciendo el posterior mantenimiento.
- El tiempo de vida de los sistemas desarrollados con esta técnica será necesariamente corto.
- Es necesario contar con especialistas.

- **Prototipo experimental.**

Objetivo: Validar o identificar los requisitos del sistema.

El proceso comienza con unos requisitos poco claros. Su principal objetivo es clarificar los requisitos y proporcionar a los gestores información para reducir riesgos. El prototipo se desarrolla a partir de una especificación inicial, luego se experimenta con él y, finalmente, se elimina. El prototipo experimental no es un sistema final: puede ignorar algunas características del sistema. El sistema está débilmente estructurado y es difícil de mantener.

- Problemas

- Algunos requisitos (como la seguridad) no pueden clasificarse dentro de un prototipo y no aparecerán en la especificación.
- Los requisitos no funcionales no se prueban adecuadamente en un prototipo.

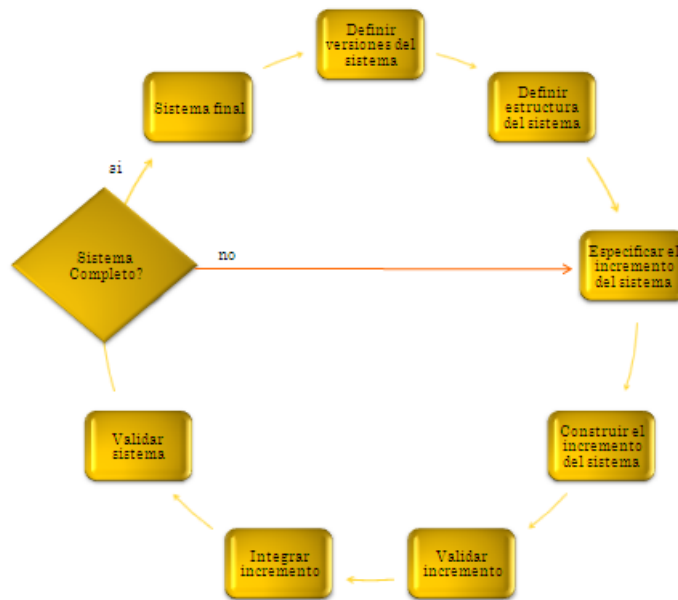


Figura 2.3: Modelo incremental experimental

Un sistema se desarrolla como una secuencia de pasos, una vez establecida la arquitectura global. Los usuarios pueden experimentar con los sistemas resultantes de cada paso. El equipo de desarrollo puede continuar su trabajo mientras los usuarios experimentan con el sistema. El objetivo es combinar las ventajas del prototipado con una mejor gestión y una mejor estructuración del sistema.

Una vez que se valida y libera un componente del sistema, no se cambia a no ser que se detecten errores. Se deben generar documentos de cada paso realizado. Una vez que se libera un componente su interfaz no se cambia, lo que obliga a que los últimos incrementos se adapten a estas interfaces.

Ventajas: ideal para sistemas que no tienen bien definidos los requerimientos.

Desventajas: es difícil distinguirlo del proceso "codifica y corrige".

Características: enfocado a la producción de prototipos.

En resumen, ningún modelo de desarrollo es el mejor ni el peor, la elección de un modelo de *software* depende de :

- La complejidad del problema y su solución.
- Financiación y tiempo disponible.
- Acceso de los desarrolladores a los usuarios.
- Certeza de los requerimientos conocidos.

Para este trabajo se utilizará el esquema que propone el modelo evolutivo en su prototipo experimental para el desarrollo del sistema y para la generación de documentación ya que los requisitos inicialmente no se encuentran definidos al 100 % y la experiencia dicta que se necesitará realizar el proceso de ciclo de vida del sistema más de una vez, no se optó por el modelo en espiral por la evaluación del riesgo, el proyecto cuenta con los recursos necesarios para continuar y esta evaluación perjudicaría el desarrollo por la cantidad de tiempo dedicado a analizar los factores de riesgo, el modelo transformacional queda descartado por que no se cuenta con alguna herramienta para generar código a partir de las especificaciones.

Para programar módulos que no son administrativos, es decir, los que no son encargados de actualizar, borrar y crear registros en la base de datos sino de procesar los datos para obtener la información deseada se requieren estructurar secuencias de procesos que logren un fin determinado, estos procesos son los algoritmos del sistema.

2.3. Algoritmos

Los algoritmos son una serie de pasos que guían hacia un objetivo específico. De manera más formal y comenzando a introducir el concepto en el ámbito computacional un algoritmo es una serie ordenada de pasos precisos que produce un resultado y termina en un tiempo finito. De una definición tan sencilla como esta surgen conceptos de teoría de la computación tales como correctez y complejidad que ayudan a determinar la utilidad del algoritmo en términos computacionales. Sin embargo tener un algoritmo definido no quiere decir que a partir de este se puede encontrar una solución viable, y menos se puede afirmar que si se traduce a lenguaje máquina, la computadora sea capaz de encontrar una solución.

La máquina de Turing(MT) se introdujo en 1936 por Alan M. Turing para resolver problemas computables. Funciona como un autómata capaz de reconocer los lenguajes generados por las gramáticas menos restrictivas, es decir, aquellas cuyos patrones están limitadamente definidos. La utilidad de la máquina de Turing es que a nivel abstracto estas reciben una entrada codificada y generan una salida codificada al final de la ejecución de la MT, por esto se puede realizar una analogía entre una MT y un programa ya que ambos poseen una entrada, una salida y un proceso que permite generar una salida correspondiente a esa entrada. Esta analogía ha llevado a realizar numerosos estudios teóricos sobre la computación, permitiendo asignar la etiqueta de «computable» a cualquier proceso que pueda representarse mediante una máquina de Turing, y más aún, debido a que se puede definir una MT Universal (MTU) como una MT capaz de ejecutar / simular cualquier MT y éstas son análogas a los programas se puede establecer una analogía entre las MTU y las computadoras[28].

El por qué estudiar si un problema es computable o no, si la implementación de un algoritmo va a terminar o no, cuanto tiempo llevará en brindar una solución y qué tan importante es el lenguaje de programación elegido para la resolución de un problema son cuestiones de la teoría de la computación y son relevantes en cualquier proyecto de investigación que ofrezca un camino computacional ante problemas cotidianos. En particular para

este proyecto se emplean algoritmos de ordenación, de búsqueda y de aprendizaje, de los cuales se demostrará que funcionan de manera correcta, y que terminan su ejecución, además se brindará un análisis de complejidad ya que el número de entradas es muy grande y se desea que el programa sea lo más eficiente posible.

2.3.1. Complejidad

Si partimos de que un problema sí puede resolverse con una computadora, la siguiente cuestión es cuánto tiempo le tomará a la máquina encontrar la solución, se puede tomar el tiempo de ejecución pero este depende en gran medida de la potencia del recurso computacional por lo que los cálculos varían de una máquina a otra, se puede también calcular la memoria requerida, pero no es un cálculo sencillo. De esta manera surge el concepto de complejidad como el número de operaciones realizadas por la computadora cuando esta ejecuta el programa. Por esto la complejidad es independiente de la velocidad de la computadora que ejecuta el programa y además esta mediada ofrece una relación directa entre la complejidad y el número de entradas.

La complejidad de un algoritmo esta denotada por $O(n)$, donde n hace referencia al número de entradas. Y es una medida de la eficiencia de los algoritmos[17].

2.3.2. Correctez

Otra característica de interés de los algoritmos es la correctez, ya que de nada sirve un algoritmo muy eficiente que no cumple la función para la que fue programado. El análisis de correctez no es inmediato, en primera instancia se debe analizar cada submódulo del programa para verificar que el resultado arrojado sea el que se desea. Otro recurso que contribuye a garantizar el correcto funcionamiento de los algoritmos es encontrar condiciones que permanecen siendo válidas a lo largo del algoritmo, a esto se le llama invariante y permite establecer puntos de partida para evaluar las variables modificadas en el programa y probar si se llega a la solución esperada[17].

2.4. Redes neuronales artificiales

El trabajo con las redes neuronales artificiales, comúnmente llamadas sólo redes neuronales en el ámbito de la computación es una solución cada vez más estudiada por tomar un camino diferente al más utilizado en el estudio de los problemas no determinísticos.

Si se compara el cerebro humano con una computadora inexistente aún, muy compleja, no lineal y paralela elaborar un algoritmo que emule el pensamiento humano en la resolución de un problema es muy complejo, los intentos de esto ha llevado a la creación de los lenguajes funcionales ya que la complejidad de un algoritmo así, no solamente es elevada por el valor de n (número de entradas necesarias para una correcta evaluación) sino por que la cantidad de operaciones necesarias para obtener una respuesta crece de manera exponencial partiendo de n . Existe un gran número de problemas en ciencia e ingeniería que implican la extracción de información a partir de datos complejos e inciertos, para muchos de estos problemas las aproximaciones tradicionales resultan inadecuadas, así que la simplificación de los datos partiendo, por ejemplo, de métodos estadísticos avanzados es un camino viable que está en proceso de investigación. A lo largo de este proceso se ha trabajado con aproximaciones al comportamiento biológico cuando se hace frente a algún problema que requiera listar o buscar una solución a lo largo de muchas soluciones posibles[11].

Dos de los resultados de este proceso son:

- Los sistemas expertos: cuya aproximación a la experiencia se realiza a través de una lista de reglas memorizadas y la capacidad de inferencia en base a esas reglas.
- Las redes neuronales artificiales (RN): que tratan de reproducir de manera general el esquema neuronal tanto en las unidades de proceso (neuronas) como en las conexiones y la creación de estas, sin embargo no presentan la complejidad de los sistemas biológicos ni se toman en cuenta todas las variables existentes en ellos.

El uso de los sistemas expertos es recomendable cuando la situación de interés puede definirse completamente mediante reglas, cuando no es así, se utilizan las redes neuronales que presentan un mejor rendimiento cuando hay incertidumbre o datos incompletos en el conocimiento empleado.

Las redes neuronales son sistemas computacionales que constan de un gran número de elementos simples (neuronas) interconectados que procesan información respondiendo dinámicamente frente a unos estímulos externos, desde el punto de vista computacional son un procesador masivo, paralelo y distribuido formado por unidades simples que tienen una tendencia natural para adquirir conocimiento a través de la experiencia y es capaz de utilizar este conocimiento en diversas situaciones. Esto asemeja al cerebro humano en dos aspectos:

- La red adquiere conocimiento de su ambiente a través de un proceso de aprendizaje.
- Las conexiones interneuronales, conocidas como pesos sinápticos, son usados para almacenar el conocimiento adquirido[11].

De manera que existen 2 componentes principales en la red neuronal: las neuronas y las interconexiones (arquitectura de la red).

2.4.1. Elementos

Los elementos de las redes neuronales artificiales han sido nombrados como algunos de los elementos de las redes neuronales biológicas tratando de imitar de manera general el proceso de entrada de datos, procesamiento interno y proporción de una respuesta. Sin embargo el nivel de simplificación efectuado en las RN es muy alto por lo que en la actualidad algunos elementos de las redes neuronales artificiales ya difieren de los elementos biológicos en los nombres de las entidades que lo componen.

Perceptrón

La neurona de la red biológica inspiró el origen del perceptrón o unidad de procesamiento, sin embargo a este perceptrón también se le llama neurona en el contexto computacional, cuando se quiere hacer alusión a la neurona biológica se especifica esto y cuando se omite se habla generalmente del perceptrón.

En el contexto de la red neuronal artificial, la neurona o perceptrón representa el componente mínimo de la red, su funcionamiento es muy simple ya que sólo transforma varias señales de entrada en una única salida, las entradas pueden proceder de otras neuronas o bien ser entradas a la red desde el exterior. La salida puede transmitirse a otras neuronas o funcionar como señal de salida de la red, aunque generalmente las funciones aplicadas a la salida de red son diferentes a las aplicadas en las salidas internas.

Cada una de estas unidades puede tener múltiples entradas que son asociadas a propiedades diferentes del proceso de interés y además se tiene una entrada opcional denominada *bias*, que permite mantener un umbral en la salida y sirve para modular una tendencia en el peso de la neurona, puede ser un valor constante o puede modificarse en el proceso de entrenamiento[11].

Arquitectura

Se trata de la manera en la que se interconectan las unidades mínimas, y la descripción de las entradas y salidas de estas unidades, para establecer una organización estándar se agrupan las neuronas base en capas, cada capa puede ser de alguno de los tres tipos siguientes:

- Entrada. Donde se reciben los datos iniciales que se desea procesar.
- Oculta. Donde se procesan los datos ingresados
- Salida. Donde el usuario puede observar los resultados del procesamiento.

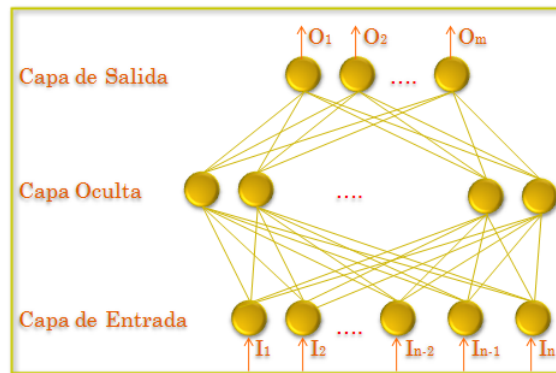


Figura 2.4: Arquitectura típica de una RN

En una capa de entrada, se encuentran las neuronas cuyas entradas son externas al sistema neuronal y la salida esta conectada con otra capa de neuronas.

En una capa oculta las entradas de la neurona provienen necesariamente de otras neuronas y las salidas se conectan a otra capa de neuronas.

En la capa de salida las neuronas reciben entradas de otras neuronas y la salida queda fuera del sistema neuronal[11].

2.4.2. Funcionamiento

Actualmente existen varios tipos de redes neuronales pero de manera general el proceso de implementar una red neuronal consta de las siguientes etapas:

1. Elección del tipo de red. Existe una gran variedad de modelos neuronales de manera que para elegir uno y trabajar con el se deben tomar en cuenta las necesidades del problema analizando las siguientes características:
 - a) Preclasificación ó aprendizaje. Con el análisis de esta característica se determinará si la red a utilizar será de aprendizaje supervisado o no supervisado, cuando no se conocen *a priori* las clasificaciones que se desean obtener se utiliza el aprendizaje no supervisado, cuando se tiene una determinada muestra de datos de entrada con salidas conocidas se puede utilizar tanto aprendizaje supervisado como no supervisado.

- b) Medida del error. Todas las redes neuronales tienen un gran porcentaje de tolerancia a fallos debido a que la forma en la que se almacena información es redundante, de manera que el error de las redes neuronales es bajo en la mayoría de los casos cuando se tienen situaciones separables, sin embargo existen muchas maneras de medir el error a través de diversas funciones, el error cuadrático es el más común pero la medida del error también se puede modificar según se necesite en la resolución del problema.
 - c) Capacidad computacional. Existen redes cuyo algoritmo de aprendizaje es más complejo, computacionalmente hablando, como las máquinas de Boltzmann, esto ocasiona que la velocidad del proceso de la red neuronal dependa del equipo de cómputo utilizado; cuando el problema que se está atacando es por sí solo un problema complejo con un gran número de entradas es recomendable utilizar algoritmos de aprendizaje más simples como Retropropagación o ART.
 - d) Tipos de entrada. Los vectores de entrada pueden tener valores binarios o analógicos, cuando no se requieren números analógicos se puede manejar una máquina de Boltzmann y Cauchy, ART1 o la red discreta de Hopfield, si el problema necesita ser representado con números reales se pueden utilizar las redes de retropropagación, ART2, Hopfield continuo, mapas autorganizados, etc.
2. Definición de arquitectura y parámetros. Para definir la arquitectura se debe establecer cual es el número de datos de entrada, y cuántas características de salida espera el usuario, para determinar el número de neuronas en la capa oculta se pueden probar varios valores y determinar con cuál se obtiene el mejor comportamiento. El número de capas ocultas no debe ser mayor a dos, ya que está comprobado que el resultado final no mejora cuando se añaden mayor número de capas ocultas. Incluso si el problema es linealmente separable se pueden utilizar arquitecturas sin capas ocultas. La asignación del valor adecuado para los parámetros se realiza de acuerdo a un algoritmo genético, un sistema experto o a través de la experiencia de un ser humano.
 3. Captura de datos de entrenamiento. La captura de datos tiene origen en el análisis del problema, ya que se muestrean situaciones del problema que brindan observaciones significativas, se estudia la respuesta esperada (o se utiliza un algoritmo de aprendizaje no supervisado) y se organizan los datos para ser ingresados a la red neuronal.
 4. Fase de Inicialización. Las redes neuronales modifican iterativamente los valores de las conexiones entre neuronas, de manera que el valor inicial no es muy significativo a menos que afecte el proceso de clasificación. Es común que el valor inicial de los parámetros del algoritmo no se altere en ninguna fase del proceso.
 5. Fase de aprendizaje ó entrenamiento. Se introducen los valores de entrada a la red y se analizan los resultados, con base a estos la red ajusta sus parámetros hasta obtener los resultados deseados. Algunas de las redes neuronales no cuentan explícitamente con fase de entrenamiento pero esta se crea de acuerdo al valor que se le da a sus parámetros.
 6. Pruebas y análisis. Casi siempre podemos tener *a priori* algún resultado de los datos que utilizamos como entrada de manera que se pueden comparar los resultados empíricos con los que arroja la red, a menos que el problema no sea un problema adecuado para ser abordado por una red neuronal, casi siempre se obtienen buenos resultados de esta evaluación. Si no se tiene un resultado empírico con el cual se puedan comparar

los resultados arrojados por la red se diseñan experimentos que pueden evaluar los valores obtenidos.

7. Ejecución. La red es utilizada con valores de entrada completamente desconocidos y brinda resultados sin problema, con una gran tolerancia a fallos.

2.4.3. Aprendizaje

En general la fase de aprendizaje consiste en tomar un vector de entrada procesarlo con la arquitectura inicial, y verificar que la salida sea la adecuada, si no lo es, modificar la arquitectura a manera de obtener la salida deseada, para aplicar esta metodología es necesario conocer la salida deseada para cada vector de entrenamiento, a esto se le llama aprendizaje supervisado, pero no es el único paradigma, ¿Qué ocurre cuando no se conoce la salida deseada?

2.4.3.1. Aprendizaje no supervisado

Existen técnicas en inteligencia artificial que no requieren conocer el resultado de la evaluación de las entradas, por ejemplo los métodos de clasificación como cuantización vectorial y las redes de resonancia adaptativa, en estas se tienen una serie de vectores que resumen una serie de características codificadas y los algoritmos agrupan basados en patrones representativos sin conocer *a priori* la clasificación correcta. Estas clasificaciones no supervisadas han permitido incluso encontrar relaciones anteriormente desconocidas en áreas como las proteínas en la biología y los síntomas en la medicina general.

2.4.4. Tipos de Red

Actualmente existen muchas variantes de las arquitecturas, reglas de aprendizaje, métodos de comparación, etc. en las redes neuronales, de manera que ninguna arquitectura se ha desechado completamente debido a que puede utilizarse sin problema según la aplicación que se esté desarrollando, las redes más conocidas y a partir de las cuales se han realizado la mayor cantidad de variantes son:

Adaline

ADaptative LINear Element. Su estructura se compone de una sola neurona (Fig. 2.5), por lo que técnicamente no es una red. Sin embargo es un elemento muy importante, ya que de él se derivan redes más complejas. La neurona en sí, es un dispositivo de Combinación Lineal Adaptativo (ALC), allí se realiza la suma ponderada de los componentes del vector entrada.

Se alimenta con un vector de entrada (valores observados) y la entrada x_0 tiene una naturaleza distinta de las demás (constante igual a 1) y se conoce con el nombre de término de tendencia (sesgo o *bias*).

El aprendizaje básicamente se realiza de la siguiente forma:

- La salida S generada se confronta con la respuesta correcta o esperada a través de un elemento comparador.
- La diferencia Δ es realimentada a un bloque de corrección de pesos para minimizarla o anularla por aproximaciones sucesivas, trabaja con entradas y salidas reales.

Esta es la base de los Madalines que son Múltiples Adalines, estos ya pueden considerarse redes propiamente[35].

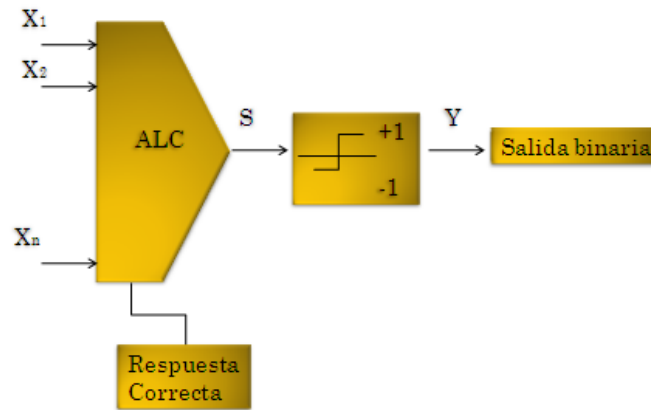


Figura 2.5: Adaline

Máquina de Boltzmann (MB)

Es una Organización Computacional Paralela que es capaz de satisfacer problemas que poseen un gran número de restricciones “débiles”. La MB está compuesta por elementos computacionales primitivos llamados unidades que se encuentran conectados entre todos por enlaces bidireccionales(Fig. 2.6), las unidades pueden tomar valores de 0 ó 1. Las MB son un simple ejemplo de una clase de modelos estocásticos que explotan la relación entre la distribución de Boltzmann y la teoría de la información. Se distingue por manejar estados para representar las características de la MB en un momento determinado, esto se mide con el nivel de energía del sistema. Se puede considerar como un modelo de implementación paralela en forma masiva del algoritmo de recocido simulado. El algoritmo de aprendizaje es supervisado y muy lento[11].

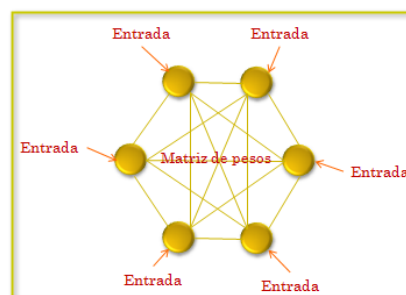


Figura 2.6: Máquina de Boltzmann

Retropropagación

La red está constituida por una capa de entrada, una capa oculta y una capa de salida, es un algoritmo de aprendizaje supervisado y consiste en minimizar un error (comúnmente

cuadrático) por medio de gradiente descendiente, por lo que la parte esencial del algoritmo (Algoritmo 2.1) es cálculo de las derivadas parciales de dicho error con respecto a los parámetros de la red neuronal. Es la red neuronal más conocida e implementada[5].

Algoritmo 2.1 Algoritmo de retropropagación

Calcular la salida de la red a partir de un vector de entrada \mathbf{V}
 Presentar a la red el vector de entrada \mathbf{V}
 Comparar la salida calculada y la que brinda la red
 Ajustar los pesos de cada neurona para minimizar la diferencia de salidas
 Repetir el proceso por cada patrón de entrenamiento

Modelo de Hopfield

Red neuronal artificial recurrente que tiene patrones de conexión sináptica tales que existe una función de Lyapunov (Fig. 2.7) relacionada con la dinámica del sistema. Comenzando de un estado inicial, el estado del sistema evoluciona a un estado final que representa un mínimo local de la función de Lyapunov[14].

$$E = -\frac{1}{2} \sum_{i < j} w_{ij} s_i s_j + \sum_i \theta_i s_i$$

Figura 2.7: Función de Lyapunov

Redes basadas en funciones de base radial

Una red basada en RBF (*Radial Basis Function*) es muy similar en la arquitectura a un Perceptrón Multicapa, pero su forma de aprendizaje es diferente.

Se dice que una función es radialmente simétrica (o es una Función de Base Radial, FBR) si su salida depende de la distancia a una entrada ejemplo (vector) desde otro vector almacenado. Las Redes Neuronales cuyas funciones en los nodos son funciones radialmente simétricas, se dicen que son Redes Neuronales de Funciones de Base Radial. Ejemplos de estas bases son: la función gaussiana, cúbica y multicuadrática.

La topología de la RBF (Fig. 2.8) se caracteriza por:

- La capa de entrada es un receptor para los datos de entrada.
- La capa intermedia realiza una transformación no lineal del espacio de entrada al espacio del nivel intermedio. Las neuronas del nivel intermedio son las funciones base para los vectores de entrada.
- La capa de salida calcula la ponderación lineal ponderada de las salidas de las neuronas de la capa intermedia.

La neurona i -ésima se activa si el patrón de entrada está próximo al centroide x_i [7].

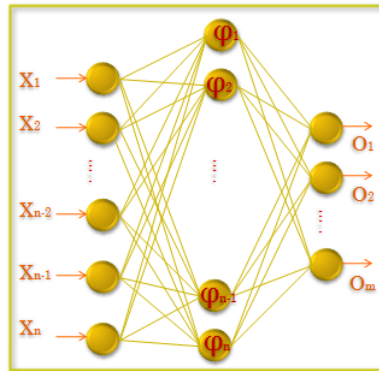


Figura 2.8: Red basada en una RBF

Maquinas de soporte vectorial (SVM)

Se basan en la teoría del aprendizaje estadístico (Vapnik y Chervonenkis). Son una implementación del método de minimización del riesgo estructural, la idea básica es que el error de generalización está acotado por la suma del error de entrenamiento y un término que depende de la dimensión VC. En el caso de patrones separables, una SVM fija el primer término y minimiza el segundo. En consecuencia una SVM puede proporcionar un alto desempeño en la generalización sobre problemas de clasificación de patrones[21].

En términos geométricos, el problema que resuelve una SVM es identificar una frontera de decisión entre dos clases, a través de un hiperplano que maximice el espacio de separación. El hiperplano está definido por vectores de soporte, que son los puntos que tocan el límite del margen.[24]

Clustering

Es el proceso de organizar objetos dentro de grupos cuyos miembros son similares de alguna manera (Fig. 2.9), algunos de los algoritmos de clustering son: k-medias, fuzzy c-medias, mapas auto-organizados, redes de cuatización vectorial[36].

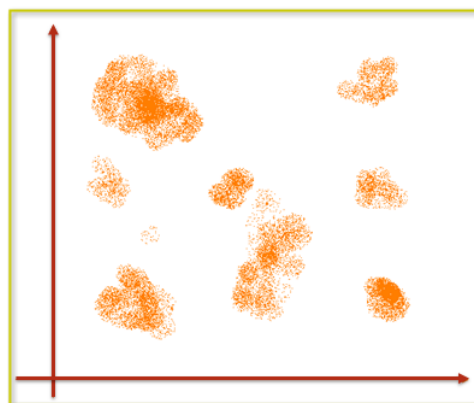


Figura 2.9: Agrupamiento

2.4.5. ART

Ante el famoso dilema de estabilidad - plasticidad se desarrolló la teoría de resonancia adaptativa que permitió generar las redes ART. El dilema aborda el problema de que un sistema de aprendizaje debe ser lo suficientemente adaptativo en respuesta a eventos significativos y a la vez permanezca estable ante situaciones no relevantes y en computación va enfocado a lograr que un sistema aprenda algo nuevo sin olvidar lo que ya se sabe.

La red neuronal ART es una clasificadora de vectores. Un vector de entrada se clasifica dentro de una de un número de categorías dependiendo de la similitud con los patrones previos. Si el criterio de similitud no se satisface se crea una nueva categoría. Así, la red ART resuelve el dilema entre plasticidad y estabilidad al habilitar que el aprendizaje ocurra solamente en un estado de resonancia.

La red neuronal ART1 trabaja con patrones binarios los cuales se comparan con los vectores prototipos de modo de disminuir similitudes hasta que una igualdad adecuada se encuentra o se crea una nueva categoría. Los patrones prototipos (vectores) se guardan en la red como vectores de peso de las conexiones de la Memoria de Periodo Largo Alti-Baja (en inglés T-D LTM).

Los vectores prototipos que no han sido usados hasta un cierto tiempo se colocan en 1. Estas T-D LTM dan la propiedad crucial del código de autoestabilización. Los nuevos patrones de entrada son también codificados en la Memoria de Periodo Largo de Abajo-Arriba (el filtro adaptativo B-U LTM).

Las B-U LTM juegan un papel en la primera etapa para determinar el grado de similitud entre el patrón de entrada y los prototipos, representados por los nodos de salida activos en la capa F2. El estado activo de una unidad de salida (la ganadora), denotado por j , indica que el vector de entrada x pertenece al racimo representado por esta unidad j .

Cabe destacar que el algoritmo de comparación entre un vector y otro es realizado sólo con norma de vectores. Se han estudiado diversas medidas de distancia no solo para las redes ART sino para muchas de las redes neuronales ofreciendo resultados para una aplicación específica lo que brinda la oportunidad de combinar los procesos markovianos con las redes de resonancia adaptativa[3].

2.4.6. ART2

Es una versión continua del modelo de resonancia adaptativa propuesta por Carpenter y Grossber en 1987. Esta red puede clasificar vectores de entrada reales (o escalas de grises), además de componentes binarios. Esta nueva red tiene la misma arquitectura que la original ART, pero en este caso los pesos de las conexiones hacia adelante y hacia atrás son iguales ($W_{ij}=T_{ij}$).

Se diferencia de la ART1 en la capacidad para tratar tramas analógicas. El precio de esta capacidad adicional es sobre todo un aumento de complejidad en el nivel de procesamiento de la capa de entrada, es decir que el nivel de la capa de entrada consta ahora de varios subniveles y varios sistemas de control de ganancia.

Algoritmo 2.2 Algoritmo ART2

Se presenta $E_k=(e^k_1, \dots, e^k_n)$ a la red.

Cada neurona de entrada recibe su valor del vector E_k y lo envía a las neuronas de salida.

Cada neurona compete con las demás de esta capa hasta que solo una permanece activa.

La neurona vencedora envía su salida a través de las conexiones hacia atrás.

La competencia de las neuronas consiste en determinar cual de los patrones de salida concuerda en mayor medida con el patron de entrada, se utiliza una función que determine la distancia entre un vector de entrada y los vectores que determinan a cada neurona de salida y el valor más pequeño es el que determina a la neurona ganadora[25].

Capítulo 3

Red Neuronal

Se sabe que los modelos neuronales artificiales están inspirados en el proceso de aprendizaje de un ser vivo y que cada modelo se diferencia de los demás por tomar diferentes modelos de aprendizaje y por el fenómeno en el que se basa ese modelo, por ejemplo la máquina de Boltzmann está basada en la temperatura de un sistema, las redes de Hopfield en la energía de los estados, las redes ART en la resonancia adaptativa, retropropagación en minimizar el error cuadrático, etc. También son muy variados los parámetros utilizados y las ventajas y desventajas que brinda cada uno de los modelos neuronales al atacar un problema específico. En particular para este problema se decidió utilizar las redes de resonancia adaptativa cuyas características se listan a continuación:

Autoorganización. Es una gran ventaja que el algoritmo de aprendizaje sea no supervisado ya que una vez que se han establecido los parámetros correctos en el entrenamiento no se requiere conocer *a priori* la clasificación resultante de un patrón, el algoritmo los encuentra y los agrupa de acuerdo al patrón que presentan y los resultados finales son muy gratos al problema particular.

Rapidez. El algoritmo aprende y clasifica rápidamente, el tiempo de ejecución del proceso de filtrado de un diario es en promedio de 4 a 5 minutos, procesando miles de archivos y ayudando a la optimización del proceso siguiente.

Atacan el dilema de elasticidad - plasticidad. Esta es la principal característica de este modelo, el dilema describe el problema originado al hacer clasificadores muy exactos en los cuales con frecuencia el aprendizaje de nuevos patrones es demasiado complicado, y cuando se busca que los nuevos patrones sean aprendidos el sistema no responde de la manera que se necesita en los patrones ya aprendidos. Las redes ART clasifican de manera muy aceptable los patrones conocidos y aprende con relativa facilidad los desconocidos sin olvidar los que ya conoce, logrando una estabilidad en el aprendizaje que ninguna otra red ofrece con tanta facilidad es por esto que fue el modelo elegido para solucionar el problema de reconocimiento de patrones que aborda esta tesis.

3.1. Arquitectura

Una de las desventajas de las redes basadas en la teoría de resonancia adaptativa es que la arquitectura final depende mucho del valor asignado al parámetro de vigilancia, sin embargo la arquitectura se limita a dos únicas capas, la de entrada y la de salida, con múltiples neuronas en cada una de ellas, en particular la estructura de la capa de entrada es un vector de neuronas cuyo tamaño máximo es igual a 53 y mínimo 14, esto quiere decir que

las entradas no siempre son del mismo tamaño, pero esto no afecta el funcionamiento general ya que las entradas sobrantes se toman como nulas inicializando el vector de entradas en cero, evitando de esta forma que se trabaje con valores anteriores que no corresponden a la entrada actual.

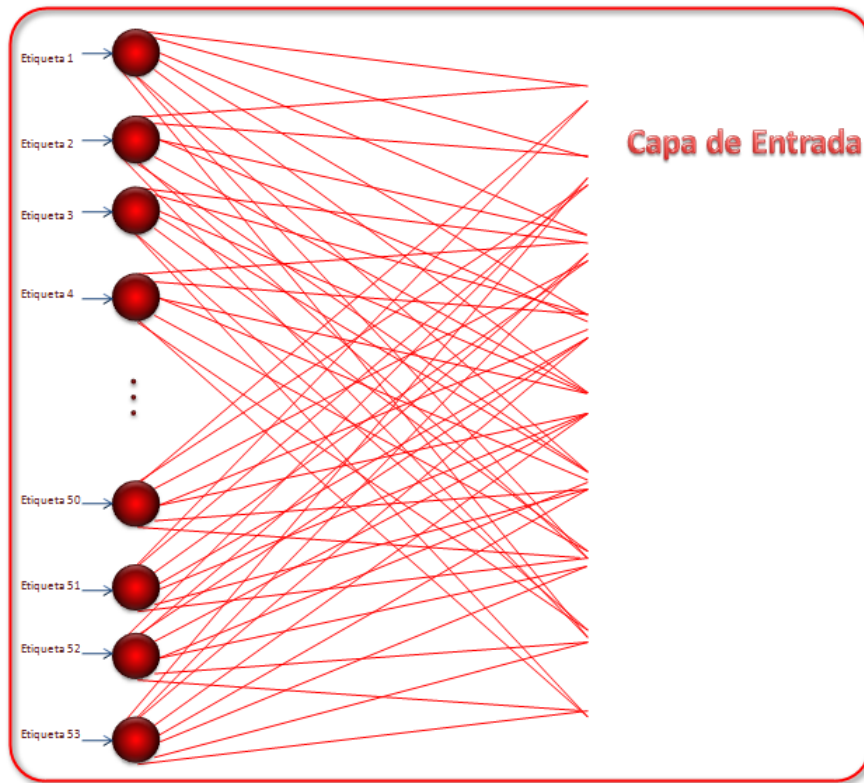


Figura 3.1: Capa de entrada

La capa de salida se inicializa con 8 neuronas(Fig. 3.1), pero a lo largo del proceso de aprendizaje se modifica el número de neuronas de acuerdo a algunos parámetros de la red neuronal, por ejemplo de acuerdo al parámetro de vigilancia (ρ).

3.2. Conexiones

La arquitectura de las conexiones no se modifica conforme se desarrolla el *proceso de entrenamiento* de manera que se conserva la relacion de $m \times n$ conexiones totales donde m son las neuronas de entrada y n son las neuronas de salida, pero sí se modifica el número de entradas y el número de salidas de manera que el número mínimo de conexiones en el proceso es de 84 y el número máximo es de 1590.

3.3. Establecimiento de constantes

Todos los procesos en redes neuronales requieren el establecimiento de constantes para un buen funcionamiento. Cuando la definición de estas constantes se realizan por otras

técnicas de inteligencia artificial como los algoritmos genéticos se requiere entonces establecer constantes para el genético de manera que la solución es más compleja mediante un proceso evolutivo que a través de la prueba de varios valores.

Parámetro de vigilancia (ρ)

Como no se tiene la experiencia suficiente para determinar el mejor valor de este parámetro se hicieron pruebas con diferentes valores y se compararon los resultados con los datos reales para seleccionar el que obtiene la mejor clasificación (Fig. 3.2). De acuerdo a este análisis el mejor valor del parámetro de vigilancia es cercano a 1 es decir, cuando se tiene un gran nivel de identificación de detalle, lo que indica que los archivos utilizados para que la red aprenda son muy similares entre sí, finalmente todos presentan elementos similares pero con un valor de $\rho = 0,95$ se lograron clasificar los archivos en las categorías correspondientes con un error de 2.4%.

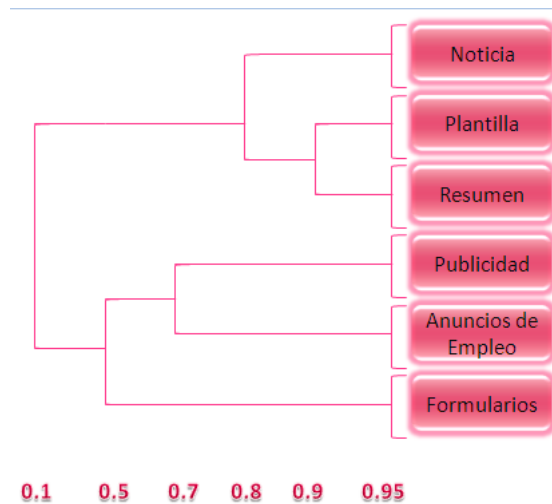


Figura 3.2: Clasificación de acuerdo al valor de ρ

Tasa de aprendizaje (β)

Los valores de β se pueden clasificar en 3 rangos, $\beta = 1, 0 < \beta < 1$ y $\beta = 0$, el aprendizaje de la red ART es más rápido conforme β se acerca a 1, si se define un valor de $\beta = 0$ la red no aprenderá los nuevos patrones, de manera que este parámetro establece la diferencia entre el proceso de entrenamiento y el proceso de clasificación, si se desea que la red clasifique y aprenda al mismo tiempo no se puede establecer un aprendizaje nulo ($\beta = 0$) o un aprendizaje máximo ($\beta = 1$). Con un aprendizaje nulo la red solo clasifica bien los patrones conocidos y es incapaz de detectar nuevo patrones, con un aprendizaje máximo se modifican constantemente los pesos W , es decir las conexiones entre las neuronas de entradas y las neuronas de salida lo que alenta el proceso de clasificación y hace ineficiente la ejecución. En particular para este caso se utilizó $\beta = 1$ para el proceso de entrenamiento y $\beta = 0$ para la clasificación, estableciendo así una marcada diferencia entre estos dos procesos y haciendo muy eficaz la clasificación en tiempo real.

Parámetro de selección (α)

El parámetro α se definió con un valor de 0.001, es un valor estándar de la literatura y sirve para que el valor de T_j no sea 1 lo cual quiere decir que los pesos W no se ven afectados por la entrada actual.

3.4. Entradas de la red

El M-PP (Módulo de preprocesamiento, descrito más adelante) regresa un vector numérico que representa el archivo html a clasificar mapeado en un espacio numérico de una dimensión, el vector no tiene un tamaño fijo puesto que depende de la estructura del diario y de la información que presenta pero la estructura general de una nota, que es el patrón que la red ART debe reconocer con mayor precisión es la siguiente:

<html>	→	1
<head>	→	3
<title>	→	8
</title>	→	8
</head>	→	3
<body>	→	2
<div>	→	12
<h1>	→	16
</h1>	→	16
</div>	→	12
<div>	→	12
<p>	→	5
	→	7
	→	7
</p>	→	5
</div>	→	12
<p>	→	5
</p>	→	5
</body>	→	2
</html>	→	1

Figura 3.3: Ejemplo de un vector de entrada

3.5. Activación de neuronas

La activación de una neurona es un evento que resulta de la competencia entre todas las neuronas de salida, una neurona activa es la que obtiene el máximo valor en la capa de salida sin embargo que una neurona se active no quiere decir que sea la categoría correcta para la entrada mostrada. La activación solo consiste en obtener la neurona que brinda el T_j máximo, siendo j la neurona de salida ($1 < j < n$), el valor de T_j es una función de las entradas, del parámetro de selección α y de los pesos W que van de las neuronas de entrada a cada neurona de salida.

3.6. Proceso de entrenamiento

El proceso de entrenamiento es la fase de la red neuronal en la que se aprenden los patrones deseados, se toma una muestra de los diarios que se desea clasificar se tienen 300 archivos por cada diario inicialmente se tienen 52 diarios de manera que se tienen en total 15600 patrones iniciales pero si se considera una sola red ART para clasificar los diarios se requiere un valor de ρ muy alto lo que causa que la arquitectura de la red sea excesivamente grande con demasiadas neuronas de salida y un procesamiento más lento, de manera que se realizó un entrenamiento para cada diario obteniendo diferentes valores de los pesos W y T almacenados en archivos de texto, este proceso arroja pesos diferentes para cada diario, la ventaja de este manejo además de mayor eficiencia y menor procesamiento computacional es que la red de cada diario es implementada en una máquina diferente logrando los resultados deseados en menor tiempo(Fig. 3.4).

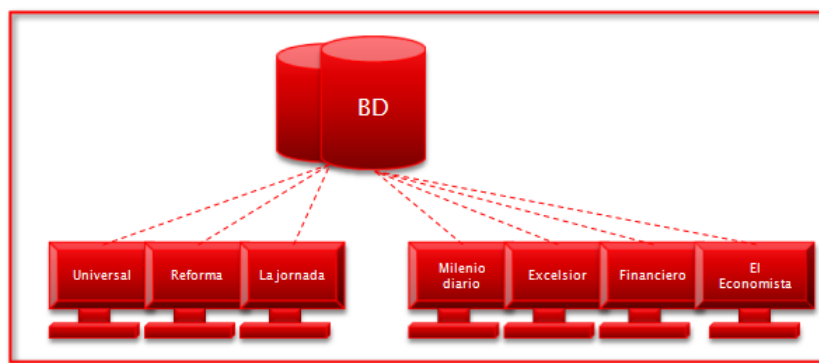


Figura 3.4: Paralelización del proceso

Capítulo 4

Desarrollo del sistema

El sistema consta de las siguientes fases:

- Análisis
- Diseño
- Codificación
- Implantación
- Pruebas
- Mantenimiento

éstas se desarrollan de acuerdo al modelo evolutivo de desarrollo de *software*.

Pero como ya se explicó este modelo es iterativo, las etapas anteriores se repitieron tres veces de acuerdo al siguiente diagrama:

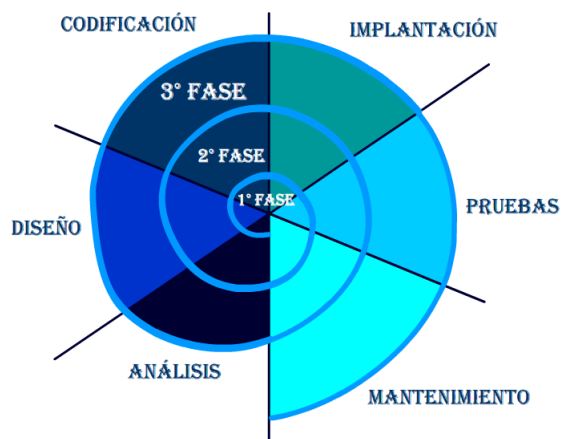


Figura 4.1: Ciclo de desarrollo del sistema

A continuación se describen las etapas mostradas en el gráfico.

4.1. Análisis

En esta etapa además de describir las características del *software* final se organizan los requisitos para facilitar la codificación, implantación y verificación en cada una de las fases de implementación de forma que la programación se facilite y esté planeada de la mejor manera posible. Uno de los métodos que establece mayor comunicación entre el desarrollador y el usuario final es el modelado a través de casos de uso.

Un modelo de casos de uso es una representación gráfica del sistema que contiene usuarios y acciones, además representa las relaciones entre estos a través de flechas, las acciones son una descripción de lo que hace el sistema, las acciones son representadas a través de elipses.

4.1.1. Requerimientos del sistema

Primera Fase

El *software* debe codificar los archivos *html* de los diarios de interés, estos solo pueden ser procesados localmente de manera que no es necesaria una interfaz *web* pero por razones de integridad se realizará también en *php*.

Una vez codificados deben procesarse con la red ART para filtrar las notas de interés de los archivos que no contienen noticias.

De los archivos filtrados se obtiene la información de interés. Se consulta en una *base de datos* las palabras clave de cada cliente, se hace una búsqueda en todos los archivos *html* y si la palabra es encontrada la nota se inserta en la tabla de la base de datos y se almacena la relación con el tema correspondiente, si la noticia no contiene ninguna mención de interés solo se almacena su contenido pero sin asociar la noticia con algún consultor o tema.

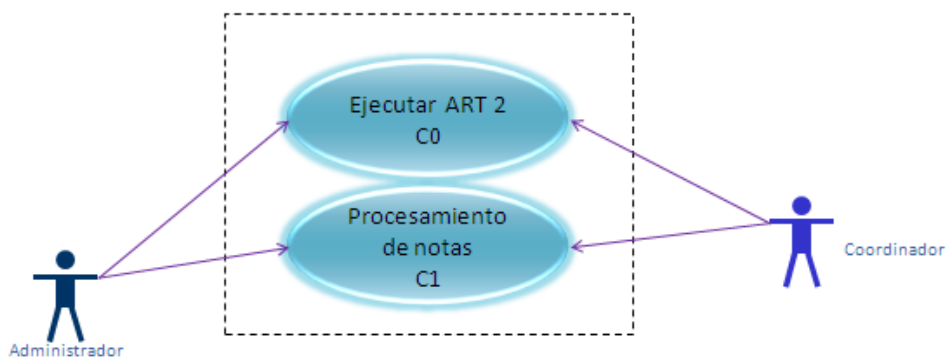


Figura 4.2: Casos de uso de la primera fase

Segunda Fase

Para la administración de los tópicos referentes al sistema y para dividir las funciones que realizarán se manejan 4 niveles de usuario: administrador, coordinador, analista y consultor. Cada uno es encargado de realizar una o más gestiones de acuerdo a la actividad que realizan.

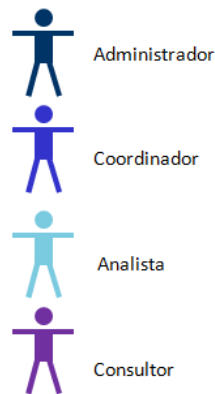


Figura 4.3: Niveles de usuario

Existen dos acciones principales en esta fase para concluir el desarrollo del sistema, el ingreso: para autenticar correctamente y la administración. De manera general el diagrama de subcasos de uso de la administración tiene 7 acciones básicas que además de proporcionar los módulos de programación base del sistema muestran sin ambigüedad el rol de cada nivel de usuario, cada una de las acciones descritas en este diagrama tiene a su vez subcasos que también son representados en esta sección.

Este diagrama no muestra los procesos básicos ni los métodos utilizados para implementar o mejorar los algoritmos, estos subprocesos se describirán más adelante.

El sistema está dividido en dos partes principales, la red ART 2 y el sistema de procesamiento, ambos requieren administración pero el primero sólo puede ser manipulado por el administrador y el segundo tienen niveles de usuario que permiten manipular el sistema por muchos usuarios con jerarquías diferentes.

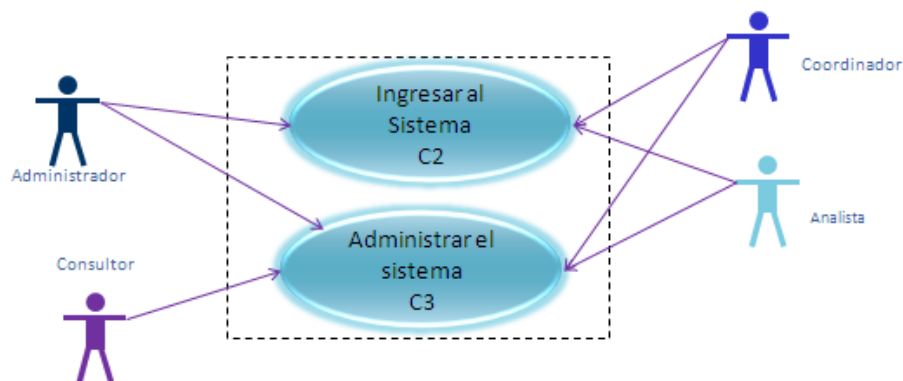


Figura 4.4: Casos de uso para la segunda fase

Para el caso 3 (C3) se tienen los siguientes subcasos de uso:

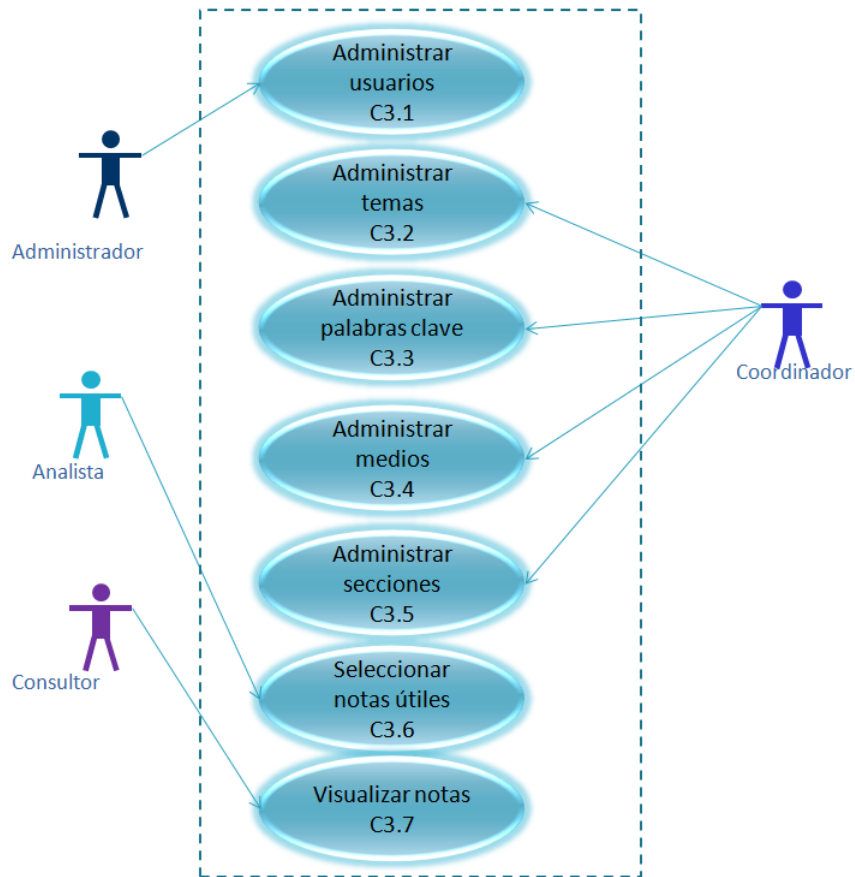


Figura 4.5: Subcasos de C3. Administrar el sistema

Cada uno de los casos anteriores puede describirse más detalladamente:

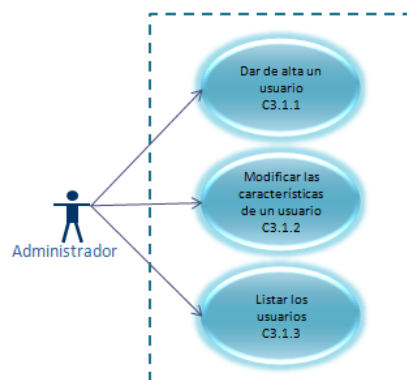


Figura 4.6: Subcasos de C3.1. Administrar usuarios

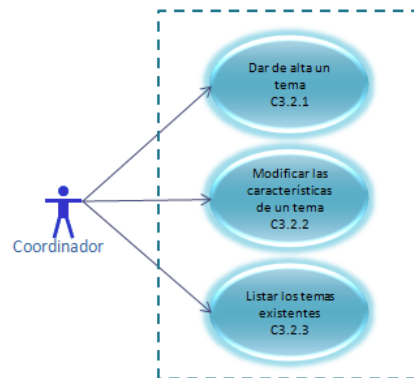


Figura 4.7: Subcasos de C3.2. Administrar temas

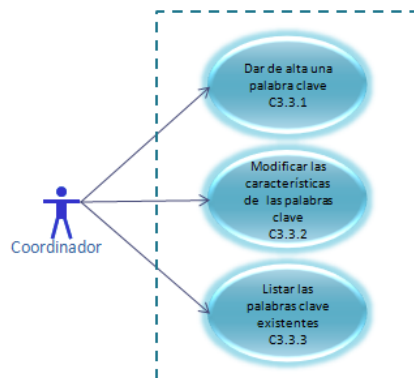


Figura 4.8: Subcasos de C3.3. Administrar palabras clave

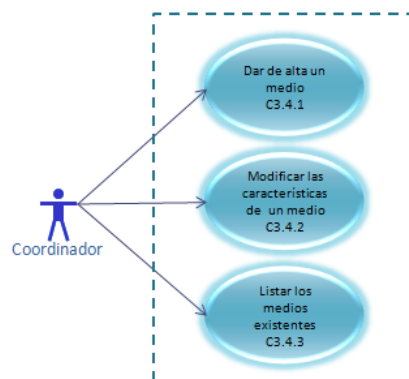


Figura 4.9: Subcasos de C3.4. Administrar medios

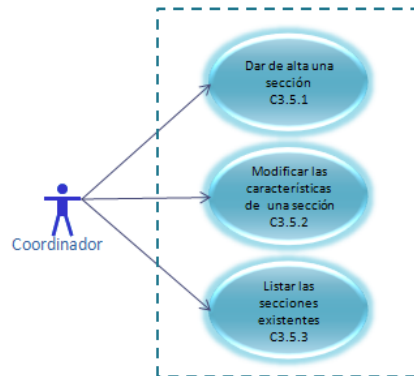


Figura 4.10: Subcasos de C3.5. Administrar secciones

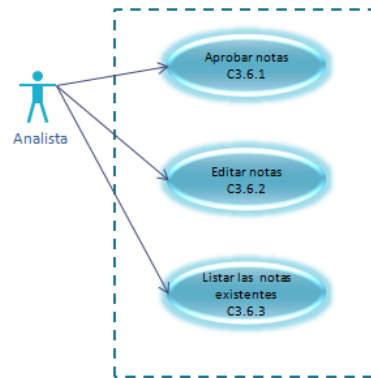


Figura 4.11: Subcasos de C3.6. Seleccionar notas útiles

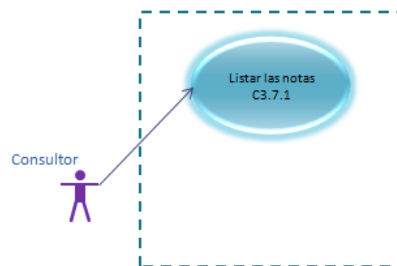


Figura 4.12: Subcasos de C3.7. Visualizar notas útiles

Tercera fase

Está enfocada a la presentación de las notas, las notas tienen uno o varios temas de interés, cada tema de interés estará asociado a un consultor, el consultor es el nombre global con el cual se identificará una página *web* y al usuario final, quien sólo observa las notas encontradas, de esta forma en una página *web* se listarán los temas asociados al consultor

y por cada tema se listarán las notas en las que se hayan encontrado las palabras clave de cada tema. Además se incorporan funciones que permiten buscar una nota por fecha, tema o contenido en toda la base de datos.



Figura 4.13: Casos de uso para la tercera fase

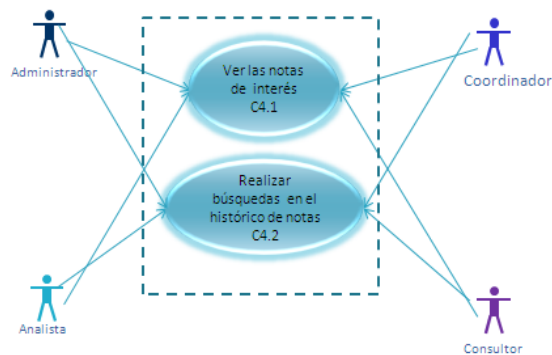


Figura 4.14: Subcasos de uso de C4. Visualizar las notas

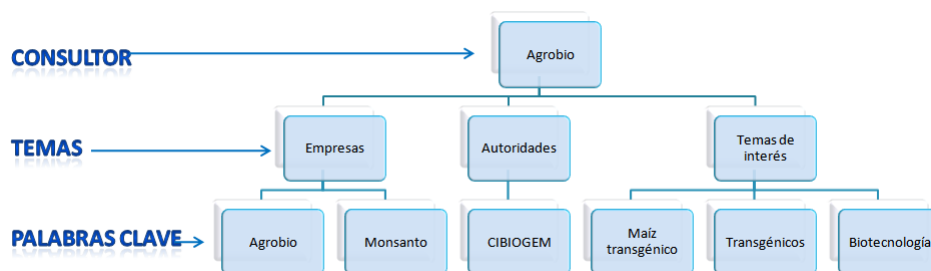


Figura 4.15: Ejemplo de la estructura web

4.1.2. Requerimientos de interfaces

Primera fase

No se requieren interfaces solo una dirección URL para ejecutar la aplicación.

Segunda Fase

Todas las interfaces deben ser intuitivas, y de apariencia corporativa, para la elaboración de estas debe contemplarse que la velocidad de la petición del cliente y la respuesta del servidor debe ser alta. La presentación de la información deberá permitir manipular el contenido de manera compatible con programas de uso común como hojas de cálculo ó procesadores de texto.

El sistema deberá permitir la autenticación del usuario.

Se deben tener interfaces de administración de los usuarios, medios, secciones y los temas de interés (con sus palabras claves).

Se debe tener una interfaz de consulta de las notas con mención de temas de interés y sin mención de estos, además de un sistema de búsqueda por tema, fecha o palabra clave.

Tercera Fase

Se debe contar con una interfaz *web* con los colores institucionales del consultor, una imagen de cabecera con el logotipo del consultor y un listado de los temas de interés con las notas que se tienen de cada tema.

The screenshot shows the AgroBIO México website interface. At the top left is the logo for AgroBIO México. The main header reads 'Informe de Cobertura' and 'Miércoles 17 de Febrero de 2010'. Below this is a navigation bar with five categories: 'AgroBIO y sus empresas' (1), 'Biotecnología y Cultivos transgénicos' (3), 'ONG'S y Asociaciones' (3), 'Autoridades' (11), and 'Sector Agropecuario' (5). The main content area is divided into two sections. The first section is titled 'Titulares del Día' and lists several news items from various sources: 'REFORMA: Arrastra Conagua obras pendientes', 'EL UNIVERSAL: Tribunal agrario: 30% del campo, de narcos', 'EL FINANCIERO: Últimátum económico de Europa a Grecia', 'EL ECONOMISTA: Se amparan 140 firmas contra consolidación', 'LA JORNADA: PRI: hubo pacto con Gómez Mont, pero fue coyuntural', 'MILENIO: El Presidente da espaldarazo a Gómez Mont', and 'LA CRÓNICA: Rojas confirma pacto en lo oscuro con Gómez Mont'. The second section is titled 'Biotecnología y Cultivos transgénicos' and also has a 'Notas' sub-header. On the right side of the page, there is a vertical navigation menu with buttons for 'Histórico a 30 días', 'AgroBIO y sus empresas', 'Biotecnología y Cultivos transgénicos', 'Búsqueda', and 'Generar Informe'.

Figura 4.16: Ejemplo de interfaz *web* para un consultor

4.2. Diseño

Las descripciones y diagramas tanto de los bosquejos de las interfaces gráficas como de las relaciones de las tablas de la base de datos, algoritmos y los módulos del sistema facilitan el proceso de codificación, además se detectan fallas en el listado de los requisitos y se presentan esquemas que facilitan al usuario final la visualización de sus peticiones para evitar que se malinterpreten las necesidades funcionales que cubrirá el sistema informático.

4.2.1. Diseño Gráfico

Primera fase

No se requieren interfaces.

Segunda fase

La interfaz debe ser en su mayoría de colores claros y muy ligera, no debe contener archivos en formato *swf* y no se debe abusar de las imágenes y animaciones. La apariencia de la página *web* debe ser independiente del contenido, es necesario utilizar hojas de estilo y Ajax para disminuir el número de consultas al servidor que almacena la base de datos.



Figura 4.17: Arquitectura de la interfaz base

Tercera fase

Se requiere una interfaz muy sencilla y de fácil consulta para cada tema de interés, además los colores deben ser independientes del diseño de la página ya que estos variarán a gusto de cada consultor.

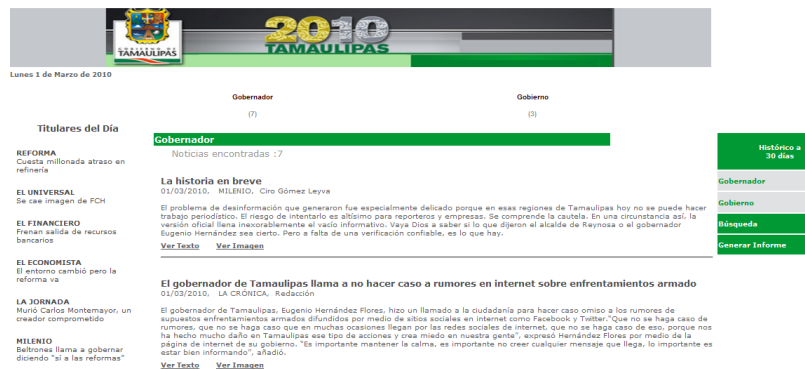


Figura 4.18: Páginas de consulta de notas

4.2.2. Diseño modular

Un diseño modular permitirá realizar un cronograma de trabajo dividido por áreas, dividiendo también la asignación de recursos materiales, financieros y humanos. Con el

diseño modular también será más fácil la detección de fallas, problemas o cambios en los diseños así mismo la solución de éstos y sobre todo en las ocasiones en que serán cambios en el diseño, se afectará lo menos posible a los demás módulos. La relativa independencia de cada subsistema con los demás hace más fácil el rediseño o la disposición de varios diseños y poder elegir un diseño completo o simplemente tomar partes de cada opción e integrarlos perfectamente.[40]

Primera fase

4.2.2.1. Módulo PP (M-PP)

En este módulo se hará un preprocesamiento al archivo html para obtener una codificación con la cual se representa a cada nota como un vector n dimensional, donde n es el número de componentes del vector, en la teoría de resonancia adaptativa es posible que los vectores de entrada tengan tamaños diferentes según las características de cada señal de entrada por lo que no es necesario estandarizar el tamaño de todos los vectores iniciales.

4.2.2.2. Módulo de Red Neuronal(M-RN)

Se implementará una red neuronal con el fin de descartar aquellos archivos html que no tengan la estructura de una noticia ya que la herramienta inicial descarga todos los archivos del sitio *web*: publicidad, plantillas, anuncios de empleo, resúmenes por secciones, etc. a manera de agilizar la fase del procesamiento de las notas reales y no perder tiempo de procesador en analizar un archivo que no es una noticia.

1. Elección del tipo de red.
2. Definición de arquitectura y parámetros
3. Captura de datos de entrenamiento
4. Fase de Inicialización
5. Fase de aprendizaje ó entrenamiento
6. Pruebas y análisis
7. Ejecución

Comúnmente las redes ART funcionan como un mapa autoorganizado permitiendo agrupar características de los vectores que no son percibidas fácilmente y además la fase de ejecución y de entrenamiento de la red neuronal se fusionan por lo cual se inicializa la red con pesos estandarizados y mientras se clasifican los vectores se van aprendiendo las nuevas características.

El proceso de implementación inicia definiendo 6 neuronas de salida por que son el número de clasificaciones ya identificadas estas corresponden a :

- noticia
- resumen por sección
- publicidad
- formularios de opinion

- plantillas
- anuncios de empleo

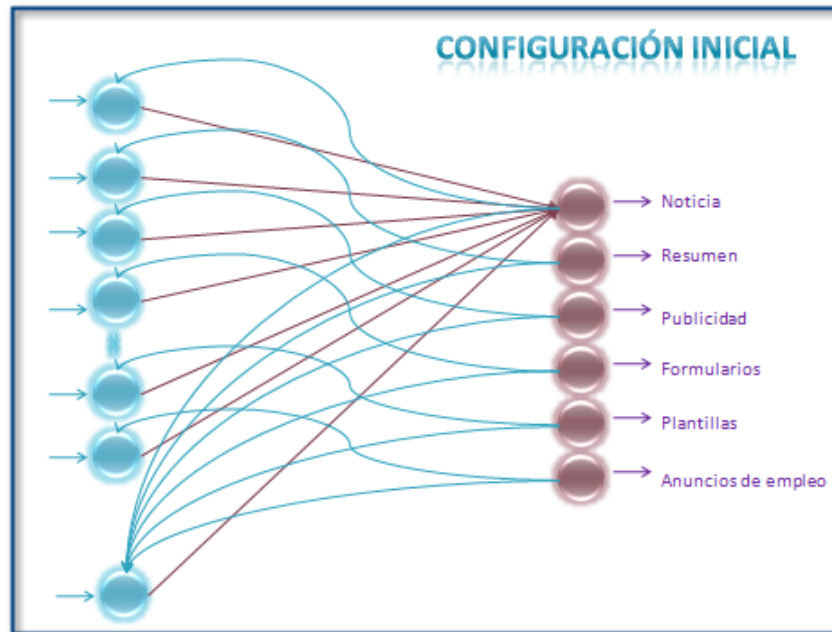


Figura 4.19: Arquitectura inicial de la red ART

4.2.2.3. Módulo de Proceso(M-Proceso)

Se encarga de evaluar solo los archivos html que pasaron el filtro de la red ART, en este módulo se elimina el concepto del archivo html y se captura en la base de datos la nota relacionanda con el tema de interés que contiene. Cada archivo es leído palabra a palabra detectando etiquetas clave para cada tipo de diario, localizando entre los marcadores de html las características de la nota tales como título, contenido, agencia de emision (EFE, NOTIMEX, AG) y en algunos casos la sección.

Cada vez que el patrón de marcadores html se detecta en el archivo se captura la información del campo necesaria correspondiente a este patrón. Un patrón esta compuesto por una o más etiquetas de inicio y una o más etiquetas de cierre. La identificación de los patrones fue un proceso de análisis manual que es almacenado en un *case* según el identificador de cada diario, de esta manera se tienen 4 patrones por diario: un patrón para el título, uno para el cuerpo de la nota, uno para el autor o agencia y uno para la sección por cada diario contemplado.

En el apéndice A se muestra la estructura de los patrones utilizados actualmente.

Algoritmo 4.1 Procesamiento de los archivos

```

foreach(diariosEnLinea)
    etiquetas=obtenerEtiquetas(diario)
    foreach(diariosEnLinea)
        resultado=leeArchivo(etiquetas)
        titulo=resultado[0]
        seccion=resultado[1]
        nota=resultado[2]
        autor=resultado[3]
        link=resultado[4]
        escribirNotaGeneral(titulo,nota,autor,link)
        obtenerActor(resultado)
    end
end

```

Segunda fase

4.2.2.4. Módulo de Administración (M-Administración).

El módulo de Administración permite agregar, modificar y listar temas, secciones, palabras clave y usuarios de manera que el código utilizado para la gestión de estos tópicos es el mismo. Se diseñó una función que a través del paso de parámetros por referencia permite ingresar a las bases de datos correspondientes y solicitar al usuario la información necesaria para agregarla a la tabla de la solicitud. Lo mismo para modificar los campos o para listarlos, la estructura de la función se basa en generalizar la petición de alta, actualización o selección de las tablas indicadas en los parámetros.

Algoritmo 4.2 Inserción de campos de interés en la BD

```

function formularioInsertar(tabla)
    camposTabla=leerTabla
    imprime(form)
    while(numCampos<camposTabla)
        imprime(Input Text)
    end while
    imprime(submit)
    imprime(endForm)
end function

function Insertar(tabla,campos,valores)
    conexionInserta()
    sql=insert into tabla (campos) values (valores);
end function

```

Y de manera análoga se realiza con la actualización y el listado.

4.2.2.5. Módulo de Diseño de la Base de Datos(M-BD).

Se presenta el diseño de la base de datos. Los requerimientos para el diseño de la base son los siguientes:

Los actores son los temas de interés buscados en las notas, cada actor tiene una o más palabras clave que ayudan en la búsqueda.

Las notas sin mención deben tener asociado un diario de emisión, una fecha de emisión y una de inserción, un título, el cuerpo de la nota, el link, el autor y la sección. Las notas con mención deben tener además información de los actores que aparecen en la nota.

El medio es el nombre del diario que emite una nota. Además cada medio posee una o más secciones.

Cada persona que desee ingresar al sistema se autenticará a través de un usuario y contraseña, pero además deberá almacenarse el nombre completo de este al momento del registro inicial.

De manera que de estos requisitos se obtiene el diagrama de base de datos normalizado.

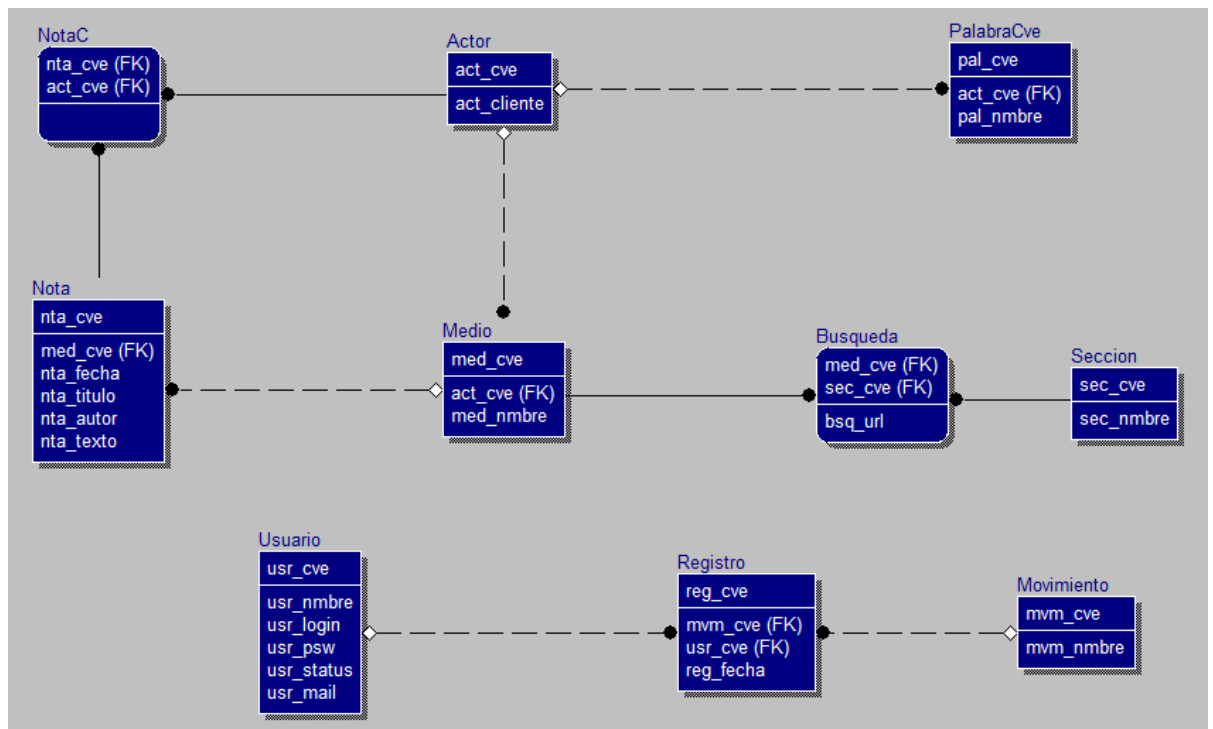


Figura 4.20: Modelo de la Base de datos

Tercera fase

4.2.2.6. Módulo de Presentación (M-Presentación).

En este módulo se debe desarrollar la interfaz que permite la consulta de la información de interés almacenada en la base de datos por parte del consultor, analista, coordinador o administrador para su manipulación en otros programas de paquetería básica.

Se presenta una interfaz de consulta de notas útiles, es decir, aquellas notas que tienen una mención de las palabras clave de algún tema de interés. Se tiene una interfaz para ver las notas del día que no contienen una mención pero que son capturadas con el afán de formar un histórico y almacenar información que actualmente no es requerida pero puede utilizarse en un futuro.

4.2.3. Algoritmos

Primera Fase

Para lograr cada una de las acciones descritas en los requerimientos se necesitan algoritmos de búsqueda y de aprendizaje supervisado y no supervisado.

Para el aprendizaje de los patrones se utilizó el algoritmo de la red de resonancia adaptativa. Se basa en la arquitectura de dos capas, solo entrada y salida, se tienen pesos de conexión entre cada neurona de entrada y las neuronas de salida estos pesos se almacenan en el vector W , también se tienen los pesos almacenados en T que son los pesos que van de la capa de salida a la de entrada, los pesos en un principio son iguales pero los almacenados en T están normalizados. El vector de entrada es almacenado en X .

Algoritmo 4.3 Pseudocódigo de la red ART

```

foreach(lecturasDeEntrenamiento)
  inicializar(parametros)
   $I[0..|W|]=X$ 
   $I[|W|.2*|W|]=1-I[0..|W|]$ 
  for j=0..numCategorias
    if categoriaDisponible(j)
      minimo=minimo(I,W[j])
       $T=norma(minimo)/(\alpha+norma(W))$ 
    enf if
  end for
  Jganadora=max(T)
  resonancia=verificarResonancia(Jganadora,X)
  if resonancia  $\geq \rho$ 
    categoriaG=Jganadora;
     $W=\beta * min(W, I) + (1 - \beta) * W$ 
  else
    restringir(Jganadora)
    if categoriasDisponibles==NULL
      creaNuevaCategoría()
    end if
  end if
end

```

El algoritmo de búsqueda empleado para encontrar una palabra en una cadena de texto es simple, se desea evitar en la medida de lo posible, leer todo el archivo demasiadas veces, de manera que se asigna cada palabra del título y el cuerpo de la nota a un arreglo *hash*, permitiendo que para buscar una palabra solo se pregunte una vez por la palabra clave buscada, este proceso se repite C x P , donde C es el número de temas de interés y P el número de palabras clave de cada tema.

Algoritmo 4.4 Búsqueda de temas en cada nota

```

foreach(Temas)
  foreach(PalabrasClave)
    if(vector[PalabraClave]==1)
      mencion=true
      actores=actores+idTema
    else
      mencion=false
    end
  end
end
end

```

4.2.3.1. Complejidad

La complejidad del algoritmo de aprendizaje de la red de resonancia adaptativa es $n(n-2)$, en donde n es el número de categorías de la neurona de salida, ya que en el peor de los casos se deberán revisar todas las categorías no restringidas sin encontrar una clasificación adecuada, la primera vez se revisarán n categorías, la segunda $n-1$, la tercera $n-2$ y así sucesivamente.

La complejidad de la búsqueda es $n \times m$, donde n es el número de temas de interés y m es el número de palabras clave por cada actor, la búsqueda de estas palabras en el archivo es lineal, ya que solo se recorre una vez el vector de palabras que forman el archivo.

En conclusión el proceso es computacionalmente complejo pero siempre en tiempo polinomial, por lo que el problema es perfectamente computable y con un tiempo de respuesta razonable para la situación en cuestión y se soluciona con un tiempo menor al necesario cuando se realiza el proceso de manera manual, por esta razón la implementación es viable.

Una de las ventajas del cálculo de la complejidad es que teniendo la expresión de complejidad de un algoritmo se puede evaluar el tiempo de ejecución variando n de manera sencilla, en las máquinas de la actualidad se tienen cuadros en los que se evalúan diferentes expresiones de complejidad con diferentes valores de n , en la figura siguiente se presenta el caso cuando la computadora es capaz de realizar 10^9 pasos de programa por segundo.

Valor de n	1	8	32	10^3	10^6	10^9
$\log_2 n$	<1 ns	3 ns	5 ns	10 ns	20 ns	30 ns
n	1 ns	8 ns	32 ns	1 μs	1 ms	1 s
$n \log_2 n$	<1 ns	24 ns	160 ns	10 μs	20 ms	30 s
n^2	1 ns	64 ns	1 μs	1 ms	17 min	38 años
n^3	1 ns	512 ns	33 μs	1 s	38 años	inabordable
2^n	2 ns	256 ns	4.3 s	inabordable	inabordable	inabordable
$n!$	1 ns	40 μs	inabordable	inabordable	inabordable	inabordable
n^n	1 ns	17 ms	inabordable	inabordable	inabordable	inabordable

Figura 4.21: Complejidad en función de n **Segunda Fase**

No requiere algoritmos complejos.

Tercera Fase

No requiere algoritmos complejos.

4.2.4. Diagramas de flujo(Ordinogramas)

Primera Fase

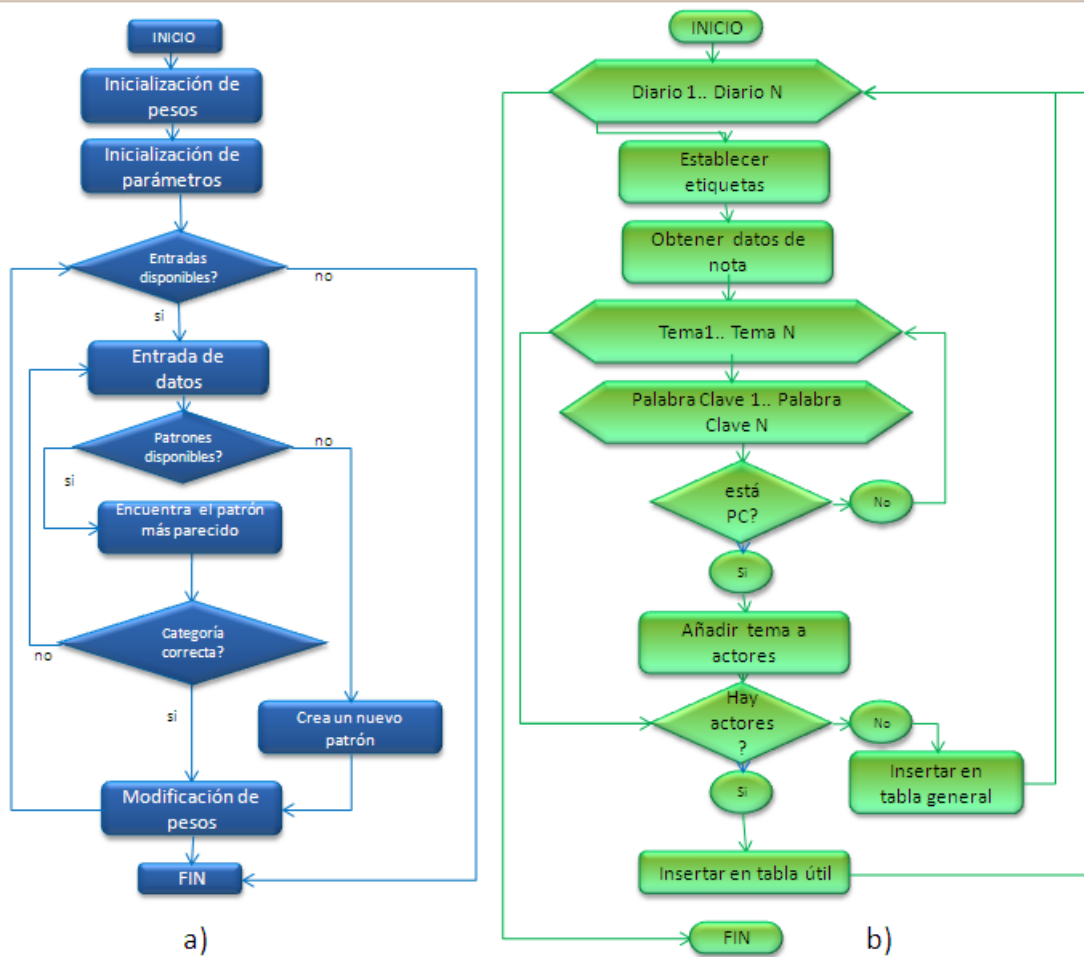


Figura 4.22: a) Funcionamiento de la red ART

b) Búsqueda de actores (temas)

Segunda Fase

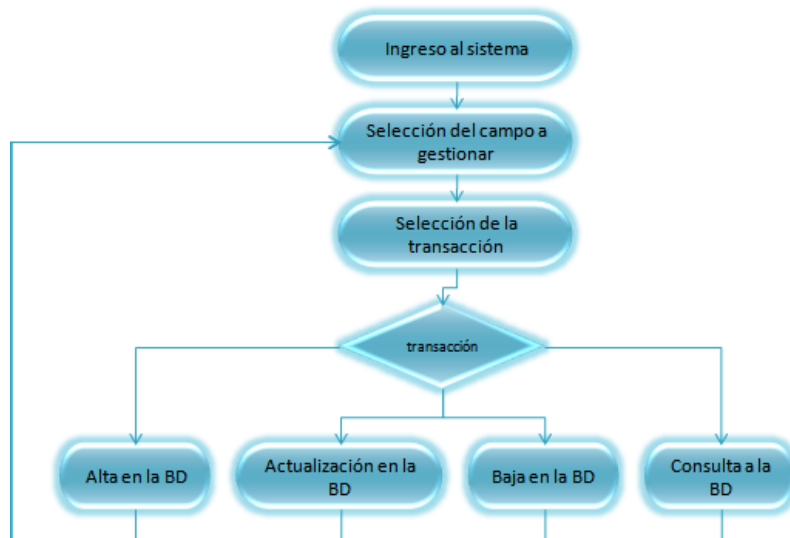


Figura 4.23: Administración del sistema

Tercera Fase

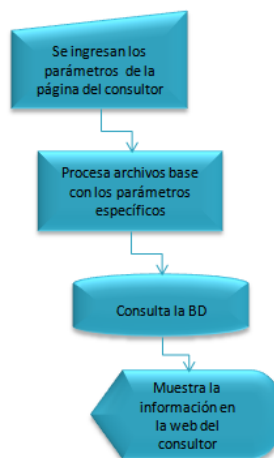


Figura 4.24: Presentación de la información

4.2.5. Diagrama de bloques

Primera Fase

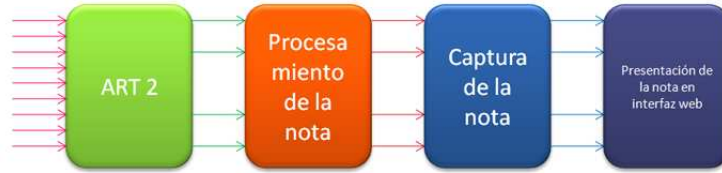


Figura 4.25: Proceso primera fase

Segunda Fase



Figura 4.26: Proceso segunda fase

Tercera Fase

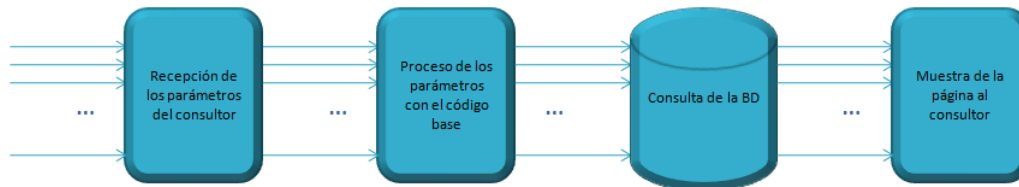


Figura 4.27: Proceso tercera fase

4.3. Codificación del sistema

Primera fase

En la primera fase fue necesario realizar los algoritmos que procesan los archivos y obtienen los contenidos de las notas, esto consta de tres programas:

- Codificación de los archivos html
- Filtro por red ART
- Procesamiento de las notas

La codificación está basada en las etiquetas html de la nota digital, el proceso de codificar inicia leyendo el archivo html y asignando un número predeterminado por cada etiqueta encontrada. Se omite el contenido de las etiquetas y se representa al archivo html mediante su estructura básica.

Etiqueta		Código
<html>	</html>	1
<body>	</body>	2
<head>	</head>	3
 	</br>	4
<p>	</p>	5
<a>		6
		7
<title>	</title>	8
<table>	</table>	9
<td>	</td>	10
<tr>	</tr>	11
<div>	</div>	12
<form>	</form>	13
<input>	</input>	14
		15
<h1>	</h1>	16
<h2>	</h2>	17
<h3>	</h3>	18

Figura 4.28: Codificación de las etiquetas html

Algoritmo 4.5 Algoritmo de codificación de los archivos html

```

numArchivo=0
while(existanArchivos)
  numComponente=0
  archivo=fopen(«file.html»,r)
  numArchivo++
  do{
    palabra=leer(archivo)
    nuevaPalabra=codifica(palabra)
    vector[numArchivo][numComponente]=nuevaPalabra
    numComponente++
  }while (existanPalabras)
  close(archivo)
end while

```

El programa que filtra los archivos html para procesar solo las notas procesa un vector con una red basada en la Teoría de Resonancia Adaptativa y debe dar como respuesta la clasificación del vector en cualquiera de los 6 grupos enumerados en el diseño del módulo de red neuronal. Si el vector no corresponde a ninguna de estas clasificaciones iniciales la red crea una nueva clasificación en la cual pueden entrar nuevos vectores tal como en las clasificaciones iniciales, cada vez que se crea una nueva clasificación se debe crear una nueva

neurona de salida. La principal desventaja de este método es que el número de neuronas de salida crece rápidamente, pero la clasificación es bastante eficiente.

Los componentes del algoritmo son:

Vector de entrada: Cada uno de los vectores de entrada \mathbf{V} , es un vector M -dimensional donde cada una de sus componentes tiene coordenadas incluidas en el intervalo $[0,1]$.

Codificación complementaria: A partir del vector de entrada \mathbf{V} , se crea un nuevo vector normalizado \mathbf{I} de dimensión $2M$ en el que la componente $I_{J+M} = 1 - I_J$.

Vector de pesos del nodo de salida j (categoría j): W_j . Inicialmente, $W_{J_1} = W_{J_2} = W_{J_{2M}} = 1$.

Velocidad de aprendizaje: ("learning rate"), β entre $[0, 1]$. Aprendizaje rápido, $\beta=1$; Aprendizaje lento, $\beta \ll 1$; Sin aprendizaje, $\beta=0$.

Parámetro de vigilancia: ρ entre $[0, 1]$. ρ cercano a cero implica menos categorías al agrupar con criterios de semejanza poco exigentes, ρ cercano a uno implica muchas clases, cada una con pocos miembros pero muy parecidos entre sí.

Parámetro de selección: $\alpha > 0$. Debe ser muy cercano a cero. Sirve para deshacer igualdades. Un valor típico es 0.001.

Selección de categoría: Para cada vector de entrada \mathbf{V} y cada categoría j se calcula la función de selección o semejanza $T_J(\mathbf{V})$ como indica la siguiente ecuación:

$$T_J = \frac{|I \wedge W_j|}{\alpha + |W_j|}$$

$$A \wedge B = \min(A, B)$$

Y la norma $\| \cdot \|$ se define como:

$$\| I \| = \sum_{i=1}^{2M} I_i$$

Las conexiones mostradas en la imagen de la estructura de la red ART2 están incompletas debido al gran número de estas que realmente se establecen, la red neuronal tiene $M \times J$ conexiones, cada una de las neuronas de entrada está conectada con todas las neuronas de salida. Se realiza la operación necesaria para obtener los T_J de cada neurona de salida y se escoge la categoría j para la que $T_J(\mathbf{V})$ es máximo.

Pero se puede dar el caso en el que la categoría j correspondiente al $T_J(\mathbf{V})$ máximo no es parecida al vector de entrada, para evitar esto, antes de asignar \mathbf{V} en la categoría j se debe verificar que estén en resonancia, es decir que el vector de la categoría y el vector de entrada tengan una diferencia pequeña en la medida que el usuario lo determine según el valor de ρ , que es el parámetro de vigilancia, esto se calcula a través de la siguiente fórmula:

$$\frac{|I \wedge W_j|}{I} \geq \rho$$

Cuando la condición anterior se cumple se activa el nodo de salida j indicando que el vector de entrada \mathbf{V} fue clasificado por la red en la categoría j .

Cuando no se cumple con la restricción del parámetro de vigilancia el proceso se repite con las neuronas de salida restantes y se prueba el criterio ρ .

Si ninguna neurona de salida pasa el criterio de vigilancia se crea una nueva neurona de salida que corresponda con la entrada \mathbf{V} .

Para este proyecto en particular se realizó una red ART2 cuya ejecución es independiente del entrenamiento o aprendizaje, de manera que se selecciona un grupo de noticias ejemplo para la fase de aprendizaje que servirán como entrenamiento a la red, en esta fase se aceptan todas las nuevas categorías generadas por el arreglo ART, pero una vez que se ha establecido la arquitectura necesaria en la ejecución se bloquea el aprendizaje a manera de que desechen los vectores que no cumplen con los patrones ya aprendidos.

Para que la red aprenda o sólo clasifique se utilizan los valores 1 ó 0 del parámetro β respectivamente.

Segunda fase

Debido a que existen procesos restringidos no se pueden implementar en una misma interfaz gráfica presentados en un menú único, se manejan tres interfaces para la administración del sistema. Para las tres interfaces de administración se usará la misma plantilla de estilos de manera que la apariencia del sistema de administración estará regida por los mismos colores y estructuras por lo cual se implementó el archivo `screen.css`, en él se especifican los colores y comportamientos de la apariencia del sistema a través del lenguaje CSS (*Cascading Style Sheets*). Este lenguaje es usado para definir la presentación de un documento estructurado escrito en HTML o XML y permite tener un mayor control del documento organizando en un documento el contenido de una página *web* y en otro la apariencia.

Además de los CSS se requiere el uso de funciones para lograr una buena organización en el sistema, éstas ayudan a evitar códigos excesivos y facilitan el mantenimiento del sistema, el escalamiento y la corrección de errores.

El proceso requiere distintas interfaces para mostrar adecuadamente las opciones que tiene cada usuario, de manera que el primer contacto con el sistema es una pantalla de autenticación, en este programa se identifica el nivel del usuario mediante los datos que pide la interfaz: nombre de usuario y contraseña. Estos datos son comparados con los que se obtienen al hacer una consulta a la tabla **usuario** en la base de datos del sistema y si existe concordancia con algún registro de la base se obtiene el nivel del usuario que es otro campo de esa tabla, si no existe coincidencia de los datos que ingresó el usuario con los datos de la base se muestra un *alert* de *javascript* que indica que hay un error de autenticación y regresa a la pantalla inicial.



Figura 4.29: Autenticación

El proceso de autenticación da como resultado el nivel de usuario, este es un número entero que permite que el sistema presente los procesos que el visitante puede realizar de acuerdo al siguiente diagrama:

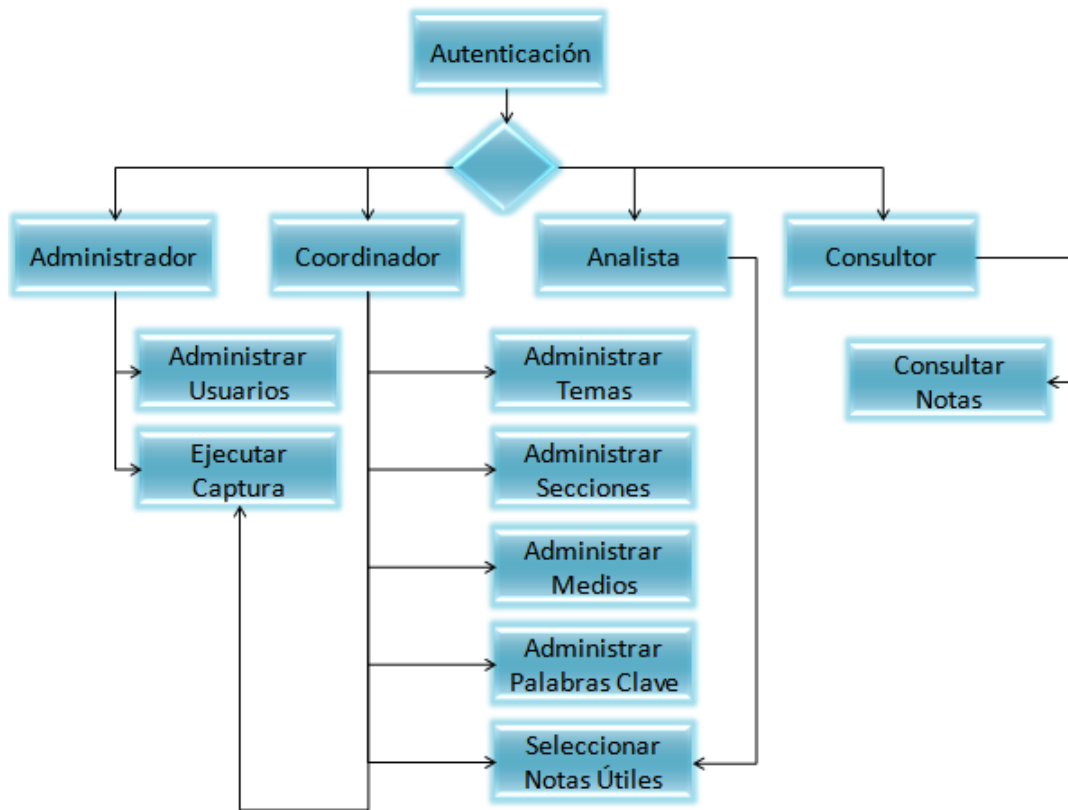


Figura 4.30: Funciones por tipo de usuario

De manera que el **administrador** del sistema puede ejecutar el proceso de clasificación de notas y administrar los usuarios, la administración, como ya se vio, se divide en 3 procesos alta, modificaciones y listado, recordando que el proceso de modificaciones involucra las bajas.

La clasificación de notas involucra consultas en las tablas siguientes: **actor**, **busqueda**, **medio**, **palabracve** y **seccion**, y la inserción de las notas en las tablas: **nota** y **notac**.

El **coordinador** también puede ejecutar el proceso de captura de notas, además es el usuario que administra más campos de manera que puede dar de alta, modificar y listar los temas, las palabras clave, los medios electrónicos y las secciones, por esta razón tiene el control (inserción y actualización) sobre las tablas **medio**, **actor**, **palabracve** y **seccion**, además puede ejecutar el proceso de clasificación de notas, es decir ingresa a un listado de las notas capturadas para cada tema y decide si la nota mostrada se presenta al usuario final, para la ejecución de este proceso se inserta en la tabla **nota**, debido a que el analista es el principal encargado de este proceso, la implementación se detallará en la descripción del rol siguiente.

El **analista** está dedicado al proceso de validación de las notas por lo que no puede realizar operaciones bajo ninguna tabla de administración, únicamente valida las notas que se muestran del tema seleccionado. El analista entra a la página correspondiente al tema de su interés, por ejemplo, el PRD. Una vez que se ingresa a la página de interés se muestran las notas encontradas por la herramienta y el analista seleccionará mediante un *check box*



Figura 4.31: Implementación de la interfaz

las notas que desea mostrar al usuario final.

Tercera fase

Es necesaria una cuarta interfaz que requiere mayor estética pues es la que visualizará el usuario final (consultor) quien es ajeno al trabajo de selección de notas, de administración de la herramienta y de la programación y desarrollo.

El **consultor** entra a la página *web* de su interés y puede observar los encabezados de las notas relacionadas con el tema de su interés, de manera que la única operación que realiza es una consulta a la tabla *nota* de acuerdo a un filtro por tema.

Figura 4.32: Ejemplo de página *web* para consulta de información

4.4. Implantación

Primera fase

Debido a que el servidor ya cuenta con PHP y con MySQL solo se necesitó crear las siguientes tablas en el servidor:

- actor
- busqueda
- medio
- movimiento
- nota
- notac
- palabracve
- registro
- seccion
- tipo
- usuario

Por otro lado se destinaron 6 máquinas para la ejecución del proceso de revisión de diarios web, por lo que el programa de procesamiento fue alojado en cada una de estas máquinas, además se determinó una división entre los diarios en línea para dividir el total de diarios entre las seis máquinas y equilibrar la carga de trabajo. La división fue la siguiente:

Máquina 1:

- El Universal
- Macroeconomía
- Bravo
- Vanguardia
- El Mañana (NL)
- El Informador
- Correo de Hoy
- El Orbe de Tapachula
- El Sol de Chilpancingo
- Diario Olmeca
- Liberal del Sur

Máquina 2:

- La Jornada
- Uno más uno
- Cambio
- Palabra
- El Mañana (Reynosa)
- Mural
- Noticias de Querétaro
- Diario de Chiapas
- Diario de Yucatán
- Imagen de Veracruz
- Síntesis de Puebla

Máquina 3:

- Milenio Diario
- Crain's
- Excelsior
- Expresso
- Herald
- El Bravo
- Diario
- Hidrocálido
- Tiempo de Oaxaca
- Novedades de Quintana Roo
- Dictamen
- Unión de Morelos

Máquina 4:	Máquina 5:	Máquina 6:
▪ El Economista	▪ El Financiero	▪ La Crónica
▪ Líderes Mexicanos	▪ Expansión	▪ Excelsior
▪ Noroeste	▪ Debate	▪ El Sol de México
▪ El Imparcial	▪ Zócalo	▪ El Diario de
▪ • La Voz de Durango	▪ Milenio Monterrey	▪ Coahuila
▪ La Voz de Michoacán	▪ Ocho Columnas	▪ El Norte
▪ Enfoque (Tepic)	▪ AM de León	▪ Público
▪ Pulso de SLP	▪ Imagen de Zacatecas	▪ Heraldo de León
▪ El Imparcial de Oaxaca	▪ Diario 17	▪ Cuarto Poder de Chiapas
▪ Por Esto de Mérida	▪ Tabasco Hoy	▪ El Sur de Guerrero
▪ Notiver	▪ AZ de Xalapa	▪ Presente
▪ Diario de Morelos	▪ ABC de Toluca	▪ La Opinión de Poza Rica
		▪ El Diario de Toluca

Pero todos los procesos insertan la información en la base de datos del servidor no en la máquina local.

Segunda fase

Los archivos que se programaron se subieron al servidor y se revisó que el sistema funcionara bajo la conexión a la base de datos principal.

Tercera fase

Se requiere realizar un archivo que unicamente reciba el identificador del tema y que liste sus subtemas y notas encontradas. Por cada tema se creara una carpeta en la cual se creará un index que pase los parametros al archivo base y que permita establecer una ruta diferente para cada tema.

4.5. Pruebas

Las pruebas son ejecuciones ordenadas del sistema que permiten establecer si se cumplen los objetivos de un programa o módulo. En particular el sistema esta basado en los casos de uso por lo que los casos de prueba son consecuencia directa de los casos de uso, esto facilita las situaciones que se tienen que contemplar en las pruebas y permite evaluar todas las características que son importantes para el usuario y que son descritas mediante los diagramas de casos de uso.

4.5.1. Plantillas de prueba(pruebas de sistema)

Primera fase

Caso de Uso C0. Ejecución de ART			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC0.1 Codificación de archivos			
	El administrador indica la ruta del directorio en el cual los archivos html son leídos	Los archivos son mapeados a una matriz que queda almacenada en el archivo: "codificados.dat"	✓
PC0.2 Clasificación			
	El administrador inicia la fase de clasificación en la cual la red ART decidirá si el archivo será procesado o no	Se obtiene como mínimo un 4% de error es decir mínimo 96 notas de cada 100 si son detectadas como notas y pasan a la fase de procesamiento	✓

Figura 4.33: Caso de prueba PC0. Ejecución de la red ART

Caso de Uso C1. Procesamiento de notas			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC1.1 Extracción de rubros			
	Se ejecuta el programa de extracción de información	Se obtiene el contenido de la nota y se determina si existe algún tema de interés en el texto. Las notas se almacenan en una BD	✓

Figura 4.34: Caso de prueba PC1. Procesamiento de notas

Segunda fase

Caso de Uso C2. Ingreso al sistema			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC2.1 Ingreso de usuario y contraseña			
	Ingresar usuario y contraseña de administrador	Ingresar como administrador	✓
	Ingresar usuario y contraseña de coordinador	Ingresar como coordinador	✓
	Ingresar usuario y contraseña de analista	Ingresar como analista	✓
	Ingresar usuario y contraseña de consultor	Ingresar como consultor	✓
	Ingresar usuario correcto de administrador y contraseña	No se ingresa al sistema	✓
	Ingresar usuario correcto de coordinador y contraseña	No se ingresa al sistema	✓
	Ingresar usuario correcto de analista y contraseña incorrecta	No se ingresa al sistema	✓
	Ingresar usuario correcto de consultor y contraseña incorrecta	No se ingresa al sistema	✓
	Ingresar usuario incorrecto y contraseña correcta de administrador	No se ingresa al sistema	✓
	Ingresar usuario incorrecto y contraseña correcta de coordinador	No se ingresa al sistema	✓
	Ingresar usuario incorrecto y contraseña correcta de analista	No se ingresa al sistema	✓
	Ingresar usuario incorrecto y contraseña correcta de consultor	No se ingresa al sistema	✓

Figura 4.35: Caso de prueba PC2. Ingreso al sistema

Caso de Uso C3.1. Administrar un usuario			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC3.1.1. Dar de alta un usuario			
	Llena los campos obligatorios con datos válidos	Almacena los datos y muestra una leyenda de confirmación de proceso. El sistema está listo para dar de alta un nuevo usuario	✓
	Los datos son válidos pero no se llenaron los campos obligatorios	No almacena datos y muestra un letrero que pide al usuario llenar los campos obligatorios	✓
	Se llenaron los campos obligatorios pero los datos son inválidos	No almacena datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
	Se llenaron los campos obligatorio con datos válidos y los demás datos son inválidos	No almacena datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
PC3.1.2. Modificar las características de un usuario			
	Seleccionar el nombre de un usuario de la lista presentada	Muestra todos los datos registrados de ese usuario	✓
	Modifica los campos obligatorios con datos válidos	Actualiza los datos del usuario y muestra una leyenda de confirmación de proceso. El sistema está listo para modificar otro usuario	✓
	Modifica los campos con datos válidos pero no se llenaron los campos obligatorios	No actualiza los datos y muestra un letrero que pide al usuario llenar los campos obligatorios	✓
	Modifica los campos obligatorios pero los datos son inválidos	No actualiza los datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
	Modifica los campos obligatorios con datos válidos y los demás datos son inválidos	No actualiza los datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
PC3.1.3. Eliminar un usuario			
	Seleccionar el nombre de un usuario de la lista presentada	El estatus del usuario cambia a baja y muestra una leyenda de confirmación de proceso. El sistema está listo para eliminar otro usuario	✓

Figura 4.36: Casos de prueba PC3.1. Administración de usuarios

Caso de Uso C3.2. Administrar temas			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC3.2.1. Dar de alta un tema			
	Llena el campo con datos válidos	Almacena los datos y muestra una leyenda de confirmación de proceso. El sistema está listo para dar de alta otro tema.	✓
	Se tienen datos inválidos en el campo de texto	No almacena datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
PC3.2.2. Modificar las características de un tema			
	Seleccionar el nombre de un tema de la lista presentada	Pantalla donde se puede editar el nombre del tema	✓
	Modifica el campo con datos válidos	Actualiza los datos del tema y muestra una leyenda de confirmación de proceso. El sistema está listo para modificar otro tema	✓
	Modifica los campos con datos inválidos	No actualiza los datos y muestra un letrero que pide al usuario llenar los campos con datos válidos	✓
PC3.2.3. Eliminar un tema			
	Seleccionar el nombre del tema a eliminar de la lista presentada	El estatus del tema cambia a baja y muestra una leyenda de confirmación de proceso. El sistema está listo para eliminar otro tema	✓

Figura 4.37: Caso de prueba PC3.2. Administración de temas

Caso de Uso C3.3. Administrar palabras Clave			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC3.3.1. Dar de alta una palabra clave			
	Llena los campos con datos válidos	Almacena los datos y muestra una leyenda de confirmación de proceso. El sistema está listo para insertar otra sección	✓
	Se tienen datos inválidos	No almacena datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
PC3.3.2. Modificar las características de una palabra clave			
	Seleccionar el nombre de una palabra clave de la lista presentada	Pantalla donde se puede editar el nombre de la palabra clave y el tema al que pertenece	✓
	Modifica el campo con datos válidos	Actualiza los datos de la palabra clave y muestra una leyenda de confirmación de proceso. El sistema está listo para modificar otra palabra clave	✓
	Modifica los campos con datos inválidos	No actualiza los datos y muestra un letrero que pide al usuario llenar los campos con datos válidos	✓
PC3.3.3. Eliminar una palabra clave			
	Seleccionar el nombre de la palabra clave a eliminar de la lista presentada	El estatus de la palabra clave cambia a baja y muestra una leyenda de confirmación de proceso. El sistema está listo para dar de baja otra palabra clave	✓

Figura 4.38: Caso de prueba PC3.3. Administración de palabras clave

Caso de Uso C3.4. Administrar medios			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC3.4.1. Dar de alta un medio			
	Llena el campo con datos válidos	Almacena los datos y muestra una leyenda de confirmación de proceso. El sistema está listo para dar de alta otro tema.	✓
	Se tienen datos inválidos en el campo de texto	No almacena datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
PC3.4.2. Modificar las características de un medio			
	Seleccionar el nombre de un medio de la lista presentada	Pantalla donde se puede editar el nombre del medio	✓
	Modifica el campo con datos válidos	Actualiza los datos del medio y muestra una leyenda de confirmación de proceso. El sistema está listo para modificar otro medio	✓
	Modifica los campos con datos inválidos	No actualiza los datos y muestra un letrero que pide al usuario llenar los campos con datos válidos	✓
PC3.4.3. Eliminar un medio			
	Seleccionar el nombre del medio a eliminar de la lista presentada	El estatus del medio cambia a baja y muestra una leyenda de confirmación de proceso. El sistema está listo para eliminar otro tema	✓

Figura 4.39: Caso de prueba PC3.4. Administración de medios

Caso de Uso C3.5. Administrar secciones			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC3.5.1. Dar de alta una sección			
	Llena los campos con datos válidos	Almacena los datos y muestra una leyenda de confirmación de proceso. El sistema está listo para insertar otra sección	✓
	Se tienen datos inválidos	No almacena datos y muestra un letrero que pide al usuario respetar los tipos de entrada de datos	✓
PC3.5.2. Modificar las características de una sección			
	Seleccionar el nombre de una sección de la lista presentada	Pantalla donde se puede editar el nombre de la sección y el diario al que pertenece	✓
	Modifica el campo con datos válidos	Actualiza los datos del tema y muestra una leyenda de confirmación de proceso. El sistema está listo para modificar otro tema	✓
	Modifica los campos con datos inválidos	No actualiza los datos y muestra un letrero que pide al usuario llenar los campos con datos válidos	✓
PC3.5.3. Eliminar una sección			
	Seleccionar el nombre de la sección a eliminar de la lista presentada	El estatus de la sección cambia a baja y muestra una leyenda de confirmación de proceso. El sistema está listo para dar de baja otra sección	✓

Figura 4.40: Caso de prueba PC3.5. Administración de secciones

Tercera fase

Caso de Uso C4.1. Ver las notas de interés			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC4.1.1. Ingresar a la liga de interés			
	Se escribe la URL del consultor seguido de la palabra news	Se presenta la página web con los colores adecuados al logotipo manejado y con las noticias asociadas a los temas de interés	✓

Figura 4.41: Caso de prueba PC4.1. Ver las notas de interés

Caso de Uso C4.2. Realizar búsquedas en el histórico de notas			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PC4.2.1. Ingresar al menú del tema			
	Se selecciona el tema de interés en el menú derecho.	Se abre una nueva ventana del navegador en la cual se muestran las notas del tema elegido en un rango de tiempo determinado por el administrador (un año, un mes, un semana)	✓
	Se selecciona alguna nota de interés	Se muestra una nueva ventana en la cual está el resumen de la nota y el <i>clipping</i> digital	✓
PC4.2.2. Realizar una búsqueda			
	Se selecciona la opción de búsqueda del menú derecho.	Se abre una ventana en la cual se pide el ingreso de datos útiles para buscar : * Uno o todos los temas de interés del consultor. * Una palabra de filtro que esté contenida en la nota. * Rango de la fecha de emisión de la nota, etc.	✓
	Se presiona el botón "Buscar"	Se muestra una lista con las notas que cumplen los criterios establecidos en el formulario anterior.	✓
	Se selecciona alguna nota de interés	Se muestra una nueva ventana en la cual está el resumen de la nota y el <i>clipping</i> digital	✓
PC4.2.3. Realizar un informe			
	Se selecciona la opción de generar informe del menú derecho.	Se abre una ventana en la cual se pide el ingreso de datos útiles para buscar : * Uno o todos los temas de interés del consultor. * Una palabra de filtro que esté contenida en la nota. * Rango de la fecha de emisión de la nota, etc.	✓
	Se presiona el botón "Reporte por medio"	Se muestra una lista con las notas que cumplen los criterios establecidos en el formulario anterior. Las notas están agrupadas por medio y se presentan datos de interés como el costo y valoración de cada nota.	✓
	Se presiona el botón "Reporte general Excel"	Se muestra una nueva ventana en la cual está el resumen de la nota y el <i>clipping</i> digital	✓
	Se selecciona alguna nota de interés	Descarga un archivo en excel con los resultados de la búsqueda.	✓

Figura 4.42: Caso de prueba PC4.2. Realizar búsqueda

4.5.2. Pruebas de Integración

Primera fase

La integración se facilitó debido a que los módulos se realizaron en base a funciones. La primera función relevante es la clasificación de los archivos html de cada diario para determinar si es nota o no (*function clasificacion()*), esta función a su vez se compone de la función que mapea el html a un vector (*function decodificación()*) y la que ejecuta el código de la red neuronal (*function redART2()*). La segunda función relevante es el procesamiento de la información (*function ftpLista2()*) esta función esta compuesta de muchas otras:

- Lectura del archivo html que ha pasado la fase de clasificación (*function LeerArchivos()*).
- Búsqueda del tema de interés a través de sus palabras clave en el contenido de la nota (*function obt_act_rss()*).
- Inserción en las tablas de la BD (*function inserta_util()*, *function inserta_general()*).

Segunda fase

La integración de los productos de la primera fase con la segunda consiste en agregar las funciones de la primera fase a la interfaz *web* que es producto de la segunda fase, para lograr esto se habilitó un apartado para el administrador y coordinador en el cual se ejecuta el proceso de la fase uno al momento de presionar un botón.

Pruebas de integración. Fase uno y dos				
Caso de Prueba	Datos de entrada		Salida	Evaluación
PCI.0. Ejecutar redArt2 con interfaz				
	Selecciona la pestaña de ejecución		Pantalla en la que se listan los diarios dados	✓
	Selecciona los diarios en los cuales se realizará la búsqueda y se presiona el boton "Procesar"		Se muestran los títulos de las notas encontradas	✓

Figura 4.43: Caso de prueba PCI.0. Integración de la red ART de la fase 1 con la interfaz *web* de la fase 2

Tercera fase

La integración de esta fase a las anteriores requiere una nueva interfaz *web* que se presentará en las pantallas de los coordinadores y del administrador una pestaña en la cual podrá crear una nueva página *web* para algún consultor, en esta se podrán agregar usuarios y temas además de los parámetros de color que se preferan para la apariencia de la página del consultor.

Pruebas de integración. Fase tres con anteriores			
Caso de Prueba	Datos de entrada	Salida	Evaluación
PCI.1. Crear una página de consulta			
	Ingresar a la pestaña de creación	Se presenta una pantalla que pide los datos necesarios para la creación de la página	✓
	Selecciona el identificador del consultor, se añaden temas existentes o se agregan nuevos temas, se eligen los colores y el logotipo del consultor y se presiona el botón "Crear"	Se crea la nueva página en la cual el consultor observa las notas de los temas de su interés	✓

Figura 4.44: Caso de prueba PCI.1. Visualización de la información

4.5.3. Pruebas de Rendimiento

Para realizar las pruebas de rendimiento se trabajó con un conjunto de notas de un día específico de todos los diarios involucrados. El set consta de 25000 archivos html en el cual existen un total de 3500 notas., lo cual es mayor a la demanda diaria. Las pruebas se realizaron en máquinas armadas con un procesador AMD Athlon X2 a 1.9 GHz, 512 MB en RAM y 40 GB de capacidad de almacenamiento en el disco duro. Se utilizaron alrededor de 80 temas de interés y 250 palabras clave.

4.6. Mantenimiento

El sistema es escalable, lo que permite incorporar nuevas funciones de manera muy sencilla pero sólo lo puede realizar un programador, el sistema es por si mismo un agente funcional, brinda mucho más de los resultados esperados pero pueden agregarse más módulos para automatizar más procesos a para simplificar pequeños de talles de administración. La actualización de la base de datos se realiza a través del sistema propiamente por lo que el mantenimiento a la información se realiza periódicamente a manera de agregar nuevo consultores y eliminar los temas que ya no son de interés de la búsqueda de noticias.

Capítulo 5

Seguridad

5.1. Seguridad del sistema

El lenguaje php tiene la etiqueta de inseguro debido a varias causas:

- **Programación que no requiere experiencia.** En primer lugar al ser un lenguaje muy bondadoso es utilizado por programadores que a falta de experiencia dejan huecos de seguridad que son aprovechados por cualquier usuario que tiene intenciones maliciosas. Sin embargo en este caso en particular se eligió el lenguaje PHP por que es un lenguaje que se domina, ya se tiene experiencia en los errores más comunes como la no validación de campos en un formulario, el envío de parámetros por GET sin codificación, manejo correcto de marcos
- **Inyecciones de SQL.** Existe la posibilidad de hacer inyecciones SQL mediante las cuales se pueden alterar los registros de las bases de datos involucradas en el sistema. Este problema fue resuelto filtrando de los campos de entrada de texto los caracteres y palabras que no se utilizan en el contexto del sistema. Un usuario, dependiendo de sus privilegios, puede insertar nombres propios, nombres de periódicos, secciones y temas de interés, ninguno de estos campos requiere insertar caracteres como `/&%$#»=><` ó palabras como OR, AND, SELECT, INSERT, DELETE, FILE, además de filtrar estos caracteres al momento de obtenerlos de las cajas de texto se escribió una función en *javascript* que evita que los caracteres sean escritos en las cajas de texto.
- **XSS.** Al ser un lenguaje *web*, php es vulnerable a XSS (Cross-site scripting) que es similar a la inyección SQL pero con el lenguaje javascript, donde se aprovecha la información que brinda el usuario al pasar parámetros y se pueden ejecutar toda clase de scripts, desde un inofensivo ciclo infinito de *alerts* hasta la captura de conversaciones, contraseñas y *cookies*. Para evitar los XSS se encapsuló el código en un frame que no permite observar el paso de parámetros GET a través de la barra de navegación, se ocultó también la información en la barra de estado y se utilizó la función *htmlentities ()* que permite convertir los caracteres que tienen un significado en HTML a sus entidades correspondientes.

Cada uno de los detalles anteriores se soluciona en base a la estructura y programación del sistema, lo que incluye la validación de las entidades gráficas con las que interactúa el usuario, sin embargo también existen huecos en los que la buena programación no es

suficiente escudo ante las intenciones de atacar un sistema, éstas son las múltiples opciones que brinda el lenguaje, por ejemplo:

- **El manejo de sesiones.** Si en algún momento se logra obtener mediante el arreglo `$_SESSION` el identificador de la sesión en algún sistema se puede perjudicar la sesión de los usuarios del sistema. Para minimizar los riesgos del uso de sesiones la literatura recomienda el uso de `session_regenerate_id()` que envía un nuevo identificador a la sesión con el que se trabaja en lugar del inicial y esto brinda mayor control sobre las transacciones de los usuarios.
- **El despliegue de errores.** PHP es noble al mostrar los errores dentro de sus sistemas facilitando la tarea de corrección de errores del programador pero también brinda información acerca de la estructura de los archivos y el código programado, incluso de las bases de datos y los usuarios y contraseñas escritos en los archivos de php, de manera que si esos errores son vistos por alguien que desea perjudicar el servicio *web* de un sistema puede obtener información que ayuda a lograr su fin. Para solucionar el despliegue de errores en las pantallas a las que el usuario final puede ingresar se escribieron los errores en un archivo de registro en lugar de presentarlos en pantalla, esto se realiza activando la directiva `log_errors`, además se utilizó la directiva `display_errors` para que los errores no se muestren en las pantallas del navegador. La modificación de las directivas puede realizarse editando el archivo `php.ini` o en los archivos de código, debido a que no se tiene acceso al archivo `php.ini` no se pueden configurar las directivas editando el contenido, en este caso particular se utilizó la función `ini_set_php(DIRECTIVA, VALOR)`, así que se configuraron las cuatro directivas a través de esta función.

```
ini_set_php(display_errors, fuera)
ini_set_php(error_reporting, 8191)
ini_set_php(log_errors,on)
ini_set_php(error_log, /home/logs/php_errors.log)
```

Figura 5.1: Funciones de despliegue de errores en PHP

La inclusión de archivos remotos. La inclusión de archivos externos al código local es una herramienta muy valiosa pero poco segura ya que permite que se ejecuten secciones de código que no se encuentran bajo el control del administrador ni bajo las normas de seguridad del sistema. Para evitar que se incluyan archivos remotos existen 2 directivas de control de archivos: `allow_url_fopen` que por default está activada y permite la inclusión de archivos remotos y `allow_url_include` que por default está desactivada y permite utilizar funciones como `require()`, `include()` y `require_once()` para incluir archivos remotos, ambas directivas se han deshabilitado para este sistema en particular.

En estos casos la solución tiene que ver con configuraciones del servidor y llamadas a funciones estándar que manipulan el uso de estas herramientas útiles pero peligrosas[8].

5.1.1. Niveles de usuario

Fuera de la configuración y de las particularidades de php una regla básica para el desarrollo de un sistema seguro e íntegro es el manejo de usuarios jerarquizados de acuerdo a una clasificación en la que se les otorga la capacidad de realizar transacciones con la información

que alberga el sistema, de manera que se estableció una tabla de permisos con respecto a las interfaces que un usuario puede ver.

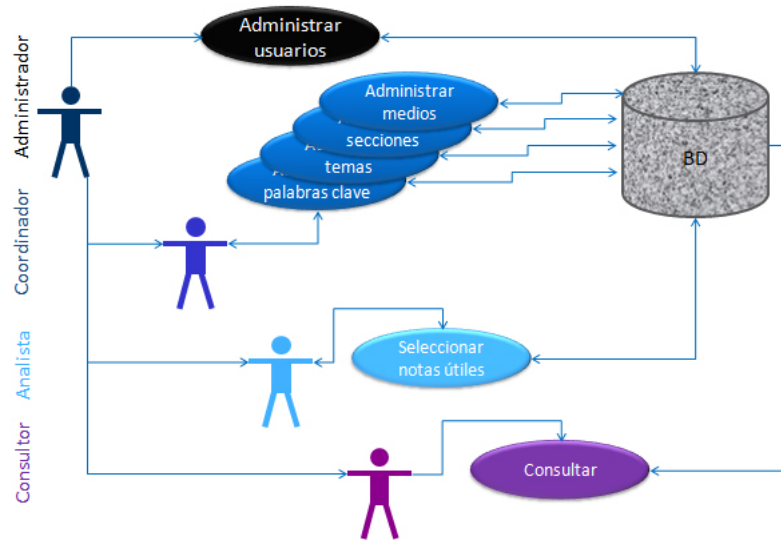


Figura 5.2: Niveles de usuario en la BD

El nivel 0 de usuario es el administrador, es el encargado de establecer en qué jerarquía se clasificará a un usuario, también puede eliminar a un usuario existente o modificar las características de los usuarios en los registros de la base de datos.

El nivel 1 es el coordinador, este es el encargado de agregar medios (El Universal, La Jornada, Reforma, etc), secciones (deportes, sociales, primera plana, política, etc), temas (Bebidas energéticas) y palabras clave (Red Bull, Full Throttle, isotónico, etc) según su criterio, esta es su función principal aunque también está capacitado para filtrar las notas que verá el consultor y el sistema le brinda ese privilegio.

El nivel 2 es el analista, su única función es juzgar si las notas que brinda la herramienta son aptas para el tema de interés, el trabajo realizado en este nivel de usuario tiene contacto directo con lo que observa el consultor por lo que es en donde se tiene una mayor cantidad de usuarios.

El nivel 3 es el consultor, solo visualiza la información que han catalogado los demás niveles de usuario.

5.2. Seguridad de la base de datos

La base de datos está alojada en un servidor externo bajo sistema operativo Debian, para realizar la conexión a la base de datos se requiere dar de alta el usuario en MySQL, y se proporcionan el nombre de usuario y la contraseña para poder ser utilizados en los códigos correspondientes.

Se manejarán dos usuarios el primero para listar catálogos y para los usuarios que se conectarán fuera del área de trabajo que solo tendrá privilegios para consultar la base de datos. El segundo usuario además de listar los datos de la tabla puede insertar datos en cualquier tabla de la base, borrar registros y actualizar.

Las mayoría de las transacciones que se realizan en la base de datos no requieren que la conexión quede establecida durante todo el proceso de ejecución de una interfaz, de manera que cuando se hace una petición a la base de datos una vez que se obtienen los datos requeridos, se cierra la conexión a la base de datos y solo se reabre cuando se requiere manipular nuevamente información de las tablas. De esta manera las conexiones a la base solo se realizan cuando son necesarias y no se mantienen abiertas ni alojadas en variables de sesión.

Algoritmo 5.1 Código SQL de creación de usuarios

–Solo privilegio de seleccionar

```
CREATE USER 'mortal'@'%' IDENTIFIED BY '*****';
```

```
GRANT SELECT ON * . * TO 'mortal'@'%'
```

–Solo privilegio

```
CREATE USER 'agregable'@'localhost' IDENTIFIED BY '*****';
```

```
GRANT SELECT , INSERT ,
```

```
UPDATE , DELETE ON * . * TO 'agregable'@'localhost' IDENTIFIED BY '*****';
```

```
WITH MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0
```

```
MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0 ;
```

5.2.1. Niveles de usuario

Con respecto a las tablas de la base de datos se establece el mismo número de niveles de usuario para delimitar cuáles tablas pueden manipular un solo usuario. Se establecieron las transacciones que se pueden realizar sobre cada una de las tablas.

Administrador
Nivel 0

	INSERTAR	CONSULTAR	MODIFICAR
USUARIO	✓	✓	✓

	INSERTAR	CONSULTAR	MODIFICAR
NOTA			

	INSERTAR	CONSULTAR	MODIFICAR
MEDIO			

	INSERTAR	CONSULTAR	MODIFICAR
SECCION			

	INSERTAR	CONSULTAR	MODIFICAR
ACTOR			

	INSERTAR	CONSULTAR	MODIFICAR
PALABRACVE			

Figura 5.3: Permisos del usuario nivel 0 en las tablas: **usuario**, **notas**, **medio**, **seccion**, **actor**, **palabracve**

Coordinador
Nivel 1

	INSERTAR	CONSULTAR	MODIFICAR
USUARIO			

	INSERTAR	CONSULTAR	MODIFICAR
NOTA	✓	✓	✓

	INSERTAR	CONSULTAR	MODIFICAR
MEDIO	✓	✓	✓

	INSERTAR	CONSULTAR	MODIFICAR
SECCION	✓	✓	✓

	INSERTAR	CONSULTAR	MODIFICAR
ACTOR	✓	✓	✓

	INSERTAR	CONSULTAR	MODIFICAR
PALABRACVE	✓	✓	✓

Figura 5.4: Permisos del usuario nivel 1 en las tablas: **usuario**, **notas**, **medio**, **seccion**, **actor**, **palabracve**

Analista
Nivel 2

	INSERTAR	CONSULTAR	MODIFICAR
USUARIO			

	INSERTAR	CONSULTAR	MODIFICAR
NOTA		✓	✓

	INSERTAR	CONSULTAR	MODIFICAR
MEDIO			

	INSERTAR	CONSULTAR	MODIFICAR
SECCION			

	INSERTAR	CONSULTAR	MODIFICAR
ACTOR			

	INSERTAR	CONSULTAR	MODIFICAR
PALABRACVE			

Figura 5.5: Permisos del usuario nivel 2 en las tablas: **usuario**, **notas**, **medio**, **seccion**, **actor**, **palabracve**

Consultor
Nivel 3

	INSERTAR	CONSULTAR	MODIFICAR
USUARIO			

	INSERTAR	CONSULTAR	MODIFICAR
NOTA		✓	

	INSERTAR	CONSULTAR	MODIFICAR
MEDIO			

	INSERTAR	CONSULTAR	MODIFICAR
SECCION			

	INSERTAR	CONSULTAR	MODIFICAR
ACTOR			

	INSERTAR	CONSULTAR	MODIFICAR
PALABRACVE			

Figura 5.6: Permisos del usuario nivel 3 en las tablas: **usuario**, **notas**, **medio**, **seccion**, **actor**, **palabracve**

El administrador pertenece al nivel 0 y puede insertar, modificar y consultar los registros de la tabla **usuario**.

Los coordinadores se clasifican en el nivel 1 y tienen privilegios para insertar, modificar y consultar los registros de las tablas **medio**, **actor**, **seccion**, **palabracve** y **nota**.

Los analistas se engloban en el nivel 2 y sólo pueden consultar y modificar la tabla **nota**.

Los consultores están en el nivel 3 y únicamente pueden consultar algunos registros de la tabla **nota**. De hecho cada consultor tendrá acceso a diferentes registros de la tabla, esto depende del tema al que pertenezca la nota insertada. Cada consultor recibe únicamente la url que contiene el tema de su interés, por ejemplo un consultor de política puede estar especialmente interesado en el partido político PRD, en los estados de la República, en las corrientes internas del PRD, en los legisladores y en el Gobierno Capitalino, de manera que recibirá la *URL*:

<http://intermediamexico.com/prdnews/consulta.php>

donde se resumirá la información de su interés.

Figura 5.7: Ejemplo de página *web* de consulta

Capítulo 6

Resultados

La duración del proceso de revisión de las notas electrónicas es un proceso laborioso que sin ayuda de herramientas de búsqueda lleva alrededor de 4 [h], requiere de 2 personas que revisan manualmente los sitios en la red y se tiene un alcance de 10 diarios electrónicos, además existe un proceso adicional que requiere separar las noticias según el tema de interés para posteriormente ser enviado por correo electrónico a los coordinadores y analistas. Con la versión automática del proceso se logró reducir el tiempo de búsqueda, cuando sólo se toman en cuenta los 10 diarios principales que comprendían el proceso manual la búsqueda se realiza en 1.2 [h] es decir se reduce el tiempo de búsqueda en un 70 % la clasificación es automática por lo que se elimina por completo esta parte del trabajo original y la parte de revisión y captura se reduce a solo 20 minutos ya que la nota ya está capturada y solo es necesario juzgar si la nota se hace visible al usuario que consulta, este proceso se reduce a cerca de 75 %, de manera que en total el proceso manual tenía una duración de 7 [h] y el proceso automático dura 1.53[h]. De manera que la revisión de diarios a través del *software* desarrollado mejora el proceso un 78.1 %

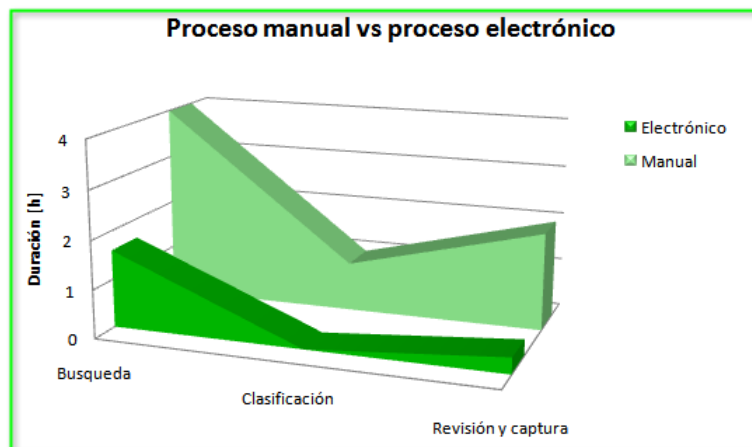


Figura 6.1: Comparación entre el proceso manual con el electrónico

De esta manera se pudieron incorporar más diarios para tomar estadísticas de interés

acerca de un tema, por ejemplo su aparición en los diarios electrónicos en México, el número de notas que un diario publica acerca de un determinado tema en estudio, la relevancia de una nota por región geográfica, etc. Se agregaron 63 diarios a la búsqueda teniendo un total de 73 periódicos digitales de toda la república.

El número de notas que se revisaban en el proceso manual es alrededor de 1000, mediante el proceso automático se revisan cerca de 4000. El proceso manual brinda en promedio 87 noticias de los temas de interés, pero se tiene una efectividad del 70 %, es decir que existió un 30 % de notas que contenían temas de interés y que no fueron detectadas, mediante la automatización se encuentran en promedio 230 notas con una efectividad de 98 %.

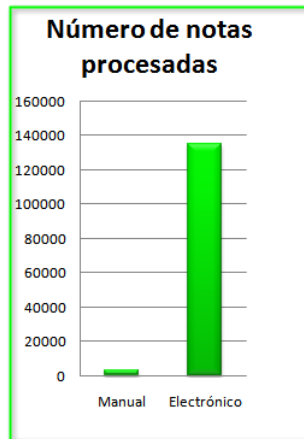


Figura 6.2: Comparación del número de notas procesadas

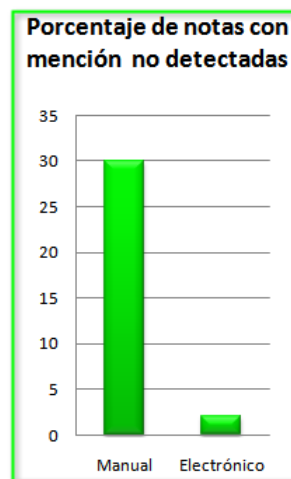


Figura 6.3: Porcentaje de error en los procesos

Una vez que se han mostrado las ventajas del proceso automatizado se practicaron algunas variantes en el programa para observar si el resultado mejora, por ejemplo se utilizaron

diferentes codificaciones en los vectores de entrada de la red neuronal para las etiquetas de apertura y cierre de html de manera que es distinto encontrar por ejemplo `<html>` que `</html>`, sin embargo los resultados no mostraron ninguna mejoría en la etapa de entrenamiento y prueba.

Cuando no se utiliza la red neuronal en el proceso la búsqueda de las palabras clave en los archivos es un proceso largo, los archivos revisados aumentan a 610 000 y el tiempo de procesamiento aumenta a 18 horas en promedio por día, lo cual es un muy mal resultado para trabajar en una situación donde se requiere información actual a determinada hora del día así que el *software* es funcional gracias a la implementación de la red ART 2.

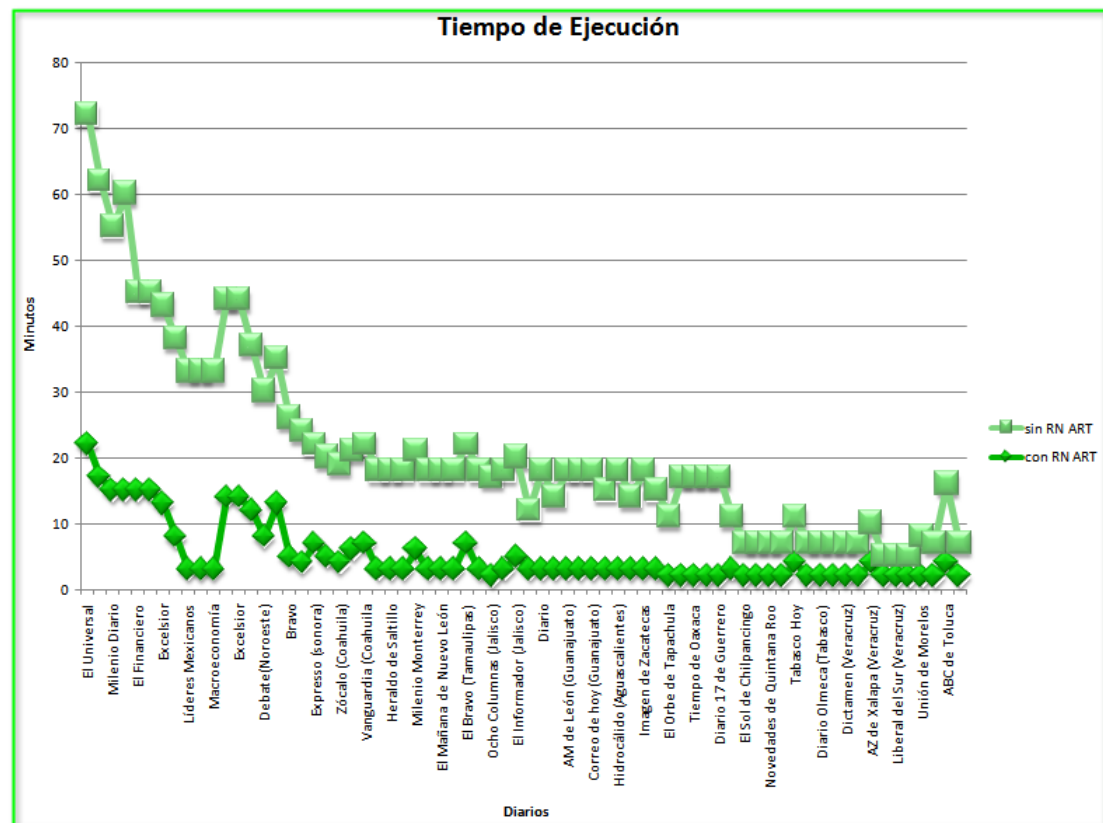


Figura 6.4: Proceso sin red neuronal y con red neuronal

La efectividad de la red neuronal para detectar una nota es, en promedio, 99.76% es decir que en promedio no detecta 30 notas por cada 4000 aproximadamente. El diario en el cual la red ART tiene menor efectividad es El Universal ya que en promedio se escapan 12 de cada 300 notas y existen 63 diarios en los cuales la red ART clasifica perfectamente los archivos.

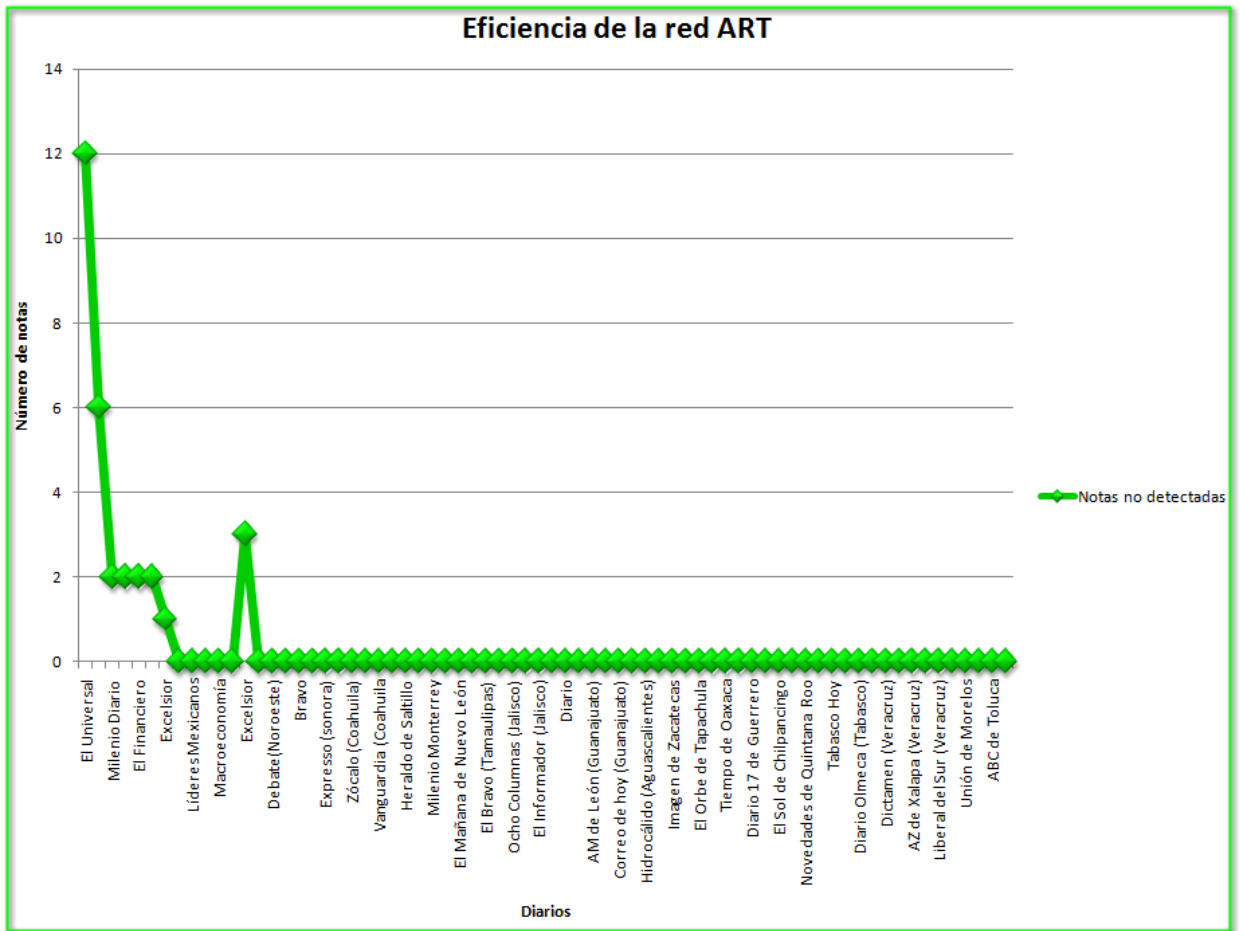


Figura 6.5: Eficiencia de la red ART2

Capítulo 7

Conclusiones

Utilizar versiones funcionales de los sistemas es un método que funciona tanto al usuario final como al programador, ya que permite ir automatizando el trabajo de los usuarios finales y éstos comprenden y valoran de mejor manera el desarrollo del sistema, por otro lado el desarrollador no se retrasa en la entrega de módulos y puede interactuar mejor con los usuarios al detectar sus necesidades en base a lo que se va desarrollando. Además los requerimientos se plantean de una forma más específica y clara para ambas partes.

El buen diseño de una base de datos es necesario, si se trabaja sobre diseños ya realizados el sistema desarrollado no mejora el proceso de automatización en la misma medida que cuando la base de datos esta normalizada.

El uso de una base de datos normalizada no es una característica de interés para el usuario final debido a que, de manera inmediata, no brinda mejor tiempo de respuesta en las peticiones de consulta, sin embargo la base normalizada presenta un mejor comportamiento conforme los datos almacenados aumentan al ser alimentado el sistema implementado.

Frente a la necesidad de obtener sistemas que funcionen más rápidamente y con mayor eficacia el concepto de la inteligencia artificial está aplicándose en todas las disciplinas en las que se busca que los procesos de análisis realizados por el ser humano sean realizados por las computadoras. El cerebro del hombre es muy complejo y no se tiene un algoritmo capaz de procesar la información en una computadora como se hace biológicamente pero del análisis e investigación han surgido ideas como las redes neuronales artificiales que, de manera muy general, simulan el recorrido de la información en las neuronas biológicas y la función de estas en el aprendizaje. Las redes neuronales artificiales y, en general, todas las técnicas de inteligencia artificial se van a utilizar cada vez más en diversas áreas y disciplinas para lograr eficiencia y automatización de las tareas más simples y cotidianas.

En particular, la red neuronal basada en la teoría de resonancia adaptativa es un excelente clasificador, el único problema con su uso es la definición de parámetros. El algoritmo clasificador fue una buena elección ya que los resultados en el proceso final se contabilizan en menos horas de trabajo y menos procesamiento de datos, lo cual retribuye directamente en ahorro de recursos humanos y hasta ahorro monetario. La definición de los parámetros de una red ART' 2 puede realizarse con métodos tan simples como el famoso prueba y error, con algoritmos genéticos ó incluso a través de un sistema experto.

La implementación de las técnicas que permiten un comportamiento adecuado de los sistemas desarrollados se vuelve una tarea más simple debido a la gran cantidad de herramientas con las que se cuenta actualmente, existe una gran cantidad de lenguajes de programación, de servidores *web*, de algoritmos específicos para una tarea como ordenamiento

o aprendizaje, de procesadores, de tarjetas de video, etc. El *hardware* y *software* con el que se cuenta es un paso muy grande para realizar las tareas que el ser humano ingeniosamente diseña, y cuando el *hardware* y *software* actual no le basta, crea otras corrientes de desarrollo utilizando métodos estadísticos, métodos de inteligencia artificial y estudio de los datos a través de la teoría de la información con el fin de resolver problemas que actualmente no se pueden resolver con las herramientas que se tienen.

PHP es un lenguaje *web* que permite desarrollar en menos líneas de código los algoritmos más complejos debido a la gran cantidad de funciones definidas que ofrece, la velocidad con la que trabaja las cadenas no es mayor a la de ningún lenguaje compilado pero el tiempo total del proceso cumple con las necesidades del usuario final. Además la velocidad con la que se programó esta aplicación fue grande gracias a las bondades que ofrece PHP como el manejo de tipos de datos, manejo de nombre de variables, arreglos multidimensionales y su familiaridad en sintaxis con el lenguaje C lo que ayuda al desarrollador a dejar a un lado los problemas de la implementación y concentrarse en el análisis de los algoritmos.

El análisis de la corrección y complejidad de un algoritmo es una buena práctica de desarrollo de *software* que no es muy común debido a las exigencias del trabajo en las entidades laborales, sin embargo es una fase que, cuando es tomada en cuenta, brinda un mejor desempeño del sistema final, aportando calidad al proceso y ahorro de tiempo y dinero.

La mejora del proceso en estudio significó desarrollar varias funciones, mucho ingenio para resolver los problemas que surgían en cada función y un nivel de abstracción muy grande para generalizar estas funciones. En la medida en la que se pueden generalizar las acciones más utilizadas se disminuye la cantidad de código al reutilizar funciones, se requiere un mayor nivel de programación y se eleva la calidad del sistema.

El uso de reconocimiento de patrones facilita la tarea descrita en gran medida. Con este trabajo se ilustra cómo la ingeniería en computación puede mejorar los procesos cotidianos con el uso de las herramientas adecuadas. Se verificó además que la red de resonancia adaptativa es un excelente clasificador de aprendizaje no supervisado, de manera que puede ser utilizado en tareas delicadas que requieran precisión en el resultado en áreas como seguridad, economía y medicina. En esta última disciplina existen muchas situaciones en las que se puede aplicar el reconocimiento de patrones: diagnóstico de enfermedades, agrupamiento de proteínas, automatización de cirugías, etc. Además la autoorganización de las redes basadas en la teoría de la resonancia adaptativa puede utilizarse en investigación para encontrar características que el hombre no ha descubierto.

La automatización de tareas permite al hombre poner su atención en los procesos que aún no pueden ser automatizados para realizarlos con mayor calidad o para estudiarlos y le permite generar nuevas ideas y aportar conocimiento en áreas en las cuales aún hay mucho por descubrir.

Referencias

- [1] Barranco de Areba, J. (2001). *Metodología del análisis estructurado de sistemas*, 2 ed, vol. 20 de *III*, Cap. 4, p. 282. Universidad Pontificia de Comillas.
- [2] Brady, R. J. (1991). *Sistema nervioso*.
- [3] Carpenter, G. The Adaptive Resonance Theory. In *The handbook of brain theory and neural networks*, pp. 87–90. M. Arbib.
- [4] Carpenter, G. A. and Grossberg, S. (1987). *ART 2: self-organization of stable category recognition codes for analog input patterns*. (26):13.
- [5] Chauvin, Yves and Rumelhart, D. E. (1995). *Backpropagation: theory, architectures, and applications*, Cap. 1, pp. 1–19. Routledge, Hillsdale, New Jersey 07642.
- [6] Cobo, A. (2009). *PHP y MySQL. Tecnologías para el desarrollo de aplicaciones*. Ediciones Díaz de Santos, España. (Cap. 1-3, 7).
- [7] Dietrich Buhmann, M. (2003). *Radial basis functions: theory and implementations*, Cap. 1, pp. 2–3. Cambridge University Press, Cambridge, UK.
- [8] Egeberg, D. (2008). PHP Security. PHP FREAKS. <http://www.phpfreaks.com/tutorial/php-security> Última consulta: 4.feb.2010.
- [9] Forouzan, B. A. (2003). *Introducción a la ciencia de la computación*. Cengage Learning Editores, México, Thomson.
- [10] Galaviz, J. (1008). *Estructuras de datos y análisis de algoritmos*. p. 87.
- [11] Haykin, S. (1999). *Neural network, A comprehensive foundation*, 2° ed. Prentice-Hall, USA, New Jersey 07458. (Cap. 14).
- [12] Herrera, J. (2004). *Automatización de bibliotecas*. p. 250.
- [13] Holdener, A. T. (2008). *Ajax: the definitive guide*. O' Reilly, USA, CA 95472.
- [14] Hopfield, J. (1982). *Neural networks and physical systems with emergent collective computational abilities*. *Proceedings of the National Academy of Sciences of the USA*, 79(8).
- [15] I Noguerras, G. B. (2008). *Introducción informal a MatLab y Octave*. Version 1.2, (Cap. 1).
- [16] Isasi, P. (1997). *Lenguajes, gramáticas y autómatas*. p. 325.

- [17] Kleinberg, Jon; Tardos, E. (2006). *Algorithm Design*, Cap. 2, pp. 29–39. Pearson.
- [18] Laurie, Ben ; Laurie, P. (2003). *Apache: the definitive guide*, 3 ed, Cap. 1, pp. 1–36. O’Reilly, USA, CA 95472.
- [19] L.Morales. *Scheduling a Bridge Club by Tabu Search*.
- [20] Loïc, M. (2005). *Introducción a la ingeniería del software*, Cap. 10, p. 112. Delta Publicaciones.
- [21] Mejia, Ivan; Kuri, A. (2008). *Sistematización completa del diseño de máquinas de soporte vectorial óptimas empleando algoritmos evolutivos*.
- [22] Meyer, E. A. (2007). *CSS: the definitive guide*, 3 ed. O’Reilly, USA, CA 95472.
- [23] Moore, A. *K-means and hierarchical clustering*. Carnegie Mellon University.
- [24] Nello, Cristianini; Taylor, S. (2000). *An Introduction to Support Vector Machines and other kernel-based learning methods*, Cap. 1.
- [25] Nigrin, A. (1993). *Neural networks for pattern recognition*. MIT Press, Cambridge, Massachusetts. (Cap. 2).
- [26] Sommerville, I. (2005). *Ingeniería de Software*. Addison-Wesley, España.
- [27] Valderrama, J., editor (1995). *Informacion Tecnologica*. Sociedad Editorial del Norte, La Serena, Chile.
- [28] Viso, E. (2008). *Introducción a la teoría de la computación (Autómatas y Lenguajes Formales)*, Cap. 8, p. 215. Facultad de Ciencias, UNAM.
- [29] Warford, J. S. (2005). *Computer systems*, 3 ed, Cap. 6, p. 255. Jones & Bartlett Learning, USA.
- [30] Web: Arredondo Guzmán, M. A. (2008). AJAX: Asynchronous JavaScript And XML. Entérate, UNAM. <http://www.enterate.unam.mx/Articulos/2006/enero/ajax.htm> Última consulta: 14.1.2010.
- [31] Web: Córdova Cruz, C. (2006). Tecnología CSS, separar contenido de presentación. Mati, UNAM. http://www.mati.unam.mx/index.php?option=com_content&task=view&id=85&Itemid=36 Última consulta: 20.marzo.2010.
- [32] Web: DevTopics. Lenguajes de Programación. UNAM. <http://www.devtopics.com/most-popular-programming-languages/> .Última consulta: 22.abril.222.
- [33] Web: Facultad de Ingeniería (2010). Reconocimiento de Patrones. Biometría Informática, Facultad de Ingeniería, UNAM. <http://redyseguridad.fi-p.unam.mx/proyectos/biometria/basesteoricas/reconocimiento.html> Última consulta: 1.marzo.2010.
- [34] Web: Facultad de Química, U. (2010). NEURONAS Y NEUROTRANSMISORES. Facultad de Química, UNAM. http://depa.pquim.unam.mx/amyd/archivero/NEURONASYNEUROTRANSMISORES_1118.pdf Última consulta: 1.marzo.2010.

- [35] Web: Galindo Riaño, P. Redes de una capa: Perceptrón, Adaline, Madaline. Universidad de Cádiz. http://www2.uca.es/dept/leng_sist_informaticos/preal/23041/transpas/D-PerceptronyAdaline/ppframe.htm Última consulta: 10.marzo.2010.
- [36] Web: Matteucci, M. A Tutorial on Clustering Algorithms. Politécnico de Milán. http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/index.html Última consulta: 22.marzo.2010.
- [37] Web: MSDN, M. a. a. (2010). Política de privacidad. Microsoft. http://msdn34.e-academy.com/elms/Storefront/PrivacyPolicy.aspx?campus=unam_ccdm&np1=21#6 Última consulta: 27.marzo.2010.
- [38] Web: Oktaba, H. (2010). Patrones. MCC, UNAM. <http://www.mcc.unam.mx/cursos/Algoritmos/javaDC99-2/patrones.html> Última consulta: 1.marzo.2010.
- [39] Web: Paris, M. (2010). Autorización (Zend_Acl) y Autenticación (Zend_Auth). Revista Mexicana de Física, Facultad de Ciencias, UNAM. <http://rmf.fciencias.unam.mx/paris/Zend/Zend-Acl-y-Zend-Auth/> Última consulta: 4.marzo.2010.
- [40] Web: Revista UNAM (2004). Estándar 568-A. UNAM. <http://www.revista.unam.mx/vol.5/num5/art28/art28-1a.htm>.

Glosario

Ajax. *Asynchronous JavaScript And XML*, por sus siglas en inglés. Es una combinación de JavaScript, que trabaja del lado del cliente, y de lenguajes que procesan la información en el servidor y la entregan como una cadena de texto o en un archivo XML, lo que proporciona una gran posibilidad de uso a los datos, en conjunto con las peticiones asíncronas, estas se refieren a que se puede mostrar la información que el servidor regresa sólo en las secciones de la página en donde nos interesa mostrarla, sin necesidad de actualizarla por completo.

Algoritmo. Secuencia de pasos finitos que permiten cumplir con un fin determinado.

ART. *Adaptive Resonance Theory*. Teoría desarrollada por Stephen Grossberg y Gail Carpenter. Surge de la analogía con el cerebro humano y su principal objetivo es dar al proceso de aprendizaje de los sistemas estabilidad pero sin eliminar la capacidad de agregar a su memoria nuevos patrones. Trabaja únicamente con valores binarios.

ART2. Versión de la ART que permite trabajar con números reales.

Autenticación. En inglés *authentication*, es comprobar que la persona sea quien dice ser.

Complejidad computacional. Es una medida independiente de las características de las computadoras que permite clasificar los algoritmos según el número de pasos u operaciones que necesita para terminar en función del número de entradas.

Computable. Proceso que puede ser descrito en términos de una Máquina de Turing y por lo tanto puede ser implementado computacionalmente.

Cookies. Pequeños bloques de información que el navegador almacena en el disco duro del equipo.

Correctez. Criterio de evaluación de un algoritmo que permite averiguar si una serie de pasos cumplen o no con una tarea determinada a través de invariantes que el mismo proceso proporciona.

CSS. *Cascading Style Sheets*. Es un lenguaje de estilo en cascada que se puede utilizar conjuntamente con HTML para definir el formato visual de una página *web*, sin forzar el

código para adaptar los documentos al diseño de presentación y sin colocar elementos en la página que no favorecen el contenido.

Datos de entrenamiento. Muestras del fenómeno que se estudia organizados como vectores.

Diagrama de flujo. Representación gráfica de un algoritmo.

Dilema elasticidad - plasticidad. Búsqueda de un equilibrio entre el aprendizaje y la buena memoria, las mayoría de las redes neuronales pierden su conocimiento actual si son entrenadas nuevamente, de manera que el dilema busca dejar intacto el conocimiento actual y no perder la capacidad de incluir nuevos patrones en la memoria.

Dimensión VC. Concepto de la teoría del aprendizaje estadístico propuesto por Vapnik y Chervonenkis que brinda una cota superior del número de vectores que pueden separarse a través de un algoritmo de clasificación estadística.

Escalamiento. Capacidad de agragar nuevas características a un sistema de manera fácil y segura según vayan aumentando las necesidades de los usuarios.

HTML. *Hyper Text Markup Language.* Es el lenguaje básico para la creación de páginas *web*.

Inteligencia artificial. Disciplina que a través de analogías con los seres vivos, la probabilidad, estadística y lógica desarrolla agentes que brindan respuestas a los problemas que no suelen resolverse por métodos deterministas, buscando que se reaccione con raciocinio sin necesidad de depender de un agente vivo.

Interfaz. Medio a través del cual interactúan dos entidades. Comúnmente se trata de un usuario final y un bloque de software.

Inyección de SQL. Inserción de código SQL a través de interfaces mal validadas con fines diferentes a las que refiere un sistema informático.

Mantenimiento de un sistema. Reparación, administración o escalamiento de un sistema informático.

Neurona. En computación llamo también perceptrón, es una entidad que permite la transmisión y procesamiento de la información, la idea de trabajar con este concepto surge de la analogía con las neuronas de los seres vivos.

En los seres vivos son las células funcionales del tejido nervioso. Ellas se interconectan formando redes de comunicación que transmiten señales por zonas definidas del sistema nervioso. Los funciones complejas del sistema nervioso son consecuencia de la interacción entre redes de neuronas, y no el resultado de las características específicas de cada neurona individual.

No computable. Proceso que no puede ser descrito en términos de una Máquina de Turing y por lo tanto no puede ser implementado computacionalmente. La complejidad de estos procesos es muy elevada.

Patrón. Modelo repetitivo y generalizado.

PHP. *PHP Hypertext Preprocessor.* Es un proyecto de la fundación de software libre Apache. Es un lenguaje de scripts muy usado, de propósito general que está especialmente hecho para desarrollo en el Web.

Proceso de clasificación. Fase en el proceso de ejecución de una red neuronal ART2 en el cual se define a qué categoría pertenece un determinado vector de entrada.

Proceso de entrenamiento. Fase en el proceso de ejecución de una red neuronal ART2 donde el valor del parámetro β es diferente de 0 y la red aprende cada nuevo patrón presente en los datos de entrenamiento.

Reconocimiento de patrones. Disciplina que se encarga de la descripción y clasificación de objetos, personas, señales, representaciones, etc. Trabaja con base en un conjunto previamente establecido de todos los posibles objetos (patrones) individuales a reconocer y una amplia gama de métodos matemáticos y heurísticos para la clasificación.

Red neuronal artificial. Es una serie de neuronas interconectadas entre si que reciben un vector de entrada de datos y ofrecen una salida. Cada conexión modifica los datos que recibe de acuerdo a un valor real o entero que la caracteriza. Las neuronas también modifican los valores de entrada de acuerdo a sus funciones de activación. El concepto de red neuronal artificial se origina de la analogía con el cerebro de los seres humanos y de algunos otros seres vivos.

Sesión. Un protocolo que permite controlar el acceso individual a una página o grupo de páginas determinado, mediante la asignación de un identificador de único para cada usuario. Con ello se puede conseguir hacer un seguimiento de cada usuario a lo largo de su visita en una página web.

SQL. *Structured Query Language.* Es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar -de una forma sencilla- información de interés de una base de datos, así como también hacer cambios sobre ella.

Tiempo polinomial. Tiempo menor o igual a una función polinomial cuya variable independiente es, comúnmente, el número de entradas.

Transacciones Una unidad básica de cómputo consistente y confiable. Es una colección de acciones que hacen transformaciones consistentes de los estados de un sistema preservando la consistencia del mismo.

XML. *Extended Markup Languaje.* Es un metalenguaje ya que el usuario define sus propias etiquetas. La estructura de un documento XML facilita la visualización de la información.

XSS. *Cross Site Scripting.* Vulnerabilidad de los sistemas *web*, basado en las etiquetas de html.

Apéndices

Apéndice A. Documentos entregables

Especificación de requisitos de *Software* (ERS)

Ámbito del sistema

El sistema a construir deberá ser capaz de manejar los archivos fuente de los diarios y brindar al usuario final la información bien organizada y sin errores de búsqueda de las palabras clave de los temas de interés.

Además deberá contar con un sistema independiente para la administración del mismo.

El ámbito del sistema desarrollado abarca, además de la realización del sistema en sí, las bases de datos que sean necesarias para dicho funcionamiento.

Visión general del documento

Este documento consta de tres secciones. En la primera se plantea una introducción al desarrollo del sistema y proporciona una visión general de la ERS. En la segunda sección se describen las principales funciones que debe realizar el software en desarrollo, los datos asociados y los factores, restricciones y dependencias que afectan al desarrollo. En la última sección se definen detalladamente los requisitos que debe satisfacer el sistema para facilitar el desarrollo y la comunicación con los usuarios finales del sistema.

Descripción general

El sistema de interactuará con todas los coordinadores, analistas y consultores, para los cuales se utilizará el concepto de usuario, entendiéndose por usuario a aquel que manipulará el *software*.

Funciones del sistema Acceso a la información

El sistema tendrá acceso a una base de datos centralizada y esta no será replicada en cada uno de los clientes para evitar falta de información.

Mecanismos de navegación

La interfaz de usuario incluirá mecanismos que faciliten la navegación y uso del sistema tanto en el procesamiento de la información como en el sistema administrativo.

Características de los usuarios

Las interfaces de usuario deberán ser de manejo intuitivo, fácil de aprender y sencillo de manejar. El sistema deberá presentar un alto grado de usabilidad. Lo deseable sería que un usuario nuevo se familiarizase con el sistema en muy poco tiempo, siendo de uso inmediato para las funciones de consulta y administración de temas, palabras clave, medios y secciones. Los usuarios finales tienen conocimientos básicos de computación pero no son programadores ni usuarios expertos.

Restricciones

1. El sistema se diseñará según un modelo cliente y cliente/servidor.
2. El sistema basará sus comunicaciones en protocolos estándar de Internet.

Requisitos específicos

En este apartado se presentan los requisitos funcionales que deberán ser satisfechos por el sistema. Todos los requisitos aquí expuestos son esenciales, es decir, no sería aceptable un sistema que no satisfaga alguno de los requisitos aquí presentados. Estos requisitos se han especificado teniendo en cuenta, entre otros, el criterio de ejecución: dado un requisito, debería ser fácilmente demostrable si es satisfecho o no por el sistema.

- El sistema debe garantizar la correcta búsqueda de las palabras clave.
- El sistema deberá obtener la información que los analistas necesitan: nota y tema de interés que se asocia.
- El módulo de administración debe restringirse con niveles de usuarios.
- Diseñar una base de datos que contenga toda la información concerniente a los temas de interés.
- Responder de forma eficaz y rápida a todas las consultas formuladas por los usuarios.

Requisitos funcionales

Sistema de procesamiento:

REQ01: Consulta de palabras clave.

El sistema podrá acceder a toda la información que se recopile de la búsqueda de las palabras claves.

REQ02: Captura de notas.

El sistema deberá encontrar la información general de las noticias y almacenarlo en una base de datos.

REQ03: Presentación.

La interfaz de usuario, permite visualizar, por tema, las noticias que se encontraron según sus palabras clave.

REQ04: Consulta.

La consulta de las notas no requiere validación, solo la administración.

Sistema de administración

REQ05: Ingreso de palabras clave.

Debe poder permitir el ingreso de los detalles de cada tema de interés con sus palabras clave.

REQ06: Administradores.

Los administradores del sistema informático pueden agregar usuarios de cualquier nivel y editar las palabras clave de los temas de interés.

REQ07: Analistas.

Los analistas deben visualizar las notas de sus temas de interés desde cualquier punto de la *www*.

Especificaciones de hardware

REQ08: Procesador.

El software completo es muy costoso computacionalmente por esto se requiere como mínimo un procesador Pentium IV y 512MB de memoria RAM, el espacio en disco duro puede ser de 40GB en el peor caso.

Especificaciones de *software*

REQ09: Interfaces de comunicación

Cliente y Cliente / Servidor

REQ10: interfaces

Apariencia de interfaces institucionales e intuitivas

Requisitos de rendimiento

REQ11: Consumo de recursos

La complejidad de las aplicaciones de usuario debe ser lo más baja posible, se deben implementar algoritmos eficientes y consumir los recursos necesarios para obtener una respuesta en el menor tiempo posible.

Requisitos de desarrollo

REQ12: Ciclo de vida.

El ciclo de vida elegido para desarrollar el sistema será el modelo evolutivo, de manera que se puedan incorporar fácilmente cambios y nuevas funciones.

Requisitos tecnológicos

REQ13: Portabilidad

El sistema debe cumplir con las normas de ingeniería de *software* y debe asegurarse el funcionamiento de este en el sistema operativo Windows XP.

REQ14: Mantenimiento

El sistema es susceptible de ser ampliado. Por tanto deberá diseñarse para su fácil mantenimiento, aplicando para su desarrollo las metodologías que para ello sean precisas.

Diagrama de Flujo de Datos

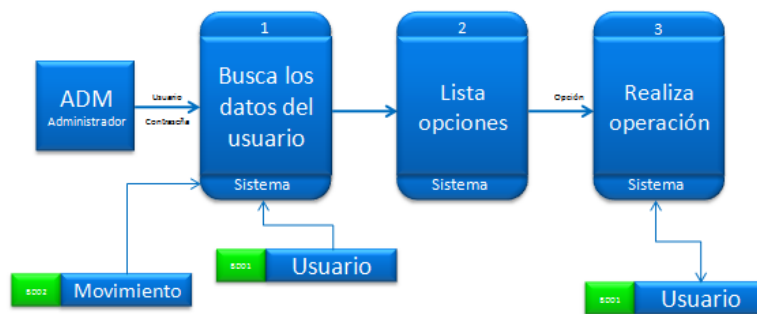


Figura 7.1: Procesos que ejecuta el Administrador



Figura 7.2: Procesos que ejecuta el Coordinador

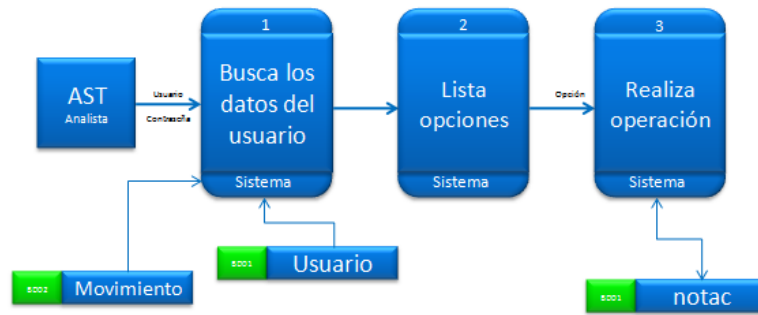


Figura 7.3: Procesos que ejecuta el Analista

Base de Datos

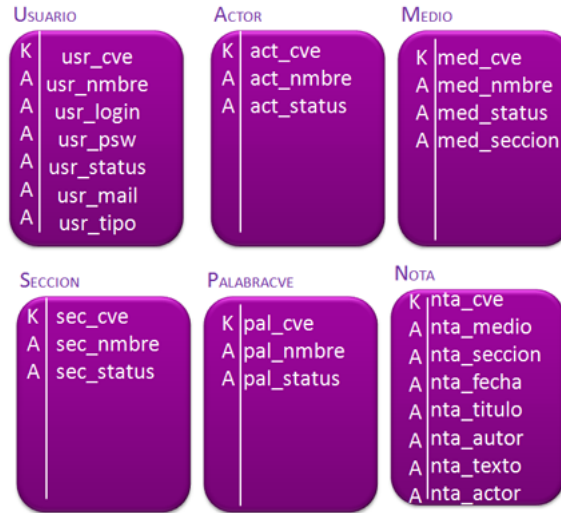


Figura 7.4: Entidades base

Diagrama Entidad-Relación

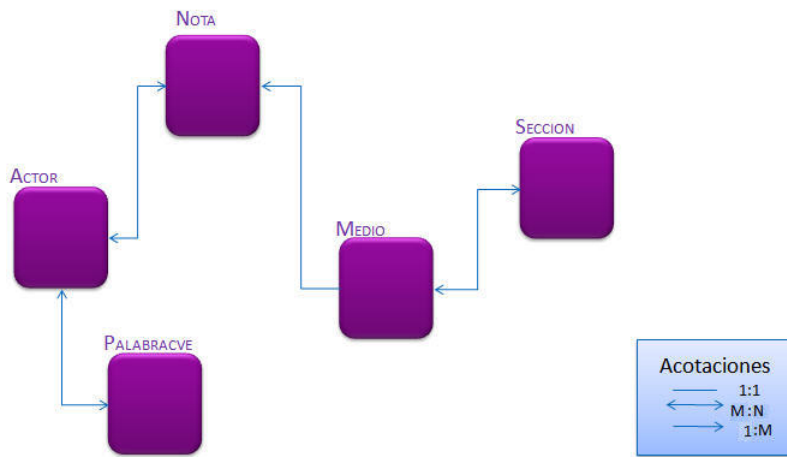


Figura 7.5: Diagrama entidad-relación

Diccionario de datos

USUARIO

- usuario. Persona que bajo un nivel jerárquico ingresa al sistema.
- usu_cve. Clave del usuario autoincrementable que enumera las personas que tienen permiso de ingresar al sistema.
- usr_nmbre. Nombre completo del usuario que desea ingresar al sistema.
- usr_login. Nombre abreviado con el que un usuario se identifica para entrar al sistema.
- usr_psw. Conjunto de caracteres alfanuméricos que permite validar el ingreso del usuario.
- usr_status. Caracter que indica si el usuario está activo o si ya fue dado de baja, sólo toma los valores 'A' o 'B'.
- usr_mail. La dirección de correo electrónico del usuario.
- usr_tipo. Es un valor entero que indica el nivel del usuario según la siguiente tabla:

Tipo	Nivel
1	Administrador
2	Coordinador
3	Analista
4	Consultor

ACTOR

- actor. Es un tema de interés para los usuarios.
- act_cve. Número entero autoincrementable que permite llevar el conteo de los temas que se dan de alta.
- act_nmbre. Descripción del tema de interés.
- act_status. Caracter que indica si el tema está activo o si ya fue dado de baja. Sólo toma los valores 'A' o 'B'.

MEDIO

- medio. Diarios electrónicos que son de interés para buscar información.
- med_cve. Número entero autoincrementable que permite llevar el conteo de los medios que se dan de alta.
- med_nmbre. Descripción del diario en línea que se registra.
- med_status. Caracter que indica si el medio está activo o si ya fue dado de baja, sólo toma los valores 'A' o 'B'.

SECCION

- *seccion*. Es una subdivisión del diario electrónico que permite clasificar las notas por su contenido.
- *sec_cve*. Número entero autoincrementable que permite llevar el conteo de las secciones que se dan de alta.
- *sec_nombre*. Descripción de la sección.
- *sec_status*. Caracter que indica si la sección esta activa o si ya fue dada de baja, sólo toma los valores 'A' o 'B'.

PALABRACVE

- *palabra clave*. Es una palabra que ayuda a identificar si el tema de interés está contenido en la nota.
- *pal_cve*. Número entero autoincrementable que permite llevar el conteo de las palabras clave que se dan de alta.
- *pal_nombre*. Cadena que forma la palabra clave.
- *pal_status*. Caracter que indica si la palabra clave está activa o si ya fue dada de baja, sólo toma los valores 'A' o 'B'.

NOTA

- *nota*. Es una noticia redactada o publicada por un medio periodístico.
- *nta_cve*. Número entero autoincrementable que permite llevar el conteo de las nota que se registran en la base de datos.
- *nta_fecha*. Fecha de la inserción de la noticia.
- *nta_titulo*. Título de la nota.
- *nta_autor*. Agencia o persona que publica o redacta la nota.
- *nta_texto*. Contenido de toda la nota.

Interfaz base

La interfaz está realizada en html pero el estilo fue realizado con hojas de estilo (CSS), los colores fueron propuestos a los usuarios y aceptados, de manera que no se modificaron de la propuesta original.

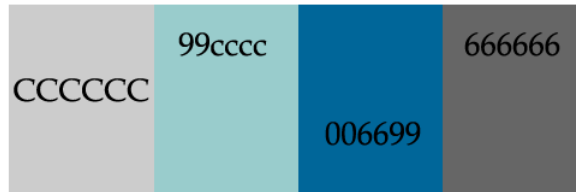


Figura 7.6: Códigos hexadecimales de los colores básicos para el sistema



Figura 7.7: Apariencia base

Manual de Usuario

El proceso inicia con el ingreso al sistema, de acuerdo al usuario y a la contraseña se presenta uno de los siguientes casos:

- El usuario es un administrador del sistema
- El usuario es un coordinador
- El usuario es un analista

Administrador

El administrador del sistema ingresa a un menú en el cual se tienen 3 pestañas:

- Inicio. En la pestaña de inicio se encuentra una descripción del perfil del usuario y un correo electrónico de contacto entre el administrador y el desarrollador. (Fig. 7.9)

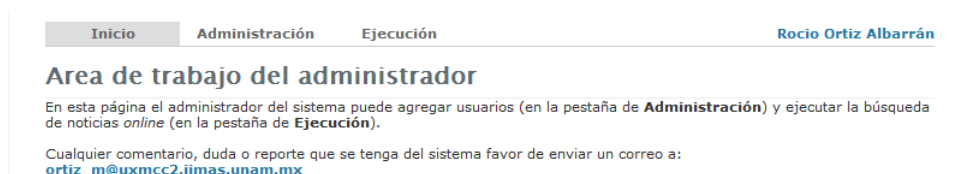


Figura 7.8: Descripción del perfil de usuario

- Administración. En la pestaña de administración se encuentra la sección del sistema que permite gestionar a los usuarios que pueden manipular alguna parte del sistema. La gestión del sistema se realiza a través de cuatro acciones principales, a través de estas se tiene un buen control sobre los usuarios y sobre las funciones que podrán realizar a lo largo del proceso.
 - Alta de usuario. Se ingresan los datos del usuario (Fig. 7.10): nombre (nombre completo del usuario), usuario (nombre corto que permite iniciar sesión) y contraseña (palabra corta que ayuda a identificar que el usuario no sea suplantado) y correo electrónico (dirección de cualquier correo que pertenezca al usuario que se desea agregar).

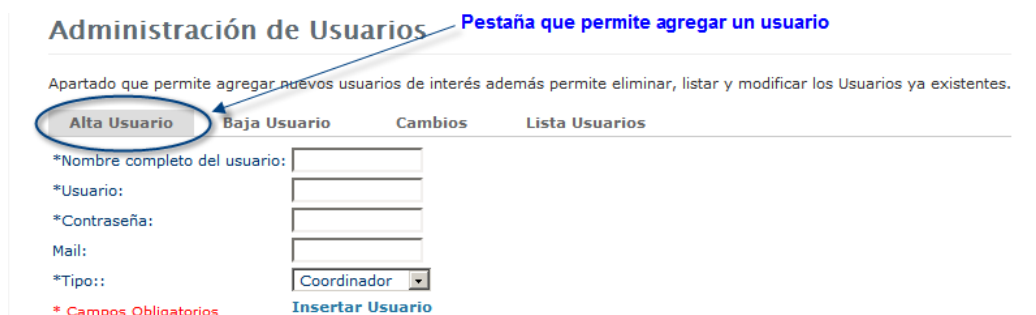


Figura 7.9: Alta de usuario

Además se debe seleccionar el tipo de usuario que se esta agregando (Fig. 7.11), inicialmente solo existen 4 opciones: administrador, coordinador, analista o consultor.

The screenshot shows the 'Alta Usuario' form with the following fields: '*Nombre completo del usuario:' (filled with 'María del Rocío Ortiz'), '*Usuario:' (filled with 'rooio'), '*Contraseña:' (filled with '*****'), 'Mail:' (filled with 'rooio.ortiz.a@gmail.c'), and '*Tipo:'. The dropdown menu for '*Tipo:' is open, showing options: 'Seleccionar', 'Administrador', 'Coordinador' (highlighted), 'Analista', and 'Consultor'. A red asterisk and the text '* Campos Obligatorios' are visible at the bottom left.

Figura 7.10: Selección del tipo de usuario

Una vez que se han llenado todos los campos obligatorios con datos válidos se presiona el texto "Insertar usuario" (Fig. 7.12) y la información quedará registrada en la base de datos.

This screenshot is identical to Figure 7.10, but with the 'Insertar Usuario' button at the bottom highlighted with a blue oval. A blue arrow points from the text 'Botón que permite agregar al usuario' to the button.

Figura 7.11: Botón para dar de lata un usuario.

- Baja de usuario.

The screenshot shows the 'Administración de Usuarios' page. At the top, there are four tabs: 'Alta Usuario', 'Baja Usuario' (highlighted with a blue oval), 'Cambios', and 'Lista Usuarios'. Below the tabs is a text description: 'Apartado que permite agregar nuevos usuarios de interés además permite eliminar, listar y modificar los Usuarios ya existentes.' Below this is a dropdown menu labeled 'Usuario:' with 'Seleccionar' selected. A blue arrow points from the text 'Pestaña que permite dar de baja un usuario activo de cualquier tipo' to the 'Baja Usuario' tab. Below the dropdown is the text 'Baja de Usuario'.

Figura 7.12: Sección de baja de usuarios

Se selecciona el nombre del usuario que se desea eliminar, en la lista se despliegan todos los usuarios activos (Fig. 7.14).



Figura 7.13: Selección de un usuario para ser eliminado

Una vez que se ha seleccionado un usuario se presiona el botón “Baja de Usuario” (Fig. 7.15) y el usuario eliminado ya no podrá ingresar al sistema.

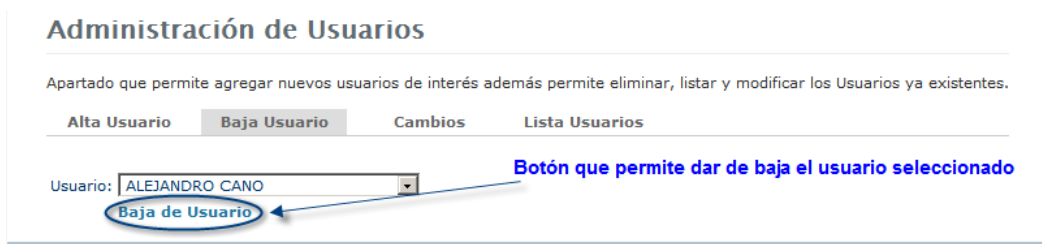


Figura 7.14: Botón que permite dar de baja a un usuario

- Modificaciones de los datos y permisos del usuario (Fig. 7.16).

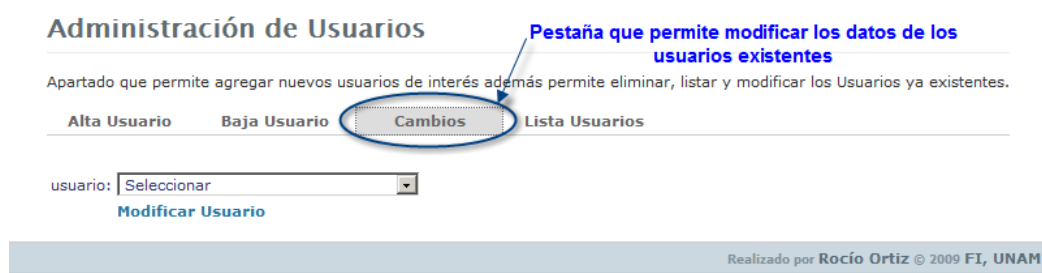


Figura 7.15: Sección de modificaciones a usuarios

En esta interfaz se tiene una lista de todos los usuarios activos y se selecciona el usuario del cual se deben modificar sus características (Fig. 7.17).

Administración de Usuarios

Apartado que permite agregar nuevos usuarios de interés

Alta Usuario Baja Usuario **Cambios**

usuario:

- Seleccionar
- ROCIO ORTIZ ALBARRAN
- JULIO ARMANDO SANTILLAN PEREZ
- ALEJANDRO CANO
- JUAN GONZALEZ LOPEZ
- ARIADNA
- MÓNICA NÚÑEZ RIVERA
- MARTHA CERVANTES
- BERENICE MARTINEZ
- YARA CITLALI DE LA FUENTE
- ALEJANDRO RODRIGUEZ
- LORENA JIMENEZ
- MARISOL AGUILAR NICASIO

Administración de Usuarios

Apartado que permite agregar nuevos usuarios de interés

Alta Usuario Baja Usuario **Cambios**

usuario:

Modificar Usuario

Botón que despliega los datos del usuario seleccionado para ser modificados

Figura 7.16: Proceso de modificación de un usuario

Una vez seleccionado un usuario se presiona el botón “Modificar Usuario” y se presenta un formulario donde se pueden modificar todos los datos mostrados (Fig. 7.18).

Administración de Usuarios

Apartado que permite agregar nuevos usuarios de interés además permite eliminar, listar y modificar los Usuarios ya existentes.

Alta Usuario Baja Usuario **Cambios** Lista Usuarios

nombre

login

psw

status

mail

Actualizar Palabra

Boton que permite guardar los cambios realizados en la base de datos

Realizado por Rocío Ortiz © 2009 FI, UNAM

Figura 7.17: Datos del usuario que se desea actualizar

De esta manera se actualizan los datos del usuario elegido en la base de datos.

- Listado de los usuarios existentes. Permite visualizar una lista de todos los usuarios activos que pueden ingresar al sistema (Fig. 7.19).

Administración de Usuarios

Apartado que permite agregar nuevos usuarios de interés además permite eliminar, listar y modificar los Usuarios ya existentes.

Alta Usuario Baja Usuario Cambios **Lista Usuarios**

Usuario
1 ROCIO ORTIZ ALBARRAN
2 JULIO ARMANDO SANTILLAN PEREZ
3 ALEJANDRO CANO
8 BERENICE MARTINEZ

Figura 7.18: Listado de los usuarios activos del sistema

- Ejecución . Ver la descripción de este proceso en el apartado *Ejecución* del nivel Coordinador.

Coordinador

El coordinador del sistema ingresa a un menú en el cual se tienen 3 pestañas principales: inicio, administración y ejecución, las cuales brindan las opciones para realizar las tareas que se le han asignado.

- Inicio. En la pestaña de inicio se encuentra una descripción del perfil del usuario y un correo electrónico de contacto entre el coordinador y el desarrollador (Fig. 7.20).



Figura 7.19: Descripción del perfil de coordinador

- Administración. En la pestaña de administración (Fig. 21) se tienen opciones para gestionar los temas, las palabras clave, los medios y las secciones de los medios.



Figura 7.20: Sección de administración

- Temas. Al seleccionar la pestaña de **temas** se despliegan las 4 opciones disponibles para la gestión de los temas de interés de un consultor (Fig. 7.22).



Figura 7.21: Gestión de temas de interés

- o Alta de un tema. Al seleccionar esta pestaña se presenta un formulario en el cual es necesario escribir el nombre del tema de interés (Fig. 7.23), una vez que está escrito se presiona el botón **Insertar Tema** y el tema queda registrado en la base de datos. Una vez presionado el botón el sistema muestra un letrero con la información registrada y está listo para dar de alta un nuevo tema.

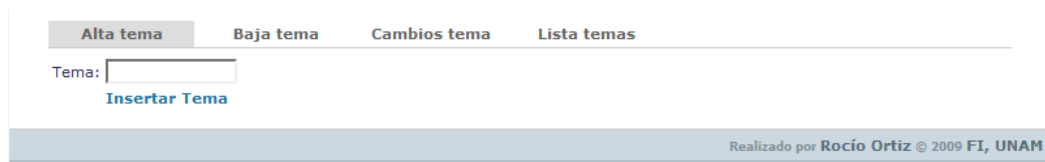


Figura 7.22: Inserción de un nuevo tema

- o Baja de un tema. Una vez que se ha seleccionado la pestaña que da de baja un tema se muestra un combo con todos los temas activos, para eliminar alguno de los tópicos mostrados debe ser seleccionado de la lista y se debe presionar el botón de baja (Fig. 7.24), al presionarlo el tema queda dado de baja, el sistema presenta un letrero con la información de la transacción y queda listo para dar de baja un nuevo tema.

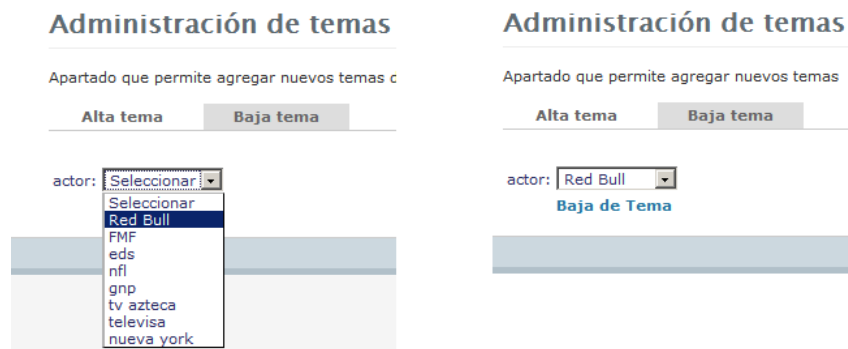


Figura 7.23: Proceso de alta de un tema

- o Cambio de un tema. Al seleccionar esta pestaña se presenta un combo en el cual se listan todos los temas (Fig. 7.25), al seleccionar uno en particular y presionar el botón para modificar el tema se muestra un formulario con los datos del tema dado de alta para ser modificados, una vez que los datos deseados ya fueron corregidos se presiona el botón de actualización del tema y la información queda actualizada en la base de datos; el sistema muestra una leyenda de confirmación del registro y queda listo para actualizar un nuevo tema.

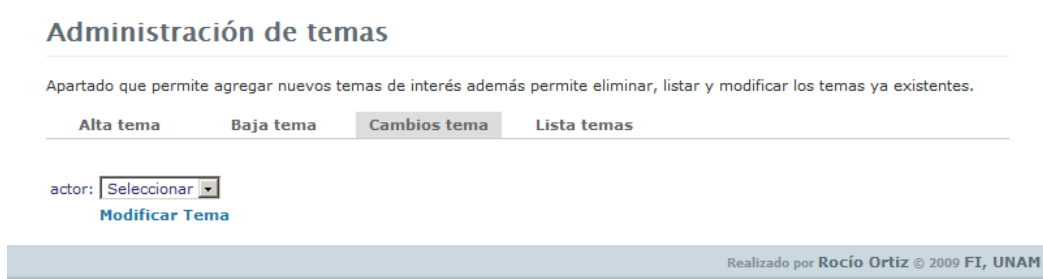


Figura 7.24: Sección de actualización de las características de un tema



Figura 7.25: Proceso de actualización de los temas

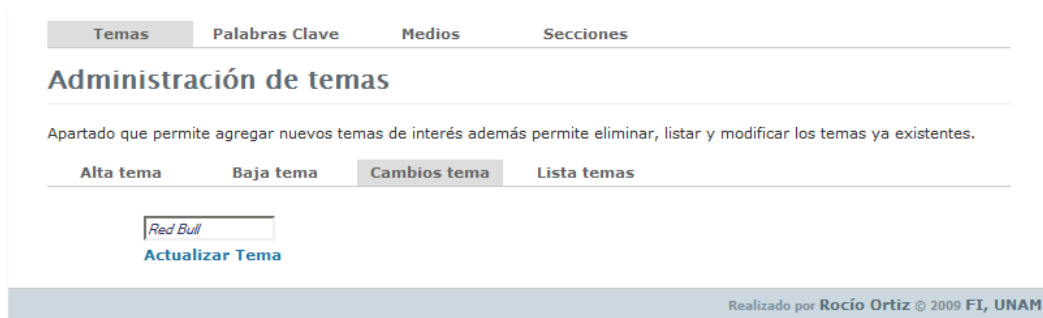


Figura 7.26: Botón que actualiza las características del tema

- Lista de los temas. Al seleccionar la pestaña del listado de temas se muestra una tabla con los temas activos del sistema (Fig. 7.28).

Administración de temas

Apartado que permite agregar nuevos temas de interés además permite eliminar, listar y modificar los temas ya existentes.

Alta tema Baja tema Cambios tema **Lista temas**

TEMA
1 Red Bull
2 FMF
3 eds
9 nfi
10 gnp
11 tv azteca
14 televisa
16 nueva york

Realizado por Rocio Ortiz © 2009 FI, UNAM

Figura 7.27: Listado de temas activos

- Palabras Clave. Al seleccionar la pestaña de palabras clave se pueden gestionar las palabras clave de cada tema dado de alta (Fig. 7.29), si el tema no existe es necesario dar de alta el tema primero ya que las palabras clave son asociadas a los temas.

Red ART2
Procesamiento digital de noticias.

Inicio **Administración** Ejecución Rocio Ortiz AlbarránVar: 2---

Sección de Administración

A través de este apartado se pueden dar de alta los rubros indicados en las pestañas siguientes: Temas, Palabras Clave, Medios y Secciones. También se pueden dar de baja los elementos ya existentes, modificarlos o listar los elementos activos

Temas **Palabras Clave** Medios Secciones

Administración de palabras clave

Figura 7.28: Gestión de palabras clave

- Alta de una palabra clave. Para dar de alta una nueva palabra clave se selecciona la pestaña de palabras clave, luego se presiona la pestaña de alta y se observa un formulario (Fig. 7.30).



Figura 7.29: Alta de palabras clave

En este formulario se tiene un cuadro de entrada de texto en el cual se escribe la palabra clave nueva y un combo que lista todos los temas dados de alta, se selecciona uno y se presiona el botón **“Insertar palabra”** (Fig. 31). El sistema inserta la palabra, esta ya será buscada cuando se procesen los archivos html y el sistema queda listo para insertar una nueva palabra.

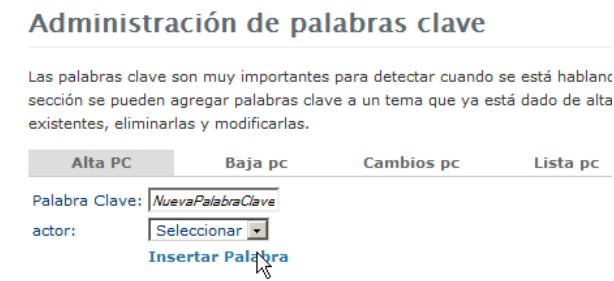


Figura 7.30: Botón que permite añadir una nueva palabra al sistema

- Baja de una palabra clave. Al ingresar a la pestaña de baja se observa un combo en el cual se listan todas las palabras activas. Se selecciona la palabra que se desea dar de baja y se presiona el botón **“Borrar palabra”** (Fig. 7.32). En ese momento la palabra ya no es buscada en el procesamiento de notas y el sistema queda listo para eliminar otra palabra clave.

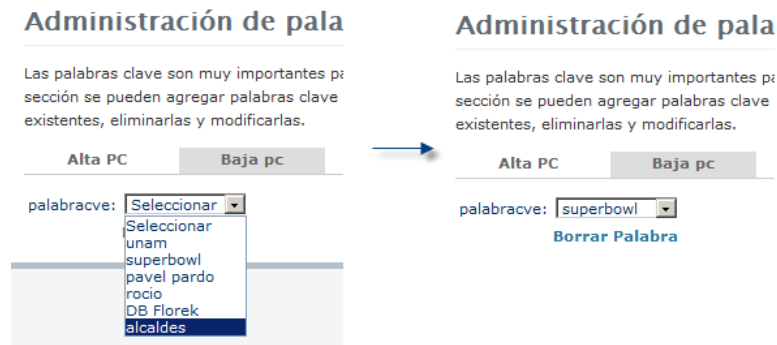


Figura 7.31: Proceso de alta de una palabra clave

- Cambio de una palabra clave. Para cambiar la información de una palabra clave se selecciona la pestaña de cambios (Fig. 7.33).

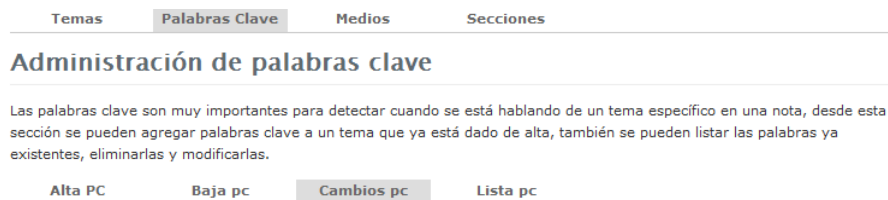


Figura 7.32: Sección de actualización de las características de una palabra clave

Al seleccionar la pestaña de cambio de una palabra clave se muestra un combo en el cual se listan las palabras clave activas, se selecciona aquella que se desea modificar y se presiona el botón de “**Modificar Palabra**” (Fig. 7.34).



Figura 7.33: Proceso de actualización de las palabras clave

Se muestra otro formulario en el que se puede cambiar el texto de la palabra clave o el tema al que esta es asociada. Una vez que la información fue cambiada se presiona el botón “**Actualizar Palabra**” para guardar los cambios

y el sistema queda listo para realizar cambios en otra palabra clave (Fig. 7.35).

Administración de palabras clave

Las palabras clave son muy importantes para detectar cuando se está hablando de un tema específico en una nota, desde esta sección se pueden agregar palabras clave a un tema que ya está dado de alta, también se pueden listar las palabras ya existentes, eliminarlas y modificarlas.

Figura 7.34: Botón que guarda los cambios de las palabras clave

- o Lista de las palabras claves. Al seleccionar la pestaña de Lista se muestra al coordinador una tabla en la que se observan todas las palabras clave activas y el tema al que pertenecen (Fig. 7.36).

Administración de palabras clave

Las palabras clave son muy importantes para detectar cuando se está hablando de un tema específico en una nota, desde esta sección se pueden agregar palabras clave a un tema que ya está dado de alta, también se pueden listar las palabras ya existentes, eliminarlas y modificarlas.

TEMA	PALABRA CLAVE
FMF	unam
nfl	superbowl
televisa	rocio
televisa	DB Florek
nueva york	alcaldes

Figura 7.35: Lista de las palabras clave y el tema al que pertenecen

- Medios. En la sección de administración de medios se puede dar de alta un nuevo diario en línea esto permite asociar las notas al medio electrónico del cual fueron extraídas (Fig. 7.37).

.: Red ART2 .:

Procesamiento digital de noticias.

Figura 7.36: Gestión de medios

- Alta de un medio. Para agregar un nuevo medio se selecciona la pestaña de alta (Fig. 7.38).

Sección de Administración

A través de este apartado se pueden dar de alta los rubros indicados en las pestañas siguientes: Temas, Palabras Clave, Medios y Secciones. También se pueden dar de baja los elementos ya existentes, modificarlos o listar los elementos activos

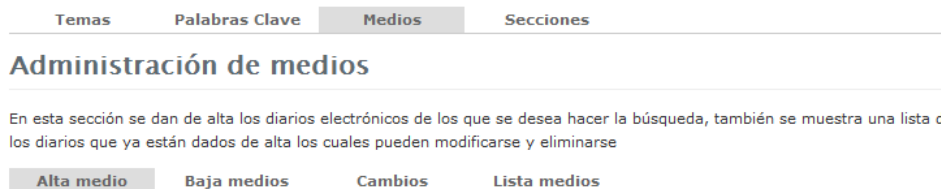


Figura 7.37: Sección de alta de medios

Se presenta un formulario en el cual existe un campo de texto, en este campo se escribe el nombre del diario y se presiona el botón de “**Insertar Medio**” (Fig. 7.39), de esta manera el diario electrónico queda registrado y el sistema está listo para agregar un nuevo medio.

Administración de medios

En esta sección se dan de alta los diarios los diarios que ya están dados de alta los cuales



Figura 7.38: Boton que permite registrar un nuevo medio

- Baja de un medio. Al seleccionar la pestaña de baja se muestra un combo que lista todos los medios activos (Fig. 7.40).

Sección de Administración

A través de este apartado se pueden dar de alta los rubros indicados en las pestañas siguientes: Temas, Palabras Clave y Secciones. También se pueden dar de baja los elementos ya existentes, modificarlos o listar los elementos activos



Figura 7.39: Sección que permite eliminar un medio activo

Se selecciona uno de los medios y se presiona el botón “**Borrar Palabra**” (Fig. 7.41), de esta manera se desactiva el uso del medio y el sistema queda

listo para eliminar otro medio electrónico.



Figura 7.40: Proceso para eliminar un medio

- o Cambio de un medio. Se selecciona la pestaña de cambios (Fig. 7.42).

Administración de medios

En esta sección se dan de alta los diarios electrónicos de los que se desea hacer la búsqueda, también se muestra una lista de los diarios que ya están dados de alta los cuales pueden modificarse y eliminarse

Alta medio Baja medios **Cambios** Lista medios

Figura 7.41: Sección para cambios en las características de los medios

Se muestra una lista de todos los medios activos y se selecciona uno. Se presiona el botón de “**Modificar Medio**” (Fig. 7.43).

Administración de medios

En esta sección se dan de alta los diarios electrónicos de los que ya están dados de alta los cuales pueden modificar

Alta medio Baja medios **Cambios**

medio: [Modificar Medio](#)

- Seleccionar
- Reforma
- El Universal
- La Jornada
- Milenio Diario
- El Economista
- El Financiero
- La Crónica
- Excelsior
- Expansion
- Yahoo
- Crain's
- Google
- Macroeconomía**
- Uno más uno
- Excelsior
- Noroeste
- Debate(Noroeste)
- El Sol de México
- bravo

Administración de medios

En esta sección se dan de alta los diarios electrónicos de los que ya están dados de alta los cuales pueden modificar

Alta medio Baja medios **Cambios**

medio: [Modificar Medio](#)

Figura 7.42: Proceso de actualización de medios

Se muestra un formulario en el cual se modifica el nombre del medio, una vez que se ha cambiado la información se presiona el botón de “**Actualizar Palabra**” (Fig. 7.44) y la información queda actualizada en la base de datos. El sistema queda listo para actualizar otro medio electrónico.

Administración de medios

En esta sección se dan de alta los diarios electrónicos de los que ya están dados de alta los cuales pueden modificar

Alta medio Baja medios **Cambios**

med_nombre [Actualizar Palabra](#)

Figura 7.43: Botón que guarda los cambios de los medios en el sistema

- Lista de los medios. En la pestaña de Lista de medios se observa una tabla con los medios activos en el sistema (Fig. 7.45).

Temas Palabras Clave **Medios** Secciones

Administración de medios

En esta sección se dan de alta los diarios electrónicos de los que se desea hacer la búsqueda los diarios que ya están dados de alta los cuales pueden modificarse y eliminarse

Alta medio Baja medios Cambios **Lista medios**

MEDIO
1 Reforma
2 El Universal
3 La Jornada
4 La Crónica
7 El Financiero
8 El Economista
9 Milenio Diario
11 Excelsior
12 Expansion
13 Yahoo
15 Crain's
16 Google
17 Macroeconomía
18 Uno más uno
19 Excelsior
20 Noroeste
21 Debate(Noroeste)
24 Cambio (sonora)
25 Expresso (sonora)

Figura 7.44: Listado de los medios activos en el sistema

- Secciones. Para dar de alta una sección se requiere que el medio electrónico al que esta pertenece ya este registrado en el sistema. Permite establecer el contexto en el cual se publicó la nota de interés.
 - Alta de una sección. Se selecciona la pestaña de alta (Fig. 7.46).

Sección de Administración

A través de este apartado se pueden dar de alta los rubros indicados en las pestañas siguientes: Temas, Palabras Clave, Medios y Secciones. También se pueden dar de baja los elementos ya existentes, modificarlos o listar los elementos activos

Temas Palabras Clave Medios **Secciones**

Administrar secciones

Cada diario en línea tiene diferentes secciones en las cuales se puede clasificar el contenido de una noticia, esto es de interés para el consultor por lo que es necesario dar de alta las secciones asociadas con cada diario, de igual manera se puede eliminar una sección existente, modificarla o listar los registros habilitados en la base de datos.

Alta Seccion Baja Seccion Cambios **Lista Seccion**

Figura 7.45: Gestión de secciones

Se muestra un formulario con dos elementos, un campo de texto en el cual se deberá escribir el nombre de la sección y un combo en el que se listan todos los medio electrónicos activos (Fig. 7.47).

Administrar secciones

Cada diario en línea tiene diferentes secciones en las cuales se puede clasificar el contenido de una noticia, esto es de interés para el consultor por lo que es necesario dar de alta las secciones asociadas con cada diario, de igual manera se puede eliminar una sección existente, modificarla o listar los registros habilitados en la base de datos.



Figura 7.46: Formulario que permite agregar una sección a un medio

Se escribe el nombre que se desea dar de alta, se selecciona el medio al cual pertenece.

Administrar secciones

Cada diario en línea tiene diferentes secciones en las cuales se puede clasificar el contenido de una noticia, esto es de interés para el consultor por lo que es necesario dar de alta las secciones asociadas con cada diario, de igual manera se puede eliminar una sección existente, modificarla o listar los registros habilitados en la base de datos.

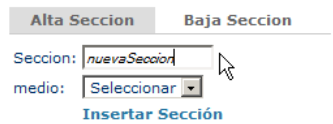


Figura 7.47: Proceso de alta de una sección

Se presiona el botón “Insertar Sección” (Fig. 7.49).

Administrar secciones

Cada diario en línea tiene diferentes secciones en las cuales se puede clasificar el contenido de una noticia, esto es de interés para el consultor por lo que es necesario dar de alta las secciones asociadas con cada diario, de igual manera se puede eliminar una sección existente, modificarla o listar los registros habilitados en la base de datos.



Figura 7.48: Botón que permite guardar los datos en el sistema

La sección queda registrada para uso del sistema y el formulario está listo para dar de alta otra sección.

- o Baja de una sección. Se selecciona la pestaña de baja de sección (Fig. 7.50).

Sección de Administración

A través de este apartado se pueden dar de alta los rubros indicados en las pestañas siguientes: Temas, Palabras Clave, Medios y Secciones. También se pueden dar de baja los elementos ya existentes, modificarlos o listar los elementos activos

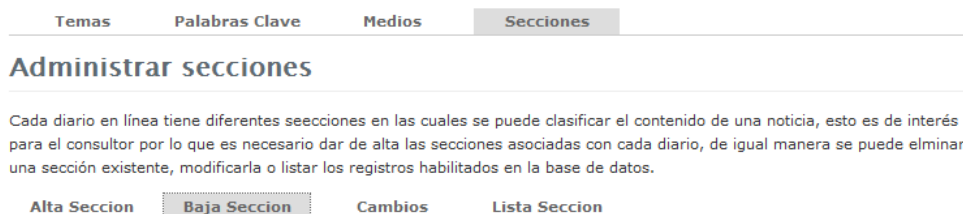


Figura 7.49: Sección de baja de secciones

Se presenta un combo con todas las secciones dadas de alta y el medio al que pertenece. Una vez que se elige la sección que se desea dar de baja se presiona el botón “**Borrar**” (Fig. 7.51)

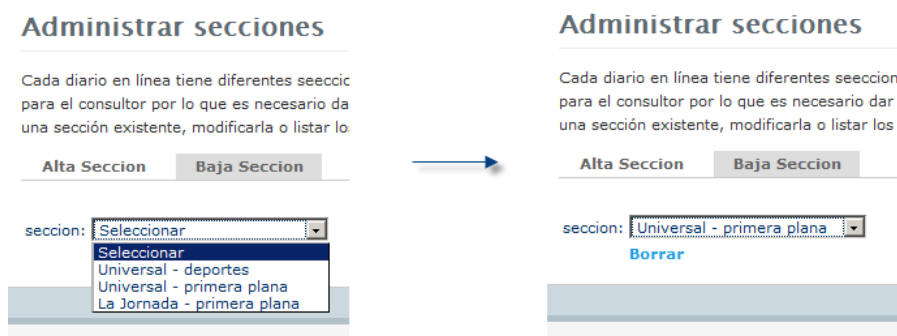


Figura 7.50: Proceso de baja de una sección

La sección queda eliminada y el sistema está listo para eliminar otra sección.

- ◇ Cambio de una sección. Para cambiar los datos de una sección se presiona la pestaña de cambios (Fig. 7.52).

Sección de Administración

A través de este apartado se pueden dar de alta los rubros indicados en las pestañas siguientes: Temas, Palabras Clave, Medios y Secciones. También se pueden dar de baja los elementos ya existentes, modificarlos o listar los elementos activos

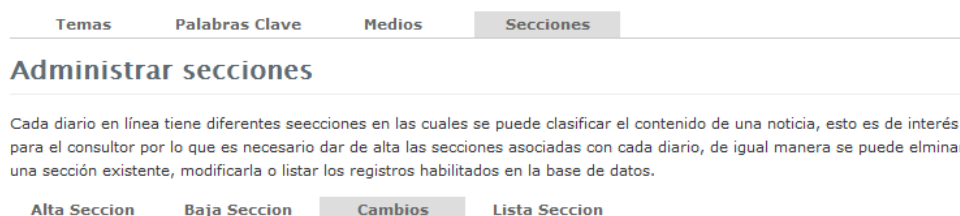


Figura 7.51: Actualización de datos de las secciones

Se muestra una lista de las secciones activas en el sistema. Se elige la sección que se desea modificar y se presiona el botón “**Modificar**” (Fig. 7.53)



Figura 7.52: Selección de una sección para actualizar

Se muestra un formulario en el que se puede modificar el nombre de la sección y un combo en el que se puede modificar el medio electrónico al que pertenece, una vez que se han actualizado los datos se presiona el boton “**Actualizar**” (Fig. 7.54). El sistema guarda los cambios en la base de datos y queda listo para modificar otra sección.

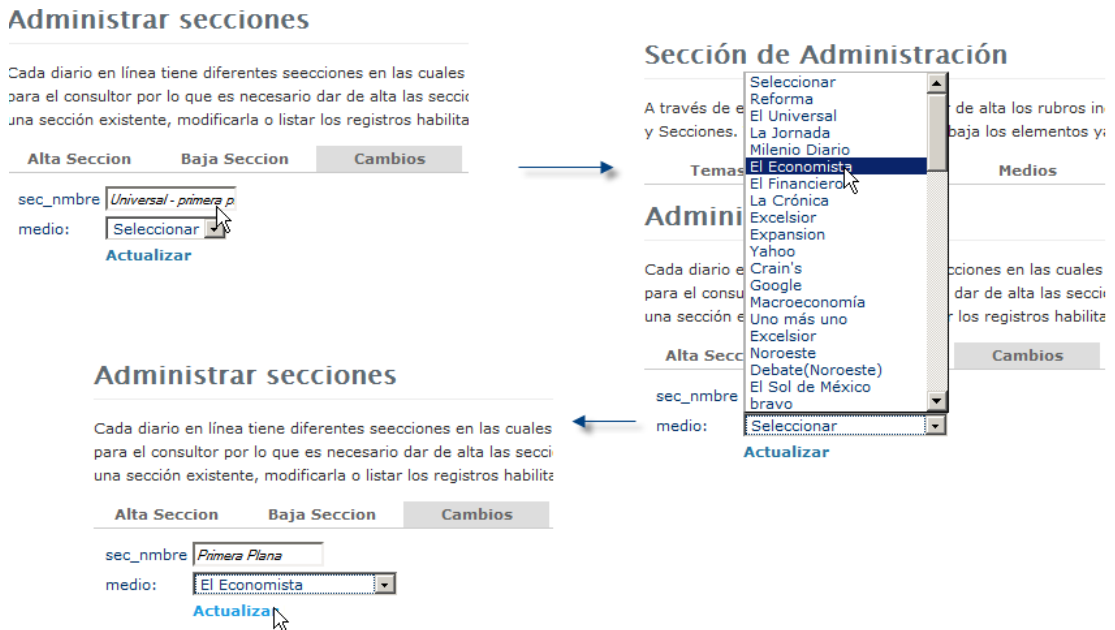


Figura 7.53: Proceso de actualización de los datos de una sección

- o Lista de las secciones. Se presiona la pestaña de Lista y se muestra una tabla en la que se observan las secciones activas en el sistema junto con el diario al que pertenecen (Fig. 7.55).

Administrar secciones

Cada diario en línea tiene diferentes secciones en las cuales se puede clasificar el contenido para el consultor por lo que es necesario dar de alta las secciones asociadas con cada una sección existente, modificarla o listar los registros habilitados en la base de datos.

Alta Seccion	Baja Seccion	Cambios	Lista Seccion
MEDIO	SECCIÓN		
El Universal	deportes		
El Universal	primera plana		

Figura 7.54: Listado de las secciones activas y el diario al que pertenecen

- Ejecución (Fig. 7.56). Para procesar o entrenar a la red ART con un conjunto de archivos html se selecciona la pestaña de ejecución, se muestran entonces 3 opciones: Entrenamiento, Procesamiento y Consulta.

Red ART 2

Procesamiento digital de noticias.

Inicio	Administración	Ejecución	Rocio Ortiz Albarrán
--------	----------------	-----------	----------------------

Figura 7.55: Sección de ejecución de la red ART' 2

- Entrenamiento. Se selecciona esta pestaña si se desea agregar un nuevo patrón al proceso de búsqueda de notas (Fig. 7.57). El entrenamiento sólo es necesario cuando un diario modifique su estructura web básica. Al seleccionar esta pestaña se muestra un formulario en el cual se listan todos los diarios dados de alta en la base de datos.

Sección de procesamiento

Clasificación del archivo

Entrenamiento	Procesamiento	Consulta
<input type="checkbox"/> Marcar/Desmarcar Todos <input type="checkbox"/> Marcar/Desmarcar Principales		
<input type="checkbox"/> Reforma	<input type="checkbox"/> El Universal	<input type="checkbox"/> La Jornada
<input type="checkbox"/> El Financiero	<input type="checkbox"/> La Crónica	<input type="checkbox"/> Excelsior
<input type="checkbox"/> Líderes Mexicanos	<input type="checkbox"/> Crain's	<input type="checkbox"/> Google
<input type="checkbox"/> Excelsior	<input type="checkbox"/> Noroeste	<input type="checkbox"/> Debate(Noroeste)
<input type="checkbox"/> Cambio (sonora)	<input type="checkbox"/> Expreso (sonora)	<input type="checkbox"/> El imparcial (sonora)
<input type="checkbox"/> Vanguardia (Coahuila)	<input type="checkbox"/> Palabra (Coahuila)	<input type="checkbox"/> Heraldo de Saltillo
<input type="checkbox"/> El Norte (Nuevo León)	<input type="checkbox"/> El Mañana de Nuevo L	<input type="checkbox"/> El Mañana de Reynosa
<input type="checkbox"/> Ocho Columnas (Jalis)	<input type="checkbox"/> Público (Jalisco)	<input type="checkbox"/> El Informador (Jalis)
<input type="checkbox"/> Enfoque (Tepic)	<input type="checkbox"/> AM de León (Guanajua)	<input type="checkbox"/> Heraldo de León (Gua)
<input type="checkbox"/> Hidrocálido (Aguasca)	<input type="checkbox"/> Pulso San Luis Potos	<input type="checkbox"/> Imagen de Zacatecas
<input type="checkbox"/> Diario de Chiapas	<input type="checkbox"/> Tiempo de Oaxaca	<input type="checkbox"/> El imparcial de Oaxa
<input type="checkbox"/> El Sol de Chilpancing	<input type="checkbox"/> Diario de Yucatán	<input type="checkbox"/> Novedades de Quintan
<input type="checkbox"/> Presente (Tabasco)	<input type="checkbox"/> Diario Olmeca (Tabas)	<input type="checkbox"/> Imagen de Veracruz
<input type="checkbox"/> AZ de Xalapa (Veracr)	<input type="checkbox"/> La Opinión de Poza R	<input type="checkbox"/> Liberal del Sur (Ver)
<input type="checkbox"/> Diario de Morelos	<input type="checkbox"/> ABC de Toluca	<input type="checkbox"/> El Diario de Toluca
<input type="checkbox"/> Milenio Diario <input type="checkbox"/> Expansion <input type="checkbox"/> Macroeconomía <input type="checkbox"/> El Sol de México <input type="checkbox"/> Zócalo (Coahuila) <input type="checkbox"/> La Voz de Durango <input type="checkbox"/> El Bravo (Tamaulipas) <input type="checkbox"/> Mural (Jalisco) <input type="checkbox"/> Correo de hoy (Guana) <input type="checkbox"/> Cuarto Poder de Chia <input type="checkbox"/> Diario 17 de Guerrer <input type="checkbox"/> Por Esto de Mérida <input type="checkbox"/> Dictamen (Veracruz) <input type="checkbox"/> Síntesis de Puebla		
<input type="checkbox"/> El Economista <input type="checkbox"/> Yahoo <input type="checkbox"/> Uno más uno <input type="checkbox"/> bravo <input type="checkbox"/> El diario de Coahuil <input type="checkbox"/> Milenio Monterrey <input type="checkbox"/> La Voz de Michoacán <input type="checkbox"/> Diario <input type="checkbox"/> Noticias de Querétar <input type="checkbox"/> El Orbe de Tapachula <input type="checkbox"/> El Sur (Guerrero) <input type="checkbox"/> Tabasco Hoy <input type="checkbox"/> Notiver (Veracruz) <input type="checkbox"/> Unión de Morelos		
<input type="button" value="Entrenar"/>		

Figura 7.56: Sección de entrenamiento de la red ART'2

En este formulario se debe seleccionar el diario que se desea procesar, pueden seleccionarse varios diarios de manera manual (Fig. 7.58).

Sección de procesamiento

Clasificación del archivo

Entrenamiento	Procesamiento	Consulta
<input type="checkbox"/> Marcar/Desmarcar Todos <input type="checkbox"/> Marcar/Desmarcar Principales		
<input type="checkbox"/> Reforma	<input checked="" type="checkbox"/> El Universal	<input checked="" type="checkbox"/> La Jornada
<input type="checkbox"/> El Financiero	<input type="checkbox"/> La Crónica	<input type="checkbox"/> Excelsior
<input type="checkbox"/> Líderes Mexicanos	<input type="checkbox"/> Crain's	<input type="checkbox"/> Google
<input type="checkbox"/> Excelsior	<input type="checkbox"/> Noroeste	<input type="checkbox"/> Debate(Noroeste)
<input type="checkbox"/> Cambio (sonora)	<input type="checkbox"/> Expreso (sonora)	<input type="checkbox"/> El imparcial (sonora)
<input type="checkbox"/> Vanguardia (Coahuila)	<input type="checkbox"/> Palabra (Coahuila)	<input type="checkbox"/> Heraldo de Saltillo
<input type="checkbox"/> El Norte (Nuevo León)	<input type="checkbox"/> El Mañana de Nuevo L	<input type="checkbox"/> El Mañana de Reynosa
<input type="checkbox"/> Ocho Columnas (Jalis)	<input type="checkbox"/> Público (Jalisco)	<input type="checkbox"/> El Informador (Jalis)
<input type="checkbox"/> Enfoque (Tepic)	<input type="checkbox"/> AM de León (Guanajua)	<input type="checkbox"/> Heraldo de León (Gua)
<input type="checkbox"/> Hidrocálido (Aguasca)	<input type="checkbox"/> Pulso San Luis Potos	<input type="checkbox"/> Imagen de Zacatecas
<input type="checkbox"/> Diario de Chiapas	<input type="checkbox"/> Tiempo de Oaxaca	<input type="checkbox"/> El imparcial de Oaxa
<input type="checkbox"/> El Sol de Chilpancin	<input type="checkbox"/> Diario de Yucatán	<input type="checkbox"/> Novedades de Quintan
<input type="checkbox"/> Presente (Tabasco)	<input type="checkbox"/> Diario Olmeca (Tabas)	<input type="checkbox"/> Imagen de Veracruz
<input type="checkbox"/> AZ de Xalapa (Veracr)	<input type="checkbox"/> La Opinión de Poza R	<input type="checkbox"/> Liberal del Sur (Ver)
<input type="checkbox"/> Diario de Morelos	<input type="checkbox"/> ABC de Toluca	<input type="checkbox"/> El Diario de Toluca
<input type="button" value="Entrenar"/>		

Figura 7.57: Selección de diarios a entrenar de manera manual

Si van a entrenarse todos los diarios se puede seleccionar la opción “**Marcar/Desmarcar Todos**” (Fig. 7.59). Si ya no se desea seleccionar todos los diarios sino uno o varios de manera manual se pueden desmarcar los diarios si se presiona nuevamente la casilla “**Marcar/Desmarcar Todos**”

<input checked="" type="checkbox"/> Marcar/Desmarcar Todos				
<input checked="" type="checkbox"/> Marcar/Desmarcar Principales				
<input checked="" type="checkbox"/> Reforma	<input checked="" type="checkbox"/> El Universal	<input checked="" type="checkbox"/> La Jornada	<input checked="" type="checkbox"/> Milenio Diario	<input checked="" type="checkbox"/> El Economista
<input checked="" type="checkbox"/> El Financiero	<input checked="" type="checkbox"/> La Crónica	<input checked="" type="checkbox"/> Excelsior	<input checked="" type="checkbox"/> Expansion	<input checked="" type="checkbox"/> Yahoo
<input checked="" type="checkbox"/> Líderes Mexicanos	<input checked="" type="checkbox"/> Crain's	<input checked="" type="checkbox"/> Google	<input checked="" type="checkbox"/> Macroeconomía	<input checked="" type="checkbox"/> Uno más uno
<input checked="" type="checkbox"/> Excelsior	<input checked="" type="checkbox"/> Noroeste	<input checked="" type="checkbox"/> Debate(Noroeste)	<input checked="" type="checkbox"/> El Sol de México	<input checked="" type="checkbox"/> bravo
<input checked="" type="checkbox"/> Cambio (sonora)	<input checked="" type="checkbox"/> Expreso (sonora)	<input checked="" type="checkbox"/> El imparcial (sonora)	<input checked="" type="checkbox"/> Zócalo (Coahuila)	<input checked="" type="checkbox"/> El diario de Coahuil
<input checked="" type="checkbox"/> Vanguardia (Coahuila)	<input checked="" type="checkbox"/> Palabra (Coahuila)	<input checked="" type="checkbox"/> Heraldo de Saltillo	<input checked="" type="checkbox"/> La Voz de Durango	<input checked="" type="checkbox"/> Milenio Monterrey
<input checked="" type="checkbox"/> El Norte (Nuevo León)	<input checked="" type="checkbox"/> El Mañana de Nuevo L	<input checked="" type="checkbox"/> El Mañana de Reynosa	<input checked="" type="checkbox"/> El Bravo (Tamaulipas)	<input checked="" type="checkbox"/> La Voz de Michoacán
<input checked="" type="checkbox"/> Ocho Columnas (Jalis)	<input checked="" type="checkbox"/> Público (Jalisco)	<input checked="" type="checkbox"/> El Informador (Jalis)	<input checked="" type="checkbox"/> Mural (Jalisco)	<input checked="" type="checkbox"/> Diario
<input checked="" type="checkbox"/> Enfoque (Tepic)	<input checked="" type="checkbox"/> AM de León (Guanajua)	<input checked="" type="checkbox"/> Heraldo de León (Gua)	<input checked="" type="checkbox"/> Correo de hoy (Guana)	<input checked="" type="checkbox"/> Noticias de Querétar
<input checked="" type="checkbox"/> Hidrocálido (Aguasca)	<input checked="" type="checkbox"/> Pulso San Luis Potos	<input checked="" type="checkbox"/> Imagen de Zacatecas	<input checked="" type="checkbox"/> Cuarto Poder de Chia	<input checked="" type="checkbox"/> El Orbe de Tapachula
<input checked="" type="checkbox"/> Diario de Chiapas	<input checked="" type="checkbox"/> Tiempo de Oaxaca	<input checked="" type="checkbox"/> El imparcial de Oaxa	<input checked="" type="checkbox"/> Diario 17 de Guerrer	<input checked="" type="checkbox"/> El Sur (Guerrero)
<input checked="" type="checkbox"/> El Sol de Chilpancin	<input checked="" type="checkbox"/> Diario de Yucatán	<input checked="" type="checkbox"/> Novedades de Quintan	<input checked="" type="checkbox"/> Por Esto de Mérida	<input checked="" type="checkbox"/> Tabasco Hoy
<input checked="" type="checkbox"/> Presente (Tabasco)	<input checked="" type="checkbox"/> Diario Olmeca (Tabas)	<input checked="" type="checkbox"/> Imagen de Veracruz	<input checked="" type="checkbox"/> Dictamen (Veracruz)	<input checked="" type="checkbox"/> Notiver (Veracruz)
<input checked="" type="checkbox"/> AZ de Xalapa (Veracr)	<input checked="" type="checkbox"/> La Opinión de Poza R	<input checked="" type="checkbox"/> Liberal del Sur (Ver)	<input checked="" type="checkbox"/> Síntesis de Puebla	<input checked="" type="checkbox"/> Unión de Morelos
<input checked="" type="checkbox"/> Diario de Morelos	<input checked="" type="checkbox"/> ABC de Toluca	<input checked="" type="checkbox"/> El Diario de Toluca		
<input type="button" value="Entrenar"/>				

Figura 7.58: Selección de los diarios a entrenar a través del boton de marcar/desmarcar

Existe una opción que selecciona únicamente los diarios etiquetados como principales "Marcar/Desmarcar Principales" (Fig. 7.60). Estos son los más conocidos en el D.F y si se desea modificar esta lista es necesario contactar al desarrollador del sistema. Una vez seleccionados los diarios principales se pueden seleccionar más diarios manualmente o deseleccionar alguno de los principales que no requiera entrenamiento. Si ya no se desea seleccionar a todos los diarios principales se presiona nuevamente la casilla "Marcar/Desmarcar Principales" y se pueden seleccionar los diarios de manera manual.



Figura 7.59: Selección que permite entrenar solo los diarios principales

Una vez que se eligieron los diarios que se desean entrenar se debe presionar el botón de Entrenar (Fig. 7.61), este iniciará el proceso de reconocimiento de patrones en los archivos de los diarios indicados.



Figura 7.60: Botón que inicia el entrenamiento

Una vez que el entrenamiento concluye se muestra una leyenda de conclusión del

entrenamiento y una liga para regresar a la pantalla inicial de ejecución (Fig. 7.62).

Sección de procesamiento



Figura 7.61: Pantalla de finalización del entrenamiento

- Procesamiento.

Sección de procesamiento



Figura 7.62: Sección de procesamiento de información

Al igual que en el entrenamiento se tienen diferentes opciones para seleccionar los diarios en los cuales se desea buscar noticias. Puede seleccionar uno o varios diarios de manera manual (Fig. 7.64).

Sección de procesamiento

Clasificación del archivo

Entrenamiento **Procesamiento** Consulta

Marcar/Desmarcar Todos
 Marcar/Desmarcar Principales

<input type="checkbox"/> Reforma	<input checked="" type="checkbox"/> El Universal	<input type="checkbox"/> La Jornada	<input type="checkbox"/> Milenio Diario	<input type="checkbox"/> El Economista
<input type="checkbox"/> El Financiero	<input checked="" type="checkbox"/> La Crónica	<input type="checkbox"/> Excelsior	<input type="checkbox"/> Expansion	<input type="checkbox"/> Yahoo
<input type="checkbox"/> Líderes Mexicanos	<input type="checkbox"/> Crain's	<input type="checkbox"/> Google	<input type="checkbox"/> Macroeconomía	<input type="checkbox"/> Uno más uno
<input type="checkbox"/> Excelsior	<input type="checkbox"/> Noroeste	<input type="checkbox"/> Debate(Noroeste)	<input type="checkbox"/> El Sol de México	<input type="checkbox"/> bravo
<input type="checkbox"/> Cambio (sonora)	<input type="checkbox"/> Expreso (sonora)	<input type="checkbox"/> El imparcial (sonora)	<input type="checkbox"/> Zócalo (Coahuila)	<input type="checkbox"/> El diario de Coahuil
<input type="checkbox"/> Vanguardia (Coahuila)	<input type="checkbox"/> Palabra (Coahuila)	<input type="checkbox"/> Heraldo de Saltillo	<input type="checkbox"/> La Voz de Durango	<input type="checkbox"/> Milenio Monterrey
<input type="checkbox"/> El Norte (Nuevo León)	<input type="checkbox"/> El Mañana de Nuevo L	<input type="checkbox"/> El Mañana de Reynosa	<input type="checkbox"/> El Bravo (Tamaulipas)	<input type="checkbox"/> La Voz de Michoacán
<input type="checkbox"/> Ocho Columnas (Jalis)	<input type="checkbox"/> Público (Jalisco)	<input type="checkbox"/> El Informador (Jalis)	<input type="checkbox"/> Mural (Jalisco)	<input type="checkbox"/> Diario
<input type="checkbox"/> Enfoque (Tepic)	<input type="checkbox"/> AM de León (Guanajua)	<input type="checkbox"/> Heraldo de León (Gua)	<input type="checkbox"/> Correo de hoy (Guana)	<input type="checkbox"/> Noticias de Querétar
<input type="checkbox"/> Hidrocálido (Aguasca)	<input type="checkbox"/> Pulso San Luis Potos	<input type="checkbox"/> Imagen de Zacatecas	<input type="checkbox"/> Cuarto Poder de Chia	<input type="checkbox"/> El Orbe de Tapachula
<input type="checkbox"/> Diario de Chiapas	<input type="checkbox"/> Tiempo de Oaxaca	<input type="checkbox"/> El imparcial de Oaxa	<input type="checkbox"/> Diario 17 de Guerrer	<input type="checkbox"/> El Sur (Guerrero)
<input type="checkbox"/> El Sol de Chilpancín	<input type="checkbox"/> Diario de Yucatán	<input type="checkbox"/> Novedades de Quintan	<input type="checkbox"/> Por Esto de Mérida	<input type="checkbox"/> Tabasco Hoy
<input type="checkbox"/> Presente (Tabasco)	<input type="checkbox"/> Diario Olmeca (Tabas)	<input type="checkbox"/> Imagen de Veracruz	<input type="checkbox"/> Dictamen (Veracruz)	<input type="checkbox"/> Notiver (Veracruz)
<input type="checkbox"/> AZ de Xalapa (Veracr)	<input type="checkbox"/> La Opinión de Poza R	<input type="checkbox"/> Liberal del Sur (Ver)	<input type="checkbox"/> Síntesis de Puebla	<input type="checkbox"/> Unión de Morelos
<input type="checkbox"/> Diario de Morelos	<input type="checkbox"/> ABC de Toluca	<input type="checkbox"/> El Diario de Toluca		

Procesar diarios

Figura 7.63: Selección de los diarios de manera manual

Se pueden marcar o desmarcar todos los diarios dados de alta (Fig. 7.65).

Sección de procesamiento

Clasificación del archivo

Entrenamiento **Procesamiento** Consulta

Marcar/Desmarcar Todos
 Marcar/Desmarcar Principales

<input checked="" type="checkbox"/> Reforma	<input checked="" type="checkbox"/> El Universal	<input checked="" type="checkbox"/> La Jornada	<input checked="" type="checkbox"/> Milenio Diario	<input checked="" type="checkbox"/> El Economista
<input checked="" type="checkbox"/> El Financiero	<input checked="" type="checkbox"/> La Crónica	<input checked="" type="checkbox"/> Excelsior	<input checked="" type="checkbox"/> Expansion	<input checked="" type="checkbox"/> Yahoo
<input checked="" type="checkbox"/> Líderes Mexicanos	<input checked="" type="checkbox"/> Crain's	<input checked="" type="checkbox"/> Google	<input checked="" type="checkbox"/> Macroeconomía	<input checked="" type="checkbox"/> Uno más uno
<input checked="" type="checkbox"/> Excelsior	<input checked="" type="checkbox"/> Noroeste	<input checked="" type="checkbox"/> Debate(Noroeste)	<input checked="" type="checkbox"/> El Sol de México	<input checked="" type="checkbox"/> bravo
<input checked="" type="checkbox"/> Cambio (sonora)	<input checked="" type="checkbox"/> Expreso (sonora)	<input checked="" type="checkbox"/> El imparcial (sonora)	<input checked="" type="checkbox"/> Zócalo (Coahuila)	<input checked="" type="checkbox"/> El diario de Coahuil
<input checked="" type="checkbox"/> Vanguardia (Coahuila)	<input checked="" type="checkbox"/> Palabra (Coahuila)	<input checked="" type="checkbox"/> Heraldo de Saltillo	<input checked="" type="checkbox"/> La Voz de Durango	<input checked="" type="checkbox"/> Milenio Monterrey
<input checked="" type="checkbox"/> El Norte (Nuevo León)	<input checked="" type="checkbox"/> El Mañana de Nuevo L	<input checked="" type="checkbox"/> El Mañana de Reynosa	<input checked="" type="checkbox"/> El Bravo (Tamaulipas)	<input checked="" type="checkbox"/> La Voz de Michoacán
<input checked="" type="checkbox"/> Ocho Columnas (Jalis)	<input checked="" type="checkbox"/> Público (Jalisco)	<input checked="" type="checkbox"/> El Informador (Jalis)	<input checked="" type="checkbox"/> Mural (Jalisco)	<input checked="" type="checkbox"/> Diario
<input checked="" type="checkbox"/> Enfoque (Tepic)	<input checked="" type="checkbox"/> AM de León (Guanajua)	<input checked="" type="checkbox"/> Heraldo de León (Gua)	<input checked="" type="checkbox"/> Correo de hoy (Guana)	<input checked="" type="checkbox"/> Noticias de Querétar
<input checked="" type="checkbox"/> Hidrocálido (Aguasca)	<input checked="" type="checkbox"/> Pulso San Luis Potos	<input checked="" type="checkbox"/> Imagen de Zacatecas	<input checked="" type="checkbox"/> Cuarto Poder de Chia	<input checked="" type="checkbox"/> El Orbe de Tapachula
<input checked="" type="checkbox"/> Diario de Chiapas	<input checked="" type="checkbox"/> Tiempo de Oaxaca	<input checked="" type="checkbox"/> El imparcial de Oaxa	<input checked="" type="checkbox"/> Diario 17 de Guerrer	<input checked="" type="checkbox"/> El Sur (Guerrero)
<input checked="" type="checkbox"/> El Sol de Chilpancín	<input checked="" type="checkbox"/> Diario de Yucatán	<input checked="" type="checkbox"/> Novedades de Quintan	<input checked="" type="checkbox"/> Por Esto de Mérida	<input checked="" type="checkbox"/> Tabasco Hoy
<input checked="" type="checkbox"/> Presente (Tabasco)	<input checked="" type="checkbox"/> Diario Olmeca (Tabas)	<input checked="" type="checkbox"/> Imagen de Veracruz	<input checked="" type="checkbox"/> Dictamen (Veracruz)	<input checked="" type="checkbox"/> Notiver (Veracruz)
<input checked="" type="checkbox"/> AZ de Xalapa (Veracr)	<input checked="" type="checkbox"/> La Opinión de Poza R	<input checked="" type="checkbox"/> Liberal del Sur (Ver)	<input checked="" type="checkbox"/> Síntesis de Puebla	<input checked="" type="checkbox"/> Unión de Morelos
<input checked="" type="checkbox"/> Diario de Morelos	<input checked="" type="checkbox"/> ABC de Toluca	<input checked="" type="checkbox"/> El Diario de Toluca		

Procesar diarios

Figura 7.64: Selección de los diarios a través del botón de marcado y desmarcado

Se pueden seleccionar únicamente los diarios principales (Fig. 7.66).

Sección de procesamiento

Clasificación del archivo

Entrenamiento **Procesamiento** Consulta

Marcar/Desmarcar Todos

Marcar/Desmarcar Principales

<input checked="" type="checkbox"/> Reforma	<input checked="" type="checkbox"/> El Universal	<input checked="" type="checkbox"/> La Jornada	<input checked="" type="checkbox"/> Milenio Diario	<input checked="" type="checkbox"/> El Economista
<input checked="" type="checkbox"/> El Financiero	<input checked="" type="checkbox"/> La Crónica	<input checked="" type="checkbox"/> Excelsior	<input type="checkbox"/> Expansion	<input type="checkbox"/> Yahoo
<input type="checkbox"/> Líderes Mexicanos	<input type="checkbox"/> Crain's	<input type="checkbox"/> Google	<input type="checkbox"/> Macroeconomía	<input type="checkbox"/> Uno más uno
<input type="checkbox"/> Excelsior	<input type="checkbox"/> Noroeste	<input type="checkbox"/> Debate(Noroeste)	<input type="checkbox"/> El Sol de México	<input type="checkbox"/> bravo
<input type="checkbox"/> Cambio (sonora)	<input type="checkbox"/> Expreso (sonora)	<input type="checkbox"/> El imparcial (sonora)	<input type="checkbox"/> Zócalo (Coahuila)	<input type="checkbox"/> El diario de Coahuil
<input type="checkbox"/> Vanguardia (Coahuila)	<input type="checkbox"/> Palabra (Coahuila)	<input type="checkbox"/> Heraldo de Saltillo	<input type="checkbox"/> La Voz de Durango	<input type="checkbox"/> Milenio Monterrey
<input type="checkbox"/> El Norte (Nuevo León)	<input type="checkbox"/> El Mañana de Nuevo L	<input type="checkbox"/> El Mañana de Reynosa	<input type="checkbox"/> El Bravo (Tamaulipas)	<input type="checkbox"/> La Voz de Michoacán
<input type="checkbox"/> Ocho Columnas (Jalis)	<input type="checkbox"/> Público (Jalisco)	<input type="checkbox"/> El Informador (Jalis)	<input type="checkbox"/> Mural (Jalisco)	<input type="checkbox"/> Diario
<input type="checkbox"/> Enfoque (Tepic)	<input type="checkbox"/> AM de León (Guanajua)	<input type="checkbox"/> Heraldo de León (Gua)	<input type="checkbox"/> Correo de hoy (Guana)	<input type="checkbox"/> Noticias de Querétar
<input type="checkbox"/> Hidrocálido (Aguasca)	<input type="checkbox"/> Pulso San Luis Potos	<input type="checkbox"/> Imagen de Zacatecas	<input type="checkbox"/> Cuarto Poder de Chia	<input type="checkbox"/> El Orbe de Tapachula
<input type="checkbox"/> Diario de Chiapas	<input type="checkbox"/> Tiempo de Oaxaca	<input type="checkbox"/> El imparcial de Oaxa	<input type="checkbox"/> Diario 17 de Guerrer	<input type="checkbox"/> El Sur (Guerrero)
<input type="checkbox"/> El Sol de Chilpancín	<input type="checkbox"/> Diario de Yucatán	<input type="checkbox"/> Novedades de Quintan	<input type="checkbox"/> Por Esto de Mérida	<input type="checkbox"/> Tabasco Hoy
<input type="checkbox"/> Presente (Tabasco)	<input type="checkbox"/> Diario Olmeca (Tabas)	<input type="checkbox"/> Imagen de Veracruz	<input type="checkbox"/> Dictamen (Veracruz)	<input type="checkbox"/> Notiver (Veracruz)
<input type="checkbox"/> AZ de Xalapa (Veracr)	<input type="checkbox"/> La Opinión de Poza R	<input type="checkbox"/> Liberal del Sur (Ver)	<input type="checkbox"/> Síntesis de Puebla	<input type="checkbox"/> Unión de Morelos
<input type="checkbox"/> Diario de Morelos	<input type="checkbox"/> ABC de Toluca	<input type="checkbox"/> El Diario de Toluca		

Procesar diarios

Figura 7.65: Selección de los diarios principales

Una vez que se han seleccionado los diarios que se desean procesar se presiona el botón “**Procesar Diarios**”

Sección de procesamiento

Clasificación del archivo

Entrenamiento **Procesamiento** Consulta

Marcar/Desmarcar Todos

Marcar/Desmarcar Principales

<input checked="" type="checkbox"/> Reforma	<input checked="" type="checkbox"/> El Universal	<input checked="" type="checkbox"/> La Jornada	<input checked="" type="checkbox"/> Milenio Diario	<input checked="" type="checkbox"/> El Economista
<input checked="" type="checkbox"/> El Financiero	<input checked="" type="checkbox"/> La Crónica	<input checked="" type="checkbox"/> Excelsior	<input type="checkbox"/> Expansion	<input type="checkbox"/> Yahoo
<input type="checkbox"/> Líderes Mexicanos	<input type="checkbox"/> Crain's	<input type="checkbox"/> Google	<input type="checkbox"/> Macroeconomía	<input type="checkbox"/> Uno más uno
<input type="checkbox"/> Excelsior	<input type="checkbox"/> Noroeste	<input type="checkbox"/> Debate(Noroeste)	<input type="checkbox"/> El Sol de México	<input type="checkbox"/> bravo
<input type="checkbox"/> Cambio (sonora)	<input type="checkbox"/> Expreso (sonora)	<input type="checkbox"/> El imparcial (sonora)	<input type="checkbox"/> Zócalo (Coahuila)	<input type="checkbox"/> El diario de Coahuil
<input type="checkbox"/> Vanguardia (Coahuila)	<input type="checkbox"/> Palabra (Coahuila)	<input type="checkbox"/> Heraldo de Saltillo	<input type="checkbox"/> La Voz de Durango	<input type="checkbox"/> Milenio Monterrey
<input type="checkbox"/> El Norte (Nuevo León)	<input type="checkbox"/> El Mañana de Nuevo L	<input type="checkbox"/> El Mañana de Reynosa	<input type="checkbox"/> El Bravo (Tamaulipas)	<input type="checkbox"/> La Voz de Michoacán
<input type="checkbox"/> Ocho Columnas (Jalis)	<input type="checkbox"/> Público (Jalisco)	<input type="checkbox"/> El Informador (Jalis)	<input type="checkbox"/> Mural (Jalisco)	<input type="checkbox"/> Diario
<input type="checkbox"/> Enfoque (Tepic)	<input type="checkbox"/> AM de León (Guanajua)	<input type="checkbox"/> Heraldo de León (Gua)	<input type="checkbox"/> Correo de hoy (Guana)	<input type="checkbox"/> Noticias de Querétar
<input type="checkbox"/> Hidrocálido (Aguasca)	<input type="checkbox"/> Pulso San Luis Potos	<input type="checkbox"/> Imagen de Zacatecas	<input type="checkbox"/> Cuarto Poder de Chia	<input type="checkbox"/> El Orbe de Tapachula
<input type="checkbox"/> Diario de Chiapas	<input type="checkbox"/> Tiempo de Oaxaca	<input type="checkbox"/> El imparcial de Oaxa	<input type="checkbox"/> Diario 17 de Guerrer	<input type="checkbox"/> El Sur (Guerrero)
<input type="checkbox"/> El Sol de Chilpancín	<input type="checkbox"/> Diario de Yucatán	<input type="checkbox"/> Novedades de Quintan	<input type="checkbox"/> Por Esto de Mérida	<input type="checkbox"/> Tabasco Hoy
<input type="checkbox"/> Presente (Tabasco)	<input type="checkbox"/> Diario Olmeca (Tabas)	<input type="checkbox"/> Imagen de Veracruz	<input type="checkbox"/> Dictamen (Veracruz)	<input type="checkbox"/> Notiver (Veracruz)
<input type="checkbox"/> AZ de Xalapa (Veracr)	<input type="checkbox"/> La Opinión de Poza R	<input type="checkbox"/> Liberal del Sur (Ver)	<input type="checkbox"/> Síntesis de Puebla	<input type="checkbox"/> Unión de Morelos
<input type="checkbox"/> Diario de Morelos	<input type="checkbox"/> ABC de Toluca	<input type="checkbox"/> El Diario de Toluca		

Procesar diarios








Figura 7.66: Botón que inicia el procesamiento de los diarios seleccionados

El procesamiento de los archivos de texto digitales y la búsqueda de temas de interés es, en general, un proceso que lleva tiempo pero al finalizar muestra los títulos de las notas encontradas en los archivos y un recuadro a la derecha que permite ver el cuerpo de la nota completa (Fig. 7.68).

Sección de procesamiento

Clasificación del archivo

Entrenamiento Procesamiento **Consulta**

2010-02-22 Jaque Mate / Miedo en Torreón	
2010-02-22 Eleva Honeywell International su meta de ganancias	
2010-02-22 Queda en libertad bajo fianza joven que atropelló a un policía al intentar escapar del alcoholímetro	
2010-02-22 Joven que mató a policía de tránsito quedó en libertad	
2010-02-22 Policía fue atropellado por conductor.	
2010-02-22 Deep Black en el ME Cancún	
2010-02-22 ?Adults Only?	
2010-02-22 St. Valentines Weekend	
2010-02-22 Cuidado del medio ambiente desde la cocina	
2010-02-22 Estiman crecimiento de construcción en 2.5%	
2010-02-22 Liberan a joven que mató a poli al huir del alcoholímetro	
2010-02-22 Breves Deportivas	
2010-02-22 Outsourcing: 10 errores que cometen los departamentos de TI	
2010-02-22 Archivado de bases de datos se suben a la nube	
2010-02-22 Softtek ahora también en Brasil	
2010-02-22 Transportes 100% automatizados	
2010-02-22 Gfi Defiende Un Modelo Híbrido De Soluciones En La Nube Y Aplicaciones En Local, Para Proteger La In	
2010-02-22 Ca Y Netapp Crean Soluciones Para Mejorar La Gestión De Las Infraestructuras Cloud Públicas Y Privad	
2010-02-22 SAP y Open Text amplían su acuerdo global de distribución	
2010-02-22 Facebook se decide por PayPal	
2010-02-22 VMware finaliza la adquisición de Zimbra	

Realizado por Rocío Ortiz © 2009 FI, UNAM






















Figura 7.67: Listado de las nota encontradas al final del procesamiento

- Consulta. En la pestaña de consulta se muestra una lista de los títulos de las noticias del día encontradas (Fig. 7.69), si se desea ver el cuerpo de la nota se puede presionar el cuadro ubicado en el lado derecho del título y se abrirá una ventana con el cuerpo de la noticia.

Sección de procesamiento

Clasificación del archivo

Entrenamiento Procesamiento **Consulta**

- 2010-02-22 Jaque Mate / Miedo en Torreón 
- 2010-02-22 Eleva Honeywell International su meta de ganancias 
- 2010-02-22 Queda en libertad bajo fianza joven que atropelló a un policía al intentar escapar del alcoholímetro 
- 2010-02-22 Joven que mató a policía de tránsito quedó en libertad 
- 2010-02-22 Policía fue atropellado por conductor. 
- 2010-02-22 Deep Black en el ME Cancún 
- 2010-02-22 ?Adults Only? 
- 2010-02-22 St. Valentines Weekend 
- 2010-02-22 Cuidado del medio ambiente desde la cocina 
- 2010-02-22 Estiman crecimiento de construcción en 2.5% 
- 2010-02-22 Liberan a joven que mató a poli al huir del alcoholímetro 
- 2010-02-22 Breves Deportivas 
- 2010-02-22 Outsourcing: 10 errores que cometen los departamentos de TI 
- 2010-02-22 Archivado de bases de datos se suben a la nube 
- 2010-02-22 Softtek ahora también en Brasil 
- 2010-02-22 Transportes 100% automatizados 
- 2010-02-22 Gfi Defiende Un Modelo Híbrido De Soluciones En La Nube Y Aplicaciones En Local, Para Proteger La In 
- 2010-02-22 Ca Y Netapp Crean Soluciones Para Mejorar La Gestión De Las Infraestructuras Cloud Públicas Y Privad 
- 2010-02-22 SAP y Open Text amplían su acuerdo global de distribución 
- 2010-02-22 Facebook se decide por PayPal 
- 2010-02-22 VMware finaliza la adquisición de Zimbra 

Realizado por **Rocío Ortiz** © 2009 FI, UNAM

Figura 7.68: Listado de las notas encontradas a lo largo del día

Analista

- Inicio. En la pestaña de inicio se encuentra una descripción del perfil del usuario y un correo electrónico de contacto entre el analista y el desarrollador (Fig. 7.70).

∴ Red ART2 ∴
Procesamiento digital de noticias.

Inicio Por revisar Aprobadas **Rocío Ortiz Albarrán**

Area de trabajo del analista

En esta página el analista del sistema puede revisar las noticias que arroja la herramienta acerca de un tema determinado para su validación, en la pestaña **Por revisar** se encontrarán las notas que no han sido aprobadas, en la pestaña **Aprobadas** se encuentran las notas que ya se muestran en las páginas del consultor.

Cualquier comentario, duda o reporte que se tenga del sistema favor de enviar un correo a:
ortiz_m@uxmcc2.iimas.unam.mx

Realizado por **Rocío Ortiz** © 2009 FI, UNAM

Figura 7.69: Sección correspondiente al analista

- Por revisar. Para determinar cuáles notas podrá visualizar el consultor el analista deberá presionar la pestaña en la que se listan todas las noticias que la herramienta encuentra acerca del tema elegido, esta es la pestaña “**Por revisar**” (Fig. 7.71), la

cual contiene todas las notas no aprobadas por los analistas filtradas por tema.

.: Red ART2 .:

Procesamiento digital de noticias.



Figura 7.70: Sección que permite ver las notas encontradas por el sistema

El analista elige el tema de interés que le corresponde revisar (Fig. 7.72).

.: Red ART2 .:

Procesamiento digital de noticias.

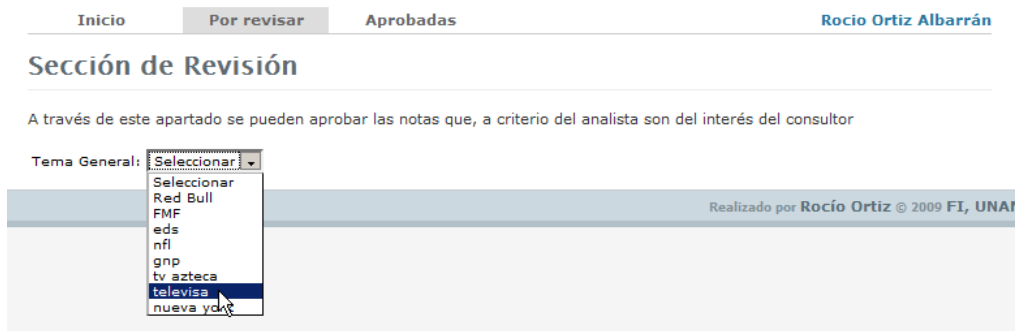


Figura 7.71: Selección de un tema de interés

Se muestran los títulos de las notas correspondientes al tema elegido y al lado izquierdo de los títulos se encuentra una casilla de verificación (Fig. 7.73).

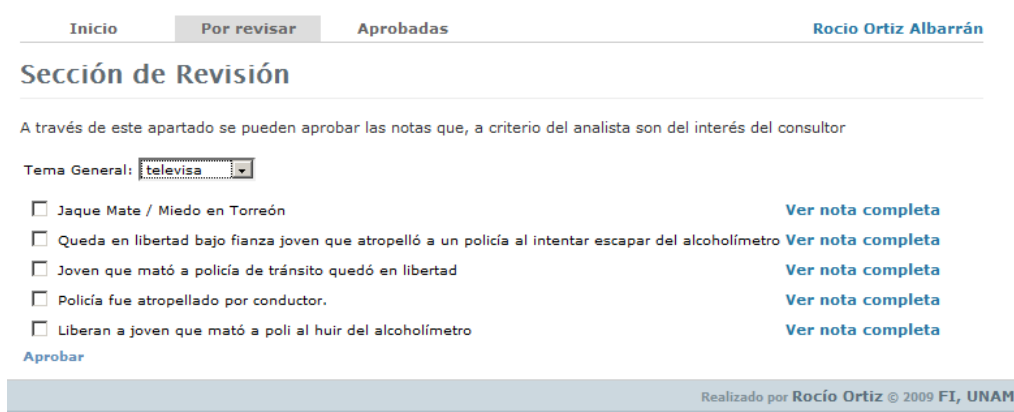


Figura 7.72: Listado de las notas correspondientes al tema de interés seleccionado

Si el analista juzga que la nota es del interés del consultor señala la casilla para que quede marcada la nota, repite el proceso con todos los títulos que desee mostrar en la página de consulta del tema (Fig. 7.74).

Sección de Revisión

A través de este apartado se pueden aprobar las notas que, a criterio del analista son del interés del consultor

Tema General: tv azteca

- | | |
|---|-----------------------------------|
| <input checked="" type="checkbox"/> Jaque Mate / Miedo en Torreón | Ver nota completa |
| <input type="checkbox"/> Queda en libertad bajo fianza joven que atropelló a un policía al intentar escapar del alcoholímetro | Ver nota completa |
| <input type="checkbox"/> Joven que mató a policía de tránsito quedó en libertad | Ver nota completa |
| <input type="checkbox"/> Policía fue atropellado por conductor. | Ver nota completa |
| <input type="checkbox"/> Liberan a joven que mató a poli al huir del alcoholímetro | Ver nota completa |

[Aprobar](#)

Realizado por Rocío Ortiz © 2009 FI, UNAM

Figura 7.73: Validación de las notas que se mostrarán al consultor final

Una vez que ha elegido todas las notas que desea mostrar presiona el botón “**Aprobar**” (Fig. 7.75).

The screenshot shows the 'Sección de Revisión' interface with the 'Por revisar' tab selected. The 'Aprobar' button is highlighted with a mouse cursor. The list of notes is the same as in Figure 7.73, but with the 'Joven que mató a policía de tránsito quedó en libertad' checkbox checked. The 'Aprobar' button is located below the list.

Figura 7.74: Botón que aprueba las notas elegidas por el analista

Si el título de la nota no es suficiente para determinar si la nota es de interés para el consultor se puede seleccionar el enlace que dice “**Ver nota completa**” (Fig. 7.76). Este enlace mostrará una ventana con el cuerpo de la nota completa.

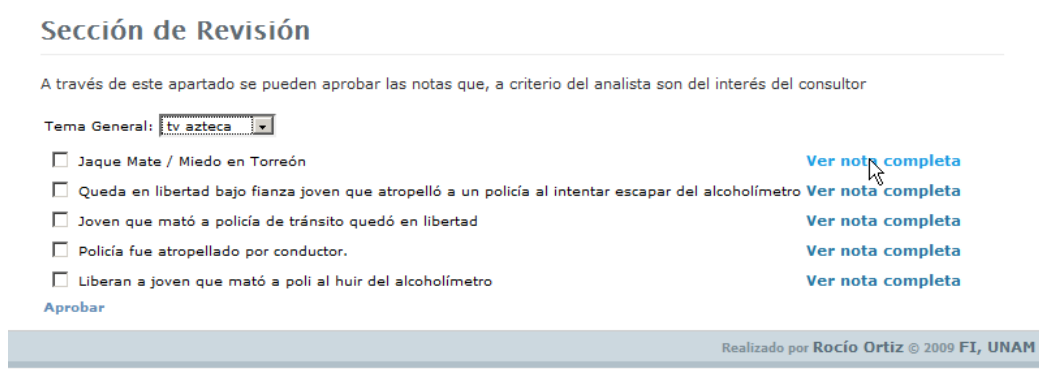


Figura 7.75: El enlace “ver nota completa” ayuda al analista a validar la nota

Una vez que se ha aprobado una nota desaparece de la lista de notas por aprobar y solo se muestran las que no han sido aprobadas (Fig. 7.77).

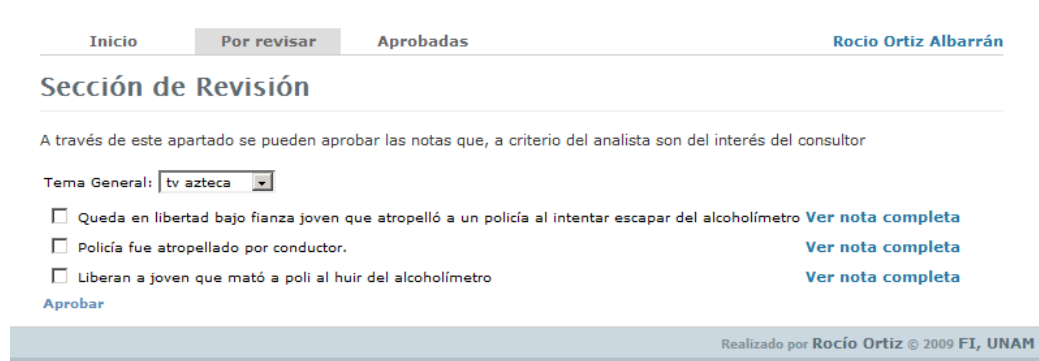


Figura 7.76: La lista muestra solo las notas no aprobadas

- Aprobadas. Para saber cuales notas ya se han aprobado acerca de un tema específico se presiona la pestaña “Aprobadas” (Fig. 7.78).

∴ Red ART2 ∴

Procesamiento digital de noticias.



Figura 7.77: Sección que permite listar las notas ya validadas

Se muestra un combo en el cual están listados todos los temas activos (Fig. 7.79).

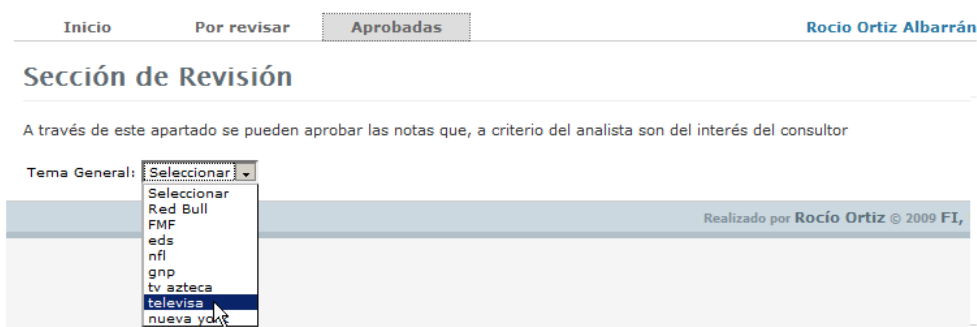


Figura 7.78: Selección de un tema para observar sus notas aprobadas

Se muestran las notas que ya han sido aprobadas de ese tema y se muestra al lado izquierdo del título un tache con el cual se puede eliminar la nota para que no se muestre en la página del consultor si es que se juzga que la nota no es de interés (Fig. 7.80).

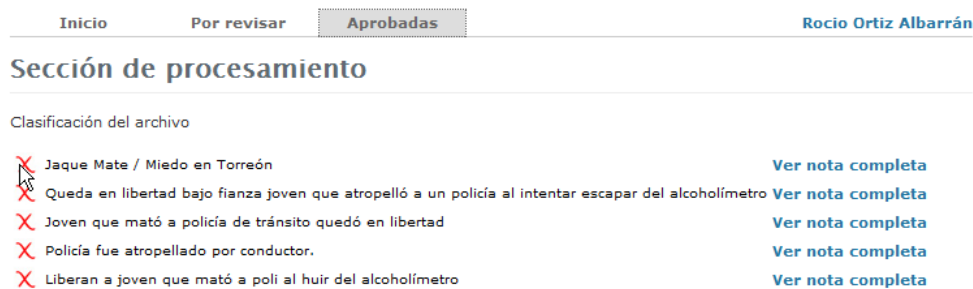


Figura 7.79: Opcion de eliminar alguna nota que ya habia sido aprobada

Una vez que la nota se elimina del apartado de notas aprobadas ya no se muestra su título en el apartado y el sistema queda listo para eliminar otra nota, además se puede observar el cuerpo de la nota completa si se presiona el enlace ubicado al lado derecho del título (Fig. 7.81).

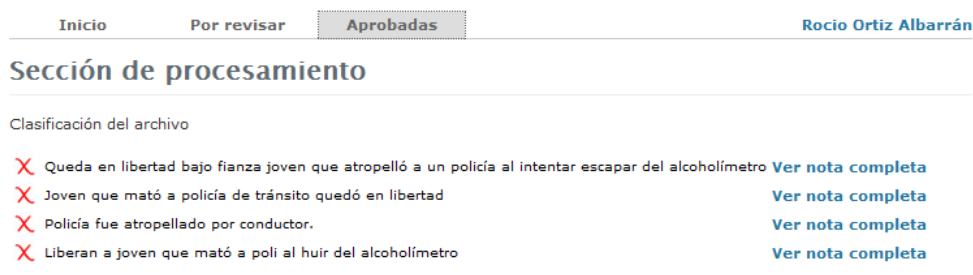


Figura 7.80: Una vez eliminada una nota desaparece de esta lista, solo se muestran las aprobadas para el consultor final

Apéndice B. Ejemplo de patrones por diario

```

1  switch ($clave_medio)
2      {
3          /* Universal */
4      case 2:
5          /* Para el título */
6          $n1[1]=1;
7          $etiqueta1[1]="title>";
8          $etiqueta2[1]="/title>";
9          $n2[1]=1;
10         /* para la seccion */
11         $n1[2]=1;
12         $etiqueta1[2]="";
13         $etiqueta2[2]="";
14         $n2[2]=1;
15         /* para la nota completa */
16         $n1[3]=1;
17         $etiqueta1[3]=" class=firma";
18         $etiqueta2[3]="/span></td></tr>";
19         $n2[3]=2;
20         $seccion=99;
21         $root_url="http://www.eluniversal.com.mx/notas/";
22     break;
23     /* Jornada */
24     case 3:
25         /* Para el título */
26         $n1[1]=1;
27         $etiqueta1[1]="documentFirstHeading";
28         $etiqueta2[1]="/h1>";
29         $n2[1]=1;
30         /* para la seccion */
31         $n1[2]=1;
32         $etiqueta1[2]="";
33         $etiqueta2[2]="";
34         $n2[2]=1;
35         /* para la nota completa */
36         $n1[3]=1;
37         $etiqueta1[3]="</p><p><i>";
38         $etiqueta2[3]="/div>";
39         $n2[3]=3;
40         $seccion=31;
41         $root_url="http://www.jornada.unam.mx/ultimas/$anio/$mes/$dia/";
42     break;
43     /* Cronica */
44     case 4:
45         /* Para el título */
46         $n1[1]=1;
47         $etiqueta1[1]="title>";
48         $etiqueta2[1]="/title>";
49         $n2[1]=1;
50         /* para la seccion */
51         $n1[2]=1;
52         $etiqueta1[2]="";
53         $etiqueta2[2]="";
54         $n2[2]=1;
55         /* para la nota completa */
56         $n1[3]=1;
57         $etiqueta1[3]=" notaPrincipalInt ";
58         $etiqueta2[3]="/div></td>";
59         $n2[3]=3;

```

```
60         $seccion=96;
61         $root_url="http://www.cronica.com.mx/";
62     break;
63     /*Financiero*/
64     case 7:
65         /*Para el título*/
66         $n1[1]=1;
67         $etiqueta1[1]="titulo-01";
68         $etiqueta2[1]="/span>";
69         $n2[1]=1;
70         /*para la seccion*/
71         $n1[2]=5;
72         $etiqueta1[2]="txt_ch-01";
73         $etiqueta2[2]="-";
74         $n2[2]=1;
75         /*para la nota completa*/
76         $n1[3]=1;
77         $etiqueta1[3]="ul<li>";
78         $etiqueta2[3]="/p></span>";
79         $n2[3]=2;
80         $seccion=53;
81         $root_url="http://www.elfinanciero.com.mx/ElFinanciero/Portal/cfpages
82     break;
```

Apéndice C. Código de la red ART2

```

1  <?
2  $beta=0.5;
3  $rho=0.97;
4  $alpha=0.001;
5  $m=array(0,0,0);
6  $categorias=3;
7  $W=inicializaEnUnos($W);
8  for($contador=0;$contador<$total;$contador++){
9      $TRestringido=array(1,1,1);
10     $lecturas[$contador]=split(" ", $lecturas[$contador]);
11     $x=$lecturas[$contador];
12     $I=$x;
13     for($k=count($x);$k<2*count($x);$k++){
14         $I[$k]=1-$I[$k-count($x)];
15     }
16     $salir=0;
17     $numCatInv=0;
18     while($salir!=1){
19         /*CALCULO DE Tj PARA CADA CATEGORÍA*/
20         for($j=0;$j<$categorias;$j++){
21             /*REVISO SI Tj NO ESTÁ BLOQUEADO*/
22             if($TRestringido[$j]!=-1000){
23                 $minimo=minimo($I,$W[$j]);
24                 $numerador=norma($minimo);
25                 $denominador=norma($W[$j]);
26                 /* */
27                 $T[$j]=$numerador/($alpha+$denominador);
28                 $TRestringido[$j]=$T[$j];
29             }
30         }
31         /*FIN DEL CALCULO DE Tj PARA CADA CATEGORÍA*/
32         /*SE OBTIENE EL MÁXIMO DE T[j] x ∈ A LA CATEGORÍA J*/
33         $Jg=max_key($TRestringido);
34         /*VERIFICACION DE LA RESONANCIA*/
35         $minimo=minimo($I,$W[$Jg]);
36         $numerador=norma($minimo);
37         $denominador=norma($I);
38         $resonancia=$numerador/$denominador;
39         if($resonancia>=$rho){
40             $categoriaG=$Jg;
41             for($k=0;$k<2*count($x);$k++){
42                 $W[$Jg][$k]=$beta*min($W[$Jg][$k],$I[$k])+(1-$beta)*$W[$Jg][$k];
43             }
44             $salir=1;
45         }
46         else{
47             $salir=0;
48             $TRestringido[$Jg]=-1000;
49             $numCatInv++;
50             if($numCatInv>$categorias){
51                 $salir=0;
52                 $W[$categorias]=inicializaEnUnos($W[$categorias]);
53                 $categorias++;
54             }
55         }/*FIN VERIFICACION DE LA RESONANCIA*/
56     }
57
58     $file=fopen("MatW.dat","w");

```



```
59     for ($renglones=0;$renglones<$categorias;$renglones++){
60         fprintf($file,"{");
61         for ($columnas=0;$columnas<2*count($x);$columnas++){
62             fprintf($file,"%A, ",$W[$renglones][$columnas]);
63         }
64         fprintf($file,"},\n");
65     }
66     echo "<font_color=#FF0000>el_vector_$contador_pertenece_a_la_categoria
        :_$categoriaG</font><<BR>";
67 }
```