Universidad de las Ciencias Informáticas

Facultad 10

Título: Propuesta de Módulo de Procesamiento Inteligente de Datos para el sistema Airesweb.

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

Autores:

Yilian Elena Matías León.

Katia Logás Fonseca.

Tutores:

Ing. Oscar Andrés Casas Machado.

Ing. Yulier Matías León.

Ciudad de La Habana, junio 2010

“No se puede dirigir si no se sabe analizar, si no hay datos verídicos, si no hay todo un sistema de recopilación de datos confiables, si no hay hombres habituados a recoger el dato y transformarlo en número de manera tal que esta sea su tarea esencial."

Ernesto Che Guevara.

Declaración de autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____

Yilian Elena Matías León

_____

Firma del Autor

Katia Logás Fonseca

_____

Firma del Autor

Oscar Andrés Casas Machado

_____

Firma del Tutor

Yulier Matías León

_____

Firma del Tutor

Agradecimientos

A mis padres por transmitirme siempre la idea y el apoyo en el momento preciso, por ayudarme a lidiar con cada una de mis dudas, gracias por transmitir la libertad de elegirlas por mí misma…les quiero enormemente.

A mi hermano por ser mi inspiración para seguir adelante, por no requerir de las palabras para saber qué sentimos y cuánto nos necesitamos.

A mis primas Kenia, Idialy y Kire por ser más que unas hermanas para mí.

A mis tías Juana, Oria y Marta por la incondicionalidad y transparencia de siempre.

A mis abuelos por ser los pilares que nos sostienen, por el cariño transparente, simplemente que nunca los olvidaré y los llevo en el alma como el mejor paradigma.

A toda mi familia, por poner más que un grano de arena para materializar este sueño compartido, a veces me he preguntado si hubiese vencido tantos obstáculos sin la seguridad plena de saber que están ahí…Gracias por el orgullo que me invade…

A Oscar por ser un tutor ejemplar, por lo mucho que he aprendido a su lado, por enseñarme a ser fuerte y a confiar en mí misma.

A los amigos y compañeros de estudio a los cuales agradezco por la compañía de estos años, por la satisfacción de estar juntos en tan diversas experiencias, en especial a Francisco y Ever por toda su alegría.

A Anay por ser una gran amiga, por sus buenos consejos y su excelente ortografía.

A Katia mi compañera de tesis y amiga inseparable de estos 5 años, por tantos momentos que compartimos juntas, por haber entrado a mi vida para quedarse.

A la UCI por resultar tan positiva para mi formación, facilitándome crecimiento profesional y personal. Agradecer al comandante Fidel, a Raúl y a la gloriosa Revolución por hacer de la utopía una realidad al crear esta obra tan inmensa.

A todos los que creyeron en mí. Muchas gracias.

Yilian Elena Matías León

Agradecimientos

A mis padres por su infinito amor y apoyo, por siempre estar a mi lado en los momentos que más los he necesitado, por enseñarme a superar los momentos difíciles y luchar para conseguir mí objetivo. A mi madre en especial por su comprensión y por ser mi ejemplo a seguir de mujer consagrada y madre abnegada. A mi padre por nunca separarse de nuestro lado y demostrarme cada día que sus hijas son lo más importante en su vida. Todo lo que tengo y lo que soy se los debo a ustedes. Nunca olviden lo mucho que los amo.

A mi hermana por todo el cariño y amor que me ha dado. Por ser una de las personas más importante en mi vida y por el inmenso amor que siento por ella.

A mis tíos Blanca y Ismael por ser unos padres más para mi, y por el apoyo que me han brindado todos estos años. A mi prima Irina por ser mi amiga y mi hermana, por su comprensión y por todo su cariño.

A mis abuelas que aunque hoy ya no se encuentran, sé que siempre han estado a mi lado dándome fuerzas para vencer todos los obstáculos que se me han cruzado en este largo camino.

A mi familia por siempre estar ahí, por ser tan unidos, por todo el cariño y amor que me han dado. Gracias ser el pilar fundamental en mi educación y mi desarrollo personal.

A mi novio Alain por demostrarme que siempre podemos encontrar nuevamente la felicidad, por darle alegría a mis días, por ser una persona especial, por el amor que siente por mí y porque a su lado he vivido los momentos más felices. Te amo mucho nene.

A mi tutor Oscar por todo lo que aprendí a su lado, por ayudarme a atravesar este recta final pero que sin duda es la más importante, por contribuir enormemente a mi desarrollo profesional y por hacernos ver que si podíamos en los momentos que pensamos que no lo lograríamos .

A Yilian mi compañera de tesis, por ser la mejor amiga que he tenido, por señalarme mis errores, por la confianza que me ha inspirado, por todo el cariño que me ha dado, por enseñarme que la verdadera amistad existe, por ser partícipe de mis alegrías y mis tristezas, y sobre todo por estar siempre ahí.

A Anay por su amistad todos estos años y tenderme siempre su mano cuando lo he necesitado. Siempre te recordaré como la gran amiga que encontré en ti y por ser una maravillosa persona.

A Francisco, Annier y Ever por todos los recuerdos que tengo de nuestra amistad, porque a pesar de todo se ganaron un lugar especial en mi corazón.

En general a todos mis amigos por estar a mi lado estos 5 años, por confiar en mí, por ayudarme a seguir adelante, por ayudarme a reconocer mis errores y a crecerme como persona y como profesional. Siempre los llevaré en mi corazón.

Katia Logás Fonseca

Dedicatoria

Con todo mi ser y las gracias innecesarias les dedico mi sueño porque sé que también es el suyo:

A mis padres que me han enseñado tanto, por la confianza y la fuerza de siempre, por ser mis guías y simplemente complementarlo todo.

A mi hermano por ser luz en mi vida, siempre la mejor y más constante compañía.

Yilian Elena Matías León

A mis padres Clara y Leovigildo porque sin su cariño y amor, sin su dedicación y entrega, y sin su empeño no hubiera logrado hacer realidad este sueño. Además por creer en mí, en apoyarme en cada paso que doy y enseñarme que con trabajo y optimismo siempre podemos lograr nuestras expectativas y hacer realidad nuestros sueños.

A mi hermana Kenia por infinito amor que siento por ella y por ser mi inspiración a seguir adelante.

A la Revolución Cubana por darnos esta oportunidad maravillosa.

En fin a todas las personas que confiaron en mí y que estuvieron a mi lado todo este tiempo logrando que hoy pudiera alcanzar mi gran sueño.

Katia Logás Fonseca

Resumen

Las herramientas de analítica web son muy utilizadas en la actualidad, siendo Airesweb (Analizador Inteligente de Registros de Servidores Web) un software de este tipo desarrollado en la Universidad de las Ciencias Informáticas. El propósito de la presente investigación consistió en proponer un módulo de procesamiento inteligente de datos para el sistema Airesweb, aprovechando las ventajas que brinda la minería de datos para el análisis de información. Para lograr dicho objetivo se realizó un estudio acerca de la minería de datos, detallándose sus métodos, técnicas y algoritmos, y se profundizó en la minería web y sus clasificaciones, estudiándose la minería web de uso y sus herramientas. Además se empleó la herramienta Weka para probar los algoritmos, seleccionándose las reglas de asociación y los patrones secuenciales para la construcción de un prototipo funcional y se estudió la arquitectura de Airesweb para determinar la integración del módulo a la misma. La propuesta del módulo constituyó una guía capaz de permitir a los miembros del proyecto Aires construir dicho módulo para ubicar al sistema Airesweb a nivel de un software de analítica web desarrollado internacionalmente.

Palabras Claves: minería de datos, minería web de uso, reglas de asociación, patrones secuenciales.

Índice General

Introducción 1

1. Fundamentación teórica 5

1.1 Trabajos similares 5

1.1.1 Ámbito Internacional 5

1.1.2 Ámbito Nacional. 8

1.2 Minería de Datos 9

1.2.1 Fases de un proceso clásico de Minería de Datos 10

1.2.2 Principales Tareas de la Minería de Datos 14

1.2.3 Técnicas de la Minería de Datos 16

1.2.4 Algoritmos de la Minería de Datos 18

1.2.5 Metodologías de aplicación de la Minería de Datos 22

1.2.6 Herramientas de Minería de Datos 25

1.3 Minería Web 31

1.3.1 Clasificación de Minería Web 33

1.3.2 Minería Web de Contenido 35

1.3.3 Minería Web de Estructura 37

1.4 Minería Web de Uso 39

1.4.1 Etapas de la Minería Web de Uso 39

1.4.2 Herramientas para la Minería Web de Uso 45

2. Diseño del Módulo de Procesamiento Inteligente de Datos 48

2.1 Arquitectura de Airesweb 48

2.1.1 Integración de MOPID a Airesweb 50

2.2 Ficheros 50

2.2.1 Fichero de registros temporales 50

2.2.2 Fichero ARFF 51
2.3 Algoritmos de búsqueda de reglas de asociación 52
2.3.1 Algoritmo Apriori 53
2.4 Algoritmo de búsqueda de patrones secuenciales 59
2.4.1 Algoritmo GeneralizedSequentialPatterns 60
2.5 Algoritmo de análisis de caminos HotSpot 62
2.6 Implementación del prototipo funcional 64
2.7 Propuesta de MOPID 66
2.8 Evaluación de los Resultados 70
Conclusiones 74
Recomendaciones 75
Referencias Bibliográficas 76
Bibliografía 80
Figuras Relacionadas 82
Glosario de Términos 89

Introducción

Internet es una inmensa red que conecta computadoras a nivel mundial, la cual les permite comunicarse, compartir información y servicios de una forma directa. El crecimiento acelerado de Internet ha tenido un gran impacto, gracias a ella millones de personas en el mundo tienen acceso a una gran cantidad de información en línea. Este intercambio de información se realiza generalmente mediante los sitios web, que no son más que un conjunto de páginas web organizadas jerárquicamente que se conectan entre ellas mediante vínculos.

Hoy por hoy, la World Wide Web (WWW) o también llamada la Web se ha convertido en el servicio más importante prestado por Internet. Mediante este servicio, el usuario dispone de un fácil acceso a la información ofrecida por multitud de servidores que se encuentran repartidos por todo el mundo. Esta información no es presentada únicamente en forma de texto, sino que la Web es un servicio multimedia que puede ofrecer textos, gráficos, sonidos, animaciones o vídeos [1].

Debido a la gran popularidad de la Web, Internet se ha transformado en una herramienta muy utilizada por las empresas para mejorar la prestación de sus servicios, generando una gran competitividad entre ellas y conduciéndolas a la búsqueda de nuevos mecanismos para enriquecer las funcionalidades de sus sitios web. Las nuevas formas de comportamiento dinámico de los usuarios para acceder a la información en estos sitios, genera grandes cantidades de datos no triviales y desconocidos, los cuales de ser analizados correctamente, podrían brindar información importante para estas instituciones. En el proceso de analizar la información generada por el tráfico de los usuarios, la minería de datos tiene una función primordial.

Se puede definir la minería de datos, como la extracción no trivial de información implícita, previamente desconocida y potencialmente útil a partir de los datos [2]. Para ser más específicos, la minería de datos utiliza diferentes técnicas para analizar grandes cantidades de datos que proceden generalmente de sistemas de información, con el objetivo de extraer conocimiento que se pueda utilizar para resolver problemas concretos de las organizaciones o empresas.

La minería de datos es muy utilizada en la Web para analizar el uso de los sitios presentes en Internet. A este proceso se le conoce como minería web, que consiste en la aplicación de las técnicas de minería de datos sobre información existente en la Web, complementada con otro tipo de información, empleando técnicas de extracción de patrones interesantes y potencialmente útiles, para descubrir correlaciones y tendencias significativas para la organización de dicha información [3]. Dentro de la minería web se encuentra la minería web de uso, que se basa en el proceso de extracción de patrones

no triviales de la información del tráfico web [4]. El uso de los patrones descubiertos ayudará a tomar decisiones más seguras que reporten algún beneficio para las empresas.

La minería web de uso, cada día es más popular y extendida. El análisis del tráfico de acceso a un determinado servidor web, puede ayudar a entender el comportamiento y los hábitos de los usuarios, así como a diseñar adecuadamente la estructura de la Web [3].

En febrero del año 2008, el Centro de Información para la Prensa (CIPRE), que tiene como misión coordinar la gestión de la información para el sistema de la prensa cubana, solicitó a la UCI un software para el análisis de los registros web. El sistema se encuentra en el desarrollo de su versión 1.0 y es necesario incorporarle el módulo que analizará de forma inteligente los datos generados por el tráfico web.

A partir de los planteamientos anteriores, se identificó por parte del grupo de proyecto “Analizador de Registros Web (Airesweb)”, perteneciente al departamento de Soluciones Informáticas para Internet, la necesidad de realizar un estudio sobre las variantes que puedan existir para el procesamiento inteligente de datos y la extracción de patrones de estos, dando lugar al siguiente **problema a resolver**: ¿Cómo procesar de forma inteligente los datos para extraer patrones no triviales de acceso a los sitios web en el sistema Airesweb?

Partiendo del problema definido, se define como **objeto de estudio** la Minería Web y como **campo de acción** la Minería Web de Uso.

Con el fin de resolver el problema planteado con anterioridad, se ha trazado como **objetivo general:** Proponer un módulo de procesamiento inteligente de datos (MOPID) para el sistema Airesweb.

Del objetivo general se desglosan los siguientes **objetivos específicos**:

- Realizar un estudio del estado del arte de los algoritmos y tecnologías de la minería de datos.
- Definir la arquitectura y el funcionamiento del MOPID.
- Evaluar la efectividad de los algoritmos y tecnologías seleccionadas.

Se establece como **idea a defender**: Probando los diferentes métodos de la minería de datos, se podrá obtener las diferentes variantes que se pueden aplicar al reconocimiento de patrones en MOPID.

Para dar cumplimiento a los objetivos específicos se formularon las siguientes **tareas investigativas**:

- Estudio de los algoritmos y tecnologías de la minería de datos.
- Estudio de los algoritmos y tecnologías de la minería web.
- Estudio de los algoritmos y tecnologías de la minería web de uso.
- Estudio de la arquitectura del sistema Airesweb y la integración de MOPID al mismo.
- Propuesta de un prototipo funcional.
- Estudio de las formas de evaluación de los resultados.

Del trabajo se esperan los siguientes **aportes prácticos**:

- Sentar un precedente en la investigación sobre minería web en el proyecto Airesweb.
- Sentar las bases teóricas para la posterior implementación del módulo.
- Dotar al software Airesweb de la investigación necesaria para el posterior desarrollo de un módulo de procesamiento inteligente de datos que lo permita colocar a la altura de un software de analítica web a nivel mundial.

Para el cumplimiento de las tareas se utilizarán los siguientes **métodos científicos** de investigación:

El Histórico-Lógico permitirá una mayor comprensión del estado actual de la minería web de uso a partir del análisis de su evolución y las etapas principales por las que han transitado.

El Analítico-Sintético se aplicará para buscar y analizar información relacionada con el objeto de estudio y formular conclusiones a través de la síntesis de los conocimientos y resultados obtenidos.

La Modelación se utilizará para diseñar un prototipo funcional de forma que refleje lo mejor posible la realidad que se presenta en el proceso de extracción de patrones no triviales.

Estructura del Contenido

El contenido del presente trabajo está estructurado en dos capítulos distribuidos de la siguiente manera:

Capítulo 1 *Fundamentación Teórica*. Este capítulo está centrado en el estado del arte relacionado con el objeto de estudio. En él se analizan las soluciones existentes a nivel nacional e internacional que puedan reutilizarse y se abordan temas relacionados con la minería web como parte de la minería de

datos, estableciéndose definiciones formales y otros aspectos fundamentales como son: aplicaciones, técnicas, algoritmos y herramientas.

Capítulo 2 *Diseño del Módulo de Procesamiento Inteligente de Datos.* En este capítulo se realiza una descripción de la arquitectura de Airesweb con el fin de aclarar la integración de MOPID en el mismo. Se propone una arquitectura interna para el módulo y se plantean las tareas, las técnicas y los algoritmos de la minería de datos que se utilizarán. Además se seleccionan las colecciones de registros para a partir de la implementación del prototipo funcional probar y evaluar cada algoritmo elegido.

1. Fundamentación teórica

1.1Trabajos similares

Las herramientas de analítica web permiten la recopilación, medición, evaluación y explicación de los datos que genera el tráfico sobre un sitio web, con el objetivo de asistir a la toma de decisiones para la optimización de este. Airesweb es un software de analítica web por lo que es necesario realizar un análisis a nivel mundial del desarrollo de este tipo de sistemas.

1.1.1 Ámbito Internacional

El crecimiento acelerado de Internet y la ampliación de los mercados tradicionales al entorno virtual han ocasionado que estos mercados sean cada día más dinámicos e inciertos, generando grandes volúmenes de datos que al analizarlos correctamente le aportarán información de gran utilidad a las organizaciones. En el análisis de los datos intervienen las herramientas de analítica web que ayudan a tomar decisiones objetivas basadas en información relevante. A nivel internacional se han desarrollado una serie de herramientas de analítica web que utilizan la minería web para la extracción de patrones interesantes, entre las que se encuentran: Google Analytics, WebTrends, AWStats, Omniture SiteCatalyst y ClickTracks.

Google Analytics[1] [5]

Google Analytics es una herramienta privativa de análisis web que proporciona información valiosa del tráfico del sitio web a las empresas. Posee funciones potentes, flexibles y fáciles de usar. Con su ayuda, las empresas pueden diseñar anuncios orientados a mejorar sus iniciativas de marketing[2]. Permite llevar la estadística del uso de un sitio web. Es una herramienta que proporciona un código Java Script que debe ser copiado en cada una de las páginas del sitio. Cuando un usuario accede al sitio, el script envía información al servidor de Google Analytics del usuario que está accediendo a la página. Es una plataforma de análisis web que no requiere de ningún servidor de alojamiento por el usuario. Con este sistema de estadística web que ofrece Google se pueden obtener informes detallados sobre el número de visitas que ha tenido determinada página web, quienes la han visitado, cuál ha sido la ruta que han seguido por sus páginas y con qué palabra han llegado a través de los buscadores. Está diseñado tanto para los especialistas como para los usuarios normales de las empresas.

[1] http://www.google.com/analytics/

[2] Conjunto de técnicas y métodos para promover la mejor venta posible de uno o varios productos.

WebTrends[3] [6]

La herramienta WebTrends es una solución de análisis de la actividad en los sitios web y brinda respuestas precisas de forma online que permiten mejorar la adquisición, conversión y retención de los usuarios. Es un servicio de estadística enfocado principalmente al sector empresarial que ofrece características extras de seguridad. Puede ajustarse a las necesidades de cualquier negocio. Permite obtener reportes de toda la actividad en un sitio web, gran capacidad de personalización y funciones de visualización intuitiva mediante una arquitectura flexible diseñada pensando en grandes proyectos. Es un software privativo, sus dos últimas versiones son: WebTrends Analytics 8 y WebTrends Analytics 9.

Esta herramienta permite disponer de los parámetros claves de la web actualizados a lo largo del día mediante Express Results, combinar el análisis web con otras fuentes de datos mediante la integración con el Excel, generar informes personalizados que den respuesta a las cuestiones más específicas y complejas, identificar los segmentos valiosos, analizar sus intereses y preferencias para optimizar la retención y ofertas de productos.

Con los productos WebTrends las empresas de marketing, ventas y desarrollo web podrán conocer en detalle la interacción de los visitantes y clientes con un sitio web. La solución de WebTrends es utilizado por miles de clientes en el mundo como son las instituciones gubernamentales y educativas, y corporaciones como American Express, IBM, Microsoft y NASA.

AWStats[4] [7]

Es una herramienta libre que genera estadísticas gráficas de avanzada para un determinado sitio web, mostrándole la información que se necesita para el seguimiento personalizado de una página. Trabaja como un CGI o desde una línea de comando.

Puede ofrecer algunas informaciones de forma gráfica como:

- Números de visitas y visitantes únicos (diarias y mensuales).
- Duración de las visitas.
- País de Origen.
- Ranking de páginas más vistas.

[3] http://www.webtrends.com/
[4] http://awstats.sourceforge.net/

- Usuarios autenticados.
- Día de la semana y hora de mayor tráfico en el sitio.
- Lista de Hosts, ultima visitas y listas de IP no resueltas.
- Páginas de entrada y salida.
- Sitios de procedencia.
- Tipo de archivo.
- Navegadores y versiones utilizados por los visitantes.
- Sistemas Operativos usados.
- Visitas de robots de búsqueda.
- Buscadores, palabras y frases claves usadas para encontrar un sitio.
- Errores HTTP.
- Cantidad de veces que un sitio es añadido a favoritos.
- Ratio de navegadores con soporte a Java, Flash, Real player, Quicktime.

Omniture SiteCatalyst[5]

Omniture SiteCatalyst es una plataforma de análisis web que permite el diseño de informes personalizados. Contiene funciones para la segmentación de los eventos de éxito y opciones para la tabulación cruzada, además buen soporte para videos y Flash. Provee herramientas de análisis sofisticadas para ayudar a convertir a los usuarios en clientes, permite medir los indicadores de los negocios a través del comportamiento de los usuarios y ofrece la estadística básica de cualquier web. Contiene varios productos adicionales empleados para la búsqueda automática de la optimización de palabras claves, encuestas, pruebas, integración de correo electrónico y minería de datos [8].

Omniture SiteCatalyst permite un análisis detallado de los negocios de forma online, ofreciendo valiosa información acerca del comportamiento de los usuarios con gran nivel de detalle y segmentación. Una de sus ventajas es que ofrece la posibilidad de crear complejos y específicos cuadros de mando adaptado a cada empresa. Proporciona información inteligente en tiempo real sobre las estrategias en línea y las iniciativas de marketing, ayuda a identificar las rutas del sitio web que generan más beneficio, los puntos que los usuarios abandonan y los factores que impulsan los eventos críticos. Es un servicio de pago y necesita para su instalación conocimiento previo [9].

Omniture SiteCatalyst proporciona [9]:

[5] http://www.omniture.com/

- Datos en tiempo real que permiten tomar decisiones más oportunas.
- Indicadores claves del rendimiento relativo a las iniciativas en línea.
- Alertas automáticas cada vez que un indicador clave cambia.
- Una ubicación de donde se pueda medir, analizar y optimizar todas las iniciativas multicanal y en línea.

ClickTracks[6]

ClickTracks es un costoso software de análisis web que se instala en los ordenadores de escritorios. Provee al usuario informes profundos y le da la posibilidad de ejecución de análisis específicos lo que le permite un gran control sobre la estadística. Está orientada hacia pequeñas y grandes empresas y no hacia un usuario particular. Posee una interfaz sencilla e intuitiva para cualquier usuario [10]. Muestra un mapa del sitio web y cómo navegan los visitantes. Es sencillo de configurar y funciona importando archivos de informe del servidor web, además que soporta servidores web como Apache y IIS. Puede comparar el rendimiento de las palabras claves en término de número de visitantes, tiempo, costos, ingresos y rendimiento [11]. ClickTracks Appetizer es una herramienta gratuita que incluye muchas de las características importantes de los paquetes no gratuitos de ClickTracks.

Entre sus ventajas se encuentran [11]:

- El análisis visual y comparativo a través de la identificación del tipo de visitantes.
- Evalúa de una forma intuitiva la eficacia de las campañas en cada página.
- Permite la segmentación del tipo de visitante.
- Mide el rendimiento de las campañas de marketing por Internet a través de la importación de datos.
- Permite la integración de datos del “carro de compra”.

1.1.2 Ámbito Nacional.

En los últimos años Cuba ha tenido un gran avance en el desarrollo de la informática, pero aún así no se tiene conocimiento de que se haya realizado ningún software de analítica web que contenga un módulo de minería web. Al ninguno de los software analizados anteriormente poder utilizarse ya que en su gran mayoría son privativos, siendo solamente libre AWStats pero su código no es reutilizable debido a que no es totalmente modular, se empieza a desarrollar en la UCI un analizador de registros

[6] http://www.lyris.com/solutions/lyris-hq/web-analytics/

web conocido como: sistema Airesweb. Actualmente se está desarrollando la versión 1.0 de dicho software y surge la necesidad de incorporarle un módulo que procese de manera inteligente los datos contenidos en los archivos de registros web para así ubicarlo a nivel de un software desarrollado internacionalmente.

Para dar cumplimiento al objetivo general de esta investigación es necesario realizar un estudio de todos los aspectos relacionados con la minería de datos y en especial la minería web.

1.2 Minería de Datos

La minería de datos ha surgido del análisis de un enorme volumen de información, con el fin de obtener resúmenes y conocimientos que ayuden a la toma de decisiones en las organizaciones. De manera general la minería de datos consiste en la explotación de los datos y se fundamenta en la inserción de diversas áreas de estudio como son: la Inteligencia Artificial, la Estadística, la Computación Gráfica, las Bases de Datos y el Procesamiento Masivo.

Según Fayyad, Piatetsky-Shapiro y Smyth, la gran cantidad de datos que se almacenan en las organizaciones hacen imposible la utilización de métodos manuales para su análisis [12]. Por ello son necesarias técnicas y herramientas informáticas capaces de ayudar al hombre de una forma inteligente en el análisis de grandes cantidades de datos. Además estos autores aportan la definición más citada sobre el tema: "minería de datos es un proceso no trivial de extracción de patrones a partir de los datos que sean válidos, previamente desconocidos, potencialmente útiles y comprensibles".

Igualmente existen otras definiciones de diversos autores sobre la minería de datos:

- Es un conjunto de técnicas y herramientas aplicadas al proceso no trivial de extraer y presentar el conocimiento implícito, previamente desconocido, potencialmente útil y humanamente comprensible a partir de grandes conjuntos de datos con el objeto de predecir de forma automatizada tendencias y comportamientos y/o descubrir de una forma automatizada modelos previamente desconocidos [13].
- Es el proceso de plantear varias preguntas y extraer información útil, patrones y tendencias de grandes cantidades de datos generalmente almacenados en bases de datos [14].
- La integración de un conjunto de áreas que tienen como propósito la identificación de un conocimiento obtenido a partir de las bases de datos que aporten un sesgo hacia la toma de decisión [4].

Desde un punto de vista histórico ha sido el resultado de un largo proceso de investigación que comenzó en los años 60 con el desarrollo de los primeros sistemas de almacenamiento y recuperación de datos. Fue a mediados de los años 80 que se dieron los primeros pasos hacia la minería de datos, como se conoce en la actualidad, cuando las bases de datos no sólo se limitaron a almacenar información, sino que se le adicionaron técnicas de procesado y modelado de datos. La necesidad de almacenar la información ha motivado el desarrollo de sistemas cada vez más eficientes y con una mayor capacidad de almacenamiento.

1.2.1 Fases de un proceso clásico de Minería de Datos

El proceso de minería de datos cuenta con seis fases fundamentales mostradas en la figura 1, que no están claramente diferenciadas, lo que hace que sea un proceso iterativo e interactivo.

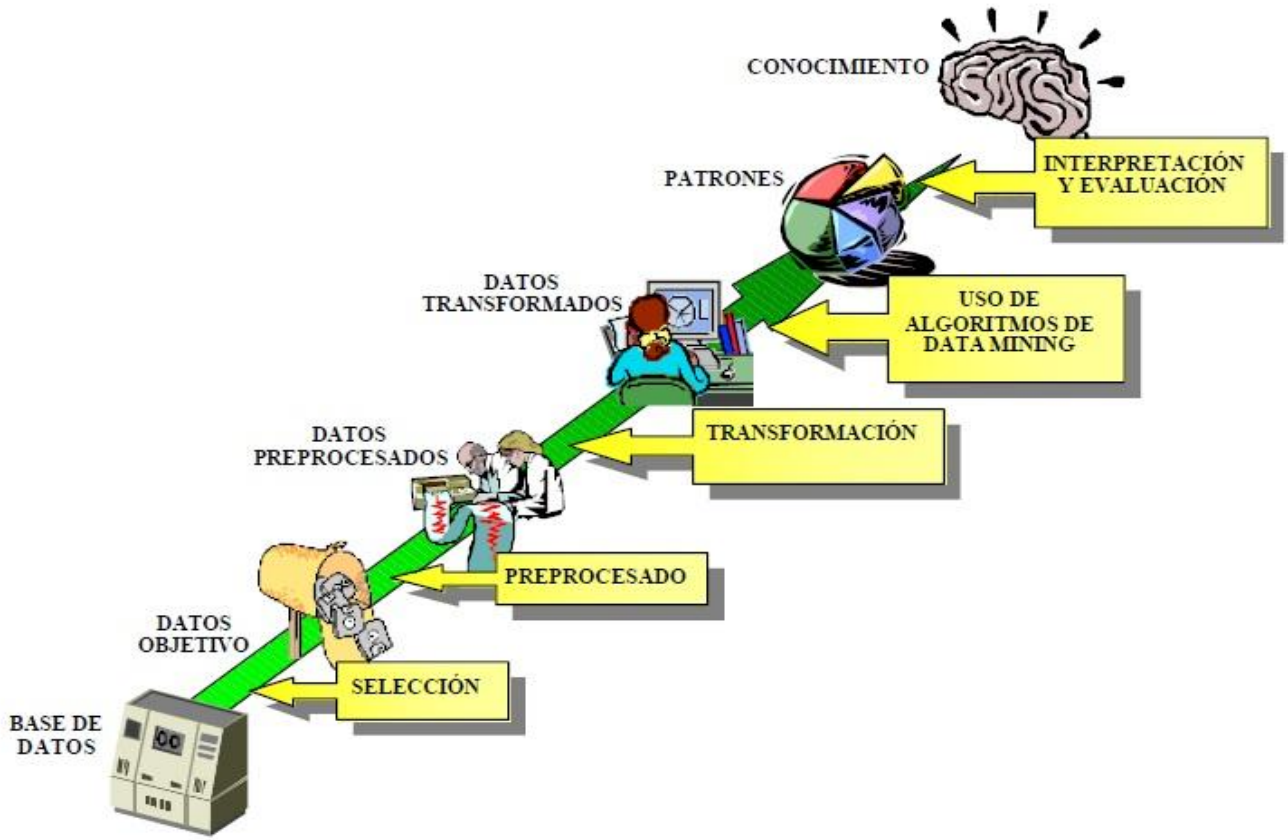

Figura 1. Fases del proceso de Minería de Datos.

Definición del Alcance y los Objetivos

El primer paso de un proyecto de minería de datos consiste en conocer el desarrollo y dominio de la aplicación, determinar el conocimiento relevante a usar, así como establecer los objetivos del usuario final [15]. Definir el alcance, los objetivos del problema a resolver y comprender el sistema en todos sus detalles garantizan el éxito del proyecto y evitan significativas pérdidas de tiempo en fases posteriores, por lo que resulta necesario emplear grandes esfuerzos en esta fase.

Según Dorian Pyle los pasos que llevan al éxito van a depender de los siguientes aspectos [16]:

- Identificación correcta de los problemas a resolver: Muchas veces esta tarea puede parecer trivial pero puede suceder que el problema no se comprenda completamente.
- Definición con precisión de problemas: Será necesario dividir las descripciones del problema que son demasiado generales en componentes más pequeños que puedan ser contrastados por la información examinada.
- Uso de mapas cognitivos: Cuando la cantidad de información es relevante puede ser necesario estructurar la información mediante un mapa cognitivo.
- Resolver las ambigüedades: Es conveniente resolver las ambigüedades que puedan surgir, ya que la imagen del problema en el cliente está formada por una cantidad de conceptos asociados, que él tiene asumidos, pero que pueden no ser tan claros ya que no conocen como se realiza el proceso.
- Determinar dentro del grupo de problemas el grado de importancia y dificultad de cada uno.
- Definir qué resultados se esperan conseguir.

Selección de los Datos Relevantes

La identificación de los datos relevantes para una operación de minería de datos es una tarea que puede hacerse de forma automática o manual, aunque se recomienda que sea realizada manualmente por un analista. En esta fase se seleccionan los datos más relevantes del proceso, teniendo en cuenta la homogeneidad y la variación de los mismos, así como la estrategia de muestreo utilizada.

Los datos pueden obtenerse directamente desde archivos o a partir de un almacén de datos (Data Warehouse), en la segunda opción existe una localización más centralizada y mayor validez de los datos.

Preprocesado y Limpieza de Datos

Esta fase es indispensable debido a que el formato de los datos en bruto no es el adecuado para aplicar los algoritmos de la minería de datos y tiene como objetivo la transformación del conjunto de datos original en un conjunto de datos más significativo y manejable [18]. Engloba todas aquellas técnicas para el análisis de datos que permiten mejorar la calidad del conjunto de datos inicial, de manera tal que los métodos utilizados para la extracción de conocimiento puedan obtener una información más precisa. Mediante este proceso se eliminan el ruido y los datos incompletos, se rellenan los datos inexistentes según las necesidades y el algoritmo a utilizar. Además se obtienen muestras de los datos filtrados y se reduce el número de valores posibles.

Debido a fallos cometidos durante el proceso de selección de los datos, ya sea este manual o automático, es muy común la ausencia de ciertos valores en los atributos, lo cual constituye un problema ya que se puede perder información relevante para el proceso general. Existen algunas técnicas para resolver este problema antes de que estos datos sean procesados mediante los algoritmos de minería de datos. La técnica más simple y menos recomendable consiste en eliminar aquellos datos que poseen atributos sin valor. En caso de ser necesarios estos datos se puede rellenar los valores ausentes con algún valor válido. Hay que tener en cuenta que el algoritmo de aprendizaje puede interpretar erróneamente estos valores como interesantes, por lo que se debe sustituir las ausencias por valores cuya influencia sea la menor posible [15].

El ruido es un error aleatorio o variación en el valor de un atributo generalmente debido a errores en la medida del mismo. El ruido es más difícil de detectar a simple vista ya que son valores presentes en el conjunto de datos y para mitigar los efectos que provoca es necesario aplicar las denominadas técnicas de suavizado. El método de suavizado más sencillo consiste en ordenar los valores de un atributo y distribuirlos en grupos o recipientes de igual número de valores o de igual rango, independientemente de los valores que contenga. Luego se realiza un tratamiento local sustituyendo los valores de cada grupo por la media, mediana o moda de dicho grupo. La aplicación de esta técnica suaviza los efectos del ruido pero no garantiza su eliminación ya que un atributo puede tomar valores que no correspondan a la característica del ejemplo al que pertenece [15].

En las primeras tareas de esta fase se identifican casi manualmente los atributos o variables existentes en la base de datos y luego se convierten en otro tipo dependiendo de las necesidades posteriores, estos deben ser acomodados a los algoritmos que se vayan a utilizar por lo que es necesario entender

cómo trabajan estos algoritmos. Las variables van a representar al conjunto datos, es decir, se eligen las variables más influyentes en el problema sin sacrificar la calidad del conjunto de datos original.

Transformación de los Datos

En esta fase se prepara la información que se tiene para que pueda ser procesada por los algoritmos de minería de datos, pues la forma en que se encuentra la información originalmente no es la adecuada. Por lo tanto se hace necesario la aplicación de alguna técnica de transformación destinada a la modificación de los datos para mejorar el proceso de aprendizaje, no para corregir errores en los mismos. Algunas de las técnicas que se pueden aplicar son: la normalización o cambio de escala, la discretización y la numerización. La normalización se puede utilizar cuando sea necesario normalizar todos los atributos al mismo rango, para poder obtener una homogeneidad en los datos.

Una de los aspectos importantes de un atributo es su tipo, ya que el hecho de que sea nominal o numérico determina en gran medida como va a ser tratado por las herramientas de minería de datos. La discretización es la conversión de un valor numérico a un valor nominal mientras que la numerización es el proceso inverso. Estas técnicas son útiles cuando el método de minería de datos a utilizar no admita atributos nominales o numéricos.

En esta fase se busca [17]:

- Extracción de las características útiles de los datos (reducción de la dimensionalidad).
- Transformación de los datos con el objetivo de proporcionar una representación de los datos más intuitiva y manejable.
- Fundamentalmente se realizan tres tareas específicas: reducción de datos, creación de datos derivados y transformación de la distribución de los datos.

Uso de los Algoritmos de la Minería de Datos

La extracción de conocimiento es la fase fundamental dentro de todo el proceso, en esta se utilizan las técnicas de la minería de datos y se obtiene un modelo de conocimiento que representa patrones de comportamiento observados en las variables del problema o relaciones de asociación de dichas variables. También se pueden utilizar varias técnicas para generar distintos modelos, teniendo en cuenta que generalmente cada una de ellas necesita un preprocesado diferente de los datos [19].

En primer lugar se determina el tipo de tarea de minería más apropiado, luego se selecciona el modelo, por ejemplo: para una tarea de clasificación se puede usar un árbol de decisión porque se

quiere obtener un modelo en forma de reglas, también se seleccionan los algoritmos que resuelvan la tarea y obtengan el tipo de modelo deseado en dependencia con el objetivo principal del proyecto.

El resultado de la aplicación de los algoritmos de descubrimiento pueden ser de carácter descriptivo (ayuda a la comprensión del problema) o de carácter predictivo (ayuda a prever el comportamiento de la entidad analizada).

Interpretación y Evaluación de los Resultados

Después de la obtención del modelo de conocimiento, ya en la fase de interpretación y evaluación se deben evaluar sus resultados, comprobando que las conclusiones que arroja son válidas. En caso de haber obtenido varios modelos, estos deben compararse en busca de aquel que se ajuste mejor al problema. Es necesario utilizar las técnicas de visualización para una mejor valoración de los resultados. Además, en esta fase hay que comprobar el cumplimiento de los objetivos propuestos en la primera fase.

La información extraída se utilizará para la extracción de nueva información para lo cual será necesario volver a fases anteriores, de esta forma el proceso de minería de datos se convierte en un proceso iterativo.

1.2.2 Principales Tareas de la Minería de Datos

Las tareas pueden considerarse como un tipo de problema a ser resuelto por un determinado algoritmo de minería de datos, es decir, cada una de estas tareas tienen sus propias exigencias o requerimientos y por tanto el resultado obtenido con una tarea específica puede ser muy diferente del resultado de otra. Estas tareas se pueden clasificar en predictivas o descriptivas, entre las predictivas se encuentran la clasificación y la regresión y entre las descriptivas el agrupamiento, las reglas de asociación, las reglas de asociación secuenciales y las correlaciones [20].

Es muy importante resaltar que no es necesario que un proyecto cumpla con todas ellas pues depende de la naturaleza del problema. A continuación se describirán las principales tareas de la minería de datos [21]:

- Clasificación. Agrupa todas las herramientas que permiten asignar a un elemento la pertenencia a un determinado grupo o clase. Más concretamente cada instancia o registro de la base de datos pertenece a una determinada clase la cual se indica mediante el valor de un atributo denominado clase de la instancia, el objetivo fundamental es predecir cuál sería la

clase de nuevas instancias de las que se desconoce la clase. Se usan árboles de decisión y sistemas de reglas o análisis de discriminantes.

- Regresión o estimación. Es el aprendizaje de una función real que asigna a cada instancia un valor real de tipo numérico, tiene como objetivo inducir un modelo para poder predecir el valor de la clase dados los valores de los atributos. Se usan árboles de regresión, redes neuronales artificiales, regresión lineal. [15]. La principal diferencia respecto a la clasificación consiste en que el valor a predecir es numérico.
- Agrupamiento o clustering. Se considera como la tarea descriptiva por excelencia y consiste en la búsqueda de la identificación de tipologías o grupos a partir de los datos en los cuales los elementos tienen gran similitud entre sí y se diferencian de los otros grupos lo cual permite el tratamiento de forma particular de cada uno de estos grupos. Los datos son agrupados según el principio de maximizar la similitud entre los elementos de un grupo minimizando la similitud entre grupos distintos. Este tipo de tareas son utilizadas en una gran variedad de ámbitos para la descripción y clasificación de información. También se encuentra estrechamente relacionada con la sumarización pues en esta cada grupo formado se considera como un resumen de los elementos que lo forman para describir de una manera concisa los datos.
- Reglas de asociación. Se establecen las posibles relaciones entre acciones o sucesos aparentemente independientes, es decir, mediante esta tarea se puede determinar cuando un suceso puede inducir a la aparición de otro. Con este fin se identifican relaciones no explícitas entre atributos categóricos o nominales. Estos patrones pueden servir para conocer el comportamiento general del problema que genera los datos a tratar [22]. Es importante señalar que estas reglas no implican relaciones de causa efecto ya que puede no existir una causa para que los datos estén asociados.
- Reglas de asociación secuenciales. Se consideran un caso especial de las reglas de asociación, pues tienen un concepto similar en el que influye además el factor tiempo, es decir, que permite reconocer el tiempo que transcurre entre el suceso inductor y los sucesos inducidos.
- Previsión. Busca establecer el comportamiento futuro más probable de una variable o una serie de variables a partir de la evolución pasada y presente de las mismas o de otras de las cuales dependan. Las técnicas asociadas a estas herramientas ya tienen un elevado grado de madurez.
- Correlaciones. Constituye una tarea descriptiva que se usa para examinar el grado de similitud de los valores de dos variables numéricas. El análisis de estas correlaciones puede ser muy útil

para establecer reglas de ítems correlacionados [20].

1.2.3 Técnicas de la Minería de Datos

Dado que la minería de datos está relacionada con muchos campos de estudio, existen diferentes paradigmas detrás de las técnicas utilizadas, como son: la inferencia estadística, árboles de decisión, redes neuronales, inducción de reglas, aprendizaje basado en instancias, algoritmos genéticos, aprendizaje bayesiano, programación lógica inductiva, y varios tipos de métodos basados en núcleo. Cada uno de estos paradigmas incluye diferentes algoritmos, así como otro tipo de restricciones que hacen que la efectividad del algoritmo dependa del dominio de aplicación, no existiendo un método universal que se pueda aplicar en todas las situaciones. A continuación se analizan los aspectos principales de algunas de las técnicas mencionadas [20]:

- Árbol de decisión: Constituye un conjunto de decisiones o condiciones organizadas en una estructura jerárquica, a modo de árbol. Los árboles de decisión siguen una aproximación “divide y vencerás” para partir el espacio del problema en subconjuntos. Encima del nodo raíz se tiene el problema a resolver, los nodos internos corresponden a particiones sobre atributos particulares, los nodos hoja representan la predicción del problema para todas las instancias que alcanzan esa hoja. Cada rama desde la raíz a las hojas se puede interpretar como una regla, siendo los nodos hojas la clase asignada y los nodos internos definen los términos de la conjunción que constituye el antecedente de la regla y la clase asignada en la hoja es el consecuente. Esta técnica se usa en tareas de clasificación, agrupamiento y regresión. Su uso ofrece grandes ventajas ya que son muy útiles para encontrar estructuras en espacios de alta dimensionalidad y en problemas que mezclen datos categóricos y numéricos.
- Inducción de Reglas: La Inducción de Reglas es un conjunto de métodos para generar un grupo de reglas comprensibles, de la forma:

  SI Condición *1* **Y** Condición *2* **Y…** Condición *n* **ENTONCES** Predicción

  El antecedente de la regla contiene una conjunción de n condiciones sobre los valores de los atributos independientes, mientras que el consecuente de la regla contiene una predicción sobre el valor de un atributo objetivo. Si se cumplen todas las condiciones que se especifican en el antecedente de la regla para una instancia del registro de datos, entonces el atributo al que se le predice un valor tomará el especificado en la regla.

Las reglas que se obtienen mediante esta técnica difieren de las que se producen a través de árboles de decisión en que son independientes, o sea, no tienen que formar un árbol, pueden entrar en conflicto en sus predicciones y además no cubrir todas las situaciones posibles.

- Redes neuronales: Constituyen una forma nueva de analizar la información con una diferencia fundamental con respecto a las técnicas tradicionales: son capaces de detectar y aprender complejos patrones y características dentro de los datos [23]. Su comportamiento es similar al cerebro del hombre, pues aprende (de la experiencia y del pasado) como resultado de un entrenamiento, lo que posibilita su gran adaptación y evolución ante una realidad cambiante y muy dinámica. Al igual que los árboles de decisión pueden usarse para tareas de clasificación, regresión y agrupamiento. Las redes neuronales trabajan directamente con datos numéricos por lo que si van a ser usadas para datos nominales estos deben ser convertidos primeramente a datos numéricos. La desventaja de este método reside en que el conocimiento se representa en una forma mucho menos explícita, en gran parte de los casos es incomprensible para las personas. Una red neuronal puede verse como un grafo dirigido con muchos nodos que representan los elementos del proceso y arcos entre ellos que representan sus interconexiones.
- Aprendizaje basado en instancias o casos: A través de esta técnica de minería se almacenan en memoria las instancias, de forma tal que al llegar una nueva instancia cuyo valor se desconoce se intenta relacionarla con las ya almacenadas, de las cuales se conoce su clase con el objetivo de hallar una similitud y poder estimar los valores de la misma; o sea, más que intentar crear reglas, trabaja de forma directa con los ejemplos. La comparación de cada nueva instancia se realiza a través de una métrica de distancia con las ya existentes y la instancia más próxima se usa para asignar su clase a la nueva instancia. El aprendizaje basado en instancia es muy útil a la hora de trabajar sobre tipos de datos no estándar como son: textos o multimedia.
- Técnicas bayesianas: Su funcionamiento se basa en estimar la probabilidad de pertenencia a una clase o grupo, mediante la estimación de probabilidades condicionales inversas o a priori, utilizando el teorema de Bayes.
- Técnicas algebraicas y estadísticas: Se basan en expresar modelos y patrones mediante fórmulas algebraicas, funciones lineales, funciones no lineales, distribuciones o valores agregados estadísticos tales como: la media, la varianza y correlaciones. Frecuentemente cuando se obtiene un patrón, se hace a partir de un modelo predeterminado del cual se estiman coeficientes o parámetros.

- Algoritmos evolutivos: Son métodos de búsqueda colectiva en el espacio de soluciones, es decir, dado una población de potenciales soluciones a un problema, la computación evolutiva expande esta población con nuevas y mejores soluciones.

1.2.4 Algoritmos de la Minería de Datos

Los algoritmos de la minería de datos pueden clasificarse en diferentes tipos, en dependencia de la función que realicen o a la fase a la que respondan [17]:

Algoritmos de Clusterizado.

Los métodos de agrupamiento deben definir una función útil de clasificación sobre un conjunto *Xi* (donde i=1,…, N) cuando N generalmente no es determinada y el número de grupo que se va a formar es desconocido. Por lo que los algoritmos de clustering utilizan una técnica basada en dos pasos donde un bucle exterior considera los posibles números de grupos y un bucle interior ajusta de la mejor manera los datos a ese número fijo de grupos.

Algoritmo vecinos más cercanos: Dado un número *k* de grupos, tiene como objetivo encontrar la mejor *k*-partición, de forma que los patrones de cada grupo de la partición estén más cercanos entre sí que los patrones de los otros grupos. Una vez determinada la partición se puede intentar representar cualquier nuevo patrón en función del más cercano. Existen variaciones de este algoritmo que se pueden agrupar en 4 grandes grupos:

- **Método de las K-medias:** Es el más sencillo de los algoritmos de agrupamiento habituales. Sean *p* vectores de *n* características o vectores característicos *Xj (j=1,…, p)* que pueden agruparse en *N* clases cada uno de sus miembros *Ni (i=1,…, N)*. Se eligen una serie de valores del espacio como centros, a partir de los cuales se empezará a generar clases o grupos. Cada vez que se presenta un patrón, se calcula su distancia a todas las medias y se le asigna la clase cuya media sea la más cercana. Se recalcula entonces la media de esta clase como el baricentro de todos los puntos que pertenecen a ella, incluido el último asignado de la forma siguiente:

$$M_i(t+1) = \frac{1}{N_i} \sum_{j=1}^{N_i} X_j$$

  Siendo *Xj (j=1,…, Ni)* patrones asignados a *Mt*; y se repite la operación tantas veces como puntos se quiera clasificar o hasta que la media en el paso *t+1* sea igual a la del paso *t*.

- **Método de K-NN o K vecinos más cercanos:** Es uno de los algoritmos más antiguos, surgió cuando se observó que un solo patrón provoca que la existencia de un único punto defectuoso desvíe la clasificación sin remedio. Durante la clasificación se calcula la distancia entre los patrones de entrada y los ejemplos almacenados. Se buscan los *k* ejemplares más cercanos y se asignan al patrón de entrada la clase más abundante entre estos *k* ejemplos.
- **Algoritmo LVQ (Learning Vector Quantization):** Sólo almacena un número controlable de patrones. El entrenamiento en este algoritmo se realiza en varias etapas. En primer lugar se determina el número de ejemplo a almacenar, generalmente con el método de K-Medias anteriormente mencionado u otro procedimiento de clustering. A partir de los ejemplares se asigna cada patrón de entrenamiento al ejemplar más cercano, penalizando si es la clase incorrecta y beneficiando si es de la correcta.
- **Método de las distancias encadenadas (Chain-map):** Consiste en elegir un vector característico al azar *Xi* de los *p* que se tiene y colocarlo en la primera posición de una lista. Después se coloca en posición siguiente de la lista el vector más cercano al primero. Se elige el siguiente más cercano al último de la lista, y así sucesivamente, quedando de la siguiente manera: *Xi(0), Xi(1), Xi(2), …, Xi(p-1),* donde *X(1)* es el vector más cercano al *X(0), X(2)* es el más cercano a *X(1)* y así sucesivamente. Una vez obtenido este valor se calculan las distancias euclídias entre ellos y se representan gráficamente.

Método de clusterizado de Montaña: Pertenece a las técnicas avanzadas de clusterizado. Consiste en crear una rejilla donde las regiones entre las intersecciones de las líneas son posibles candidatos a clusters, con centro en la intercepción de la línea. Después, se crea una función montaña que representa la densidad de datos de cada punto de la rejilla. Se selecciona el dato con mayor altura, se realiza una substracción de la montaña original con una montaña de centro en dicho dato y distribución Gaussiana, obteniéndose una nueva montaña. Luego se repite este procedimiento hasta que no quede ningún punto sin clasificar o la montaña que se obtenga tenga la altura menor que un umbral definido.

Algoritmos de Reglas de Asociación.

Los algoritmos de aprendizaje de reglas de asociación se basan en la búsqueda de reglas que cumplan unos requisitos mínimos de confianza y soporte o cobertura.

Algoritmo Apriori: es un algoritmo muy simple y utilizado. Se basa en la búsqueda de los conjuntos de ítems con determinada cobertura, para esto primeramente se construyen los conjuntos formados por un ítem que supere la cobertura mínima. Este conjunto de conjuntos se utiliza para construir el

conjunto de conjuntos de dos ítems, y así sucesivamente hasta que se llegue a un tamaño, en el cual no existan conjuntos ítems con la cobertura requerida.

Árboles de Decisión

Son unos de los algoritmos más empleados en minería de datos. Se basan en la partición del conjunto de ejemplo según ciertas condiciones que se aplican a los valores de las características, su potencia descriptiva viene limitada por las condiciones o reglas con las que se dividen el conjunto de entrenamiento. La construcción de un árbol de decisión se realiza de forma recursiva, primero se selecciona un atributo como nodo principal o raíz del árbol y a partir del mismo se divide el conjunto de observaciones en dos o más subseries de datos según el valor del atributo y se repite recursivamente para cada rama. Entre los algoritmos de árboles de decisión se encuentran CART, ID3, C4.5 (C5.0), SLIQ y M5.

CART: Realiza particiones binarias con una estrategia de poda basada en un criterio de coste y complejidad. Estas particiones binarias son el resultado de evaluar una condición que tiene dos únicas respuestas. La formulación de la regla de partición se realiza a partir de un conjunto estándar de preguntas.

ID3: También conocido como TDIDT, se basa en la entropía como función de impureza. La entropía o valor de información se define como la medida de incertidumbre que hay en un sistema, es decir, ante una determinada situación la probabilidad de que ocurra cada uno de los posibles resultados. A cada nodo se le asocia aquel atributo con mayor decrecimiento en la función de impureza que aún no se haya considerado en la trayectoria desde la raíz. Este algoritmo tiene entre sus inconvenientes su predisposición a favorecer indirectamente a aquellos atributos con muchos valores, los cuales no tienen porque ser necesariamente los más útiles.

C4.5: Este algoritmo y su extensión C5.0 es una extensión del ID3 que incluye varias mejoras como: la construcción de árboles de decisión cuando algunos de los ejemplos presentan valores desconocidos en algunos atributos, puede trabajar con atributos que presenten valores continuos, tolerancia a datos con ruido y generar reglas a partir de árboles.

SLIQ: Es un algoritmo creado para enfrentar problemas con grandes cantidades de datos, para la construcción del árbol *Tmax* utiliza la misma función de impureza que el algoritmo CART. El esquema utilizado en la fase de poda se fundamenta en el principio de longitud mínima MDL. MDL establece que

el coste total de la codificación de unos datos D mediante un modelo M viene dado por la suma del coste de bit de codificar los datos dado un modelo M y el coste de codificar el modelo M.

M5: Es un algoritmo desarrollado e implementado en Weka y es un árbol de regresión donde el final de cada rama, la clase se representa mediante el promedio del valor de las observaciones que han llegado hasta ella (de regresión) o mediante un modelo de combinación lineal (árbol de modelado). Este algoritmo tiene mecanismos para trabajar eficientemente con valores inexistentes, ruidos e incluso con valores nominales que son convertidos previamente en valores numéricos binarios.

Generadores de Reglas

Como los árboles de decisión en aplicaciones reales tienden a ser muy grandes y difíciles de interpretar se ha tratado de convertir estos en otras formas de representación como las reglas inducidas. Algunos de estos algoritmos son: AQ, CN2, RIPPER, INDUCT, PART, FOIL, CLINT.

AQ: Tiene sus raíces e influencia en métodos de ingeniería eléctrica utilizados para la simplificación de circuitos eléctricos. La estrategia seguida por el mismo es abajo-hacia-arriba donde inicialmente cada uno de los patrones de entrenamiento se considera un complejo. Luego estos complejos son examinados eliminando selectores y garantizando la consistencia del complejo resultante, de esta forma en cada etapa se construye un complejo y mediante la combinación de los complejos generalizados se construye un recubrimiento completo que cubre todos los patrones de una determinada clase. Su objetivo se basa en encontrar un conjunto de recubrimiento compacto que cubra todos los posibles casos. AQ permite encontrar un conjunto completamente consistente de reglas con todos los datos de entrenamiento, pero no puede clasificar correctamente con ruido ni considerar ninguna estrategia para evitar el sobreentrenamiento.

CN2: Se crea como una extensión del AQ permitiendo el tratamiento de ruido y sobreentrenamiento. Retiene un conjunto de complejos durante la búsqueda, de forma que estos complejos cubren un gran número de casos de una clase, aunque también pueden cubrir casos de otras clases. De forma adicional realiza un proceso de especialización, por lo que en cada paso de especialización se añaden nuevas preguntas o se elimina todo el complejo. En la búsqueda de los mejores complejos se utilizan dos tipos de heurística: significancia y bondad, donde significancia es el umbral por debajo del cual no se considera un complejo para ser seleccionada como mejor complejo y la bondad es una medida de la cualidad del complejo utilizada para establecer un orden entre los complejos candidatos a la inclusión final en el recubrimiento.

RIPPER: Conjunto con el CN2 y C4.5 han sido los métodos básicos de este tipo de algoritmos. Es muy parecido a C4.5 ya que genera una serie inicial de reglas, lo que luego en C4.5 son depuradas y en este algoritmo son combinadas. Genera reglas muy simples que luego se recombinan y reemplazan en otras más complejas.

INDUCT: Se basa en una versión más sencilla nombrada PRISM que usa AQ y ID3 así como su extensión C4.5 para generar reglas diferentes para cada clase a clasificar. Usa distribución binomial para determinar la bondad de una regla, lo que permite crear mejores reglas con datos con cierto porcentaje de ruido. Ha sido mejorado con el algoritmo RDR para incluir reglas con excepciones.

PART: Es un algoritmo generador de reglas basado en subárboles el cual está implementado en Weka, está basado en técnicas para generar árboles y reglas, de forma que genera subárboles que luego son convertidos a reglas. Es bastante robusto a ruidos y valores ausentes.

FOIL: Se basa en técnicas de aprendizaje con lógica de predicados ILP, esta lógica inductiva de predicado es especialmente útil cuando se disponen de una base de conocimiento que aplicar, de esta forma se puede alimentar al sistemas con una serie de reglas que ayudan a obtener nuevos patrones de comportamiento de los datos.

CLINT: Se basa en la construcción de un árbol que explica los ejemplos negativos, identifica las cláusulas que cubren los casos negativos, borra las cláusulas de la base de conocimiento y recompone la estructura de conocimiento estudiando las cláusulas positivas que habían sido cubiertas por la cláusula anterior.

1.2.5 Metodologías de aplicación de la Minería de Datos

Con el paso del tiempo y el desarrollo de proyectos de minería de datos se ha ido acumulando experiencia mediante la cual se han podido desarrollar diferentes metodologías que facilitan el proceso de la minería de datos. Algunas de estas metodologías son CRISP-DM(CRoss-Industry Standard Process for Data Mining), SEMMA (Sample, Explore, Modify, Model and Assess), Modelo de proceso de Minería de Datos de Two Crows, CRITIKAL (Client-Server Rule Induction Technology for Industrial Knowledge Acquisition from Large Databases) y Metodología SQL Server- 2005. El análisis de estas metodologías y sus principales características permitió concluir que todas estructuran la minería de datos en fases similares, que se encuentran interrelacionadas entre sí; y la describen de forma iterativa e interactiva, aunque las más utilizadas para realizar un proyecto de minería de datos son CRISP-MD y SEMMA.

Metodología CRISP-MD[7]

CRISP-MD es una metodología para el desarrollo de proyectos de minería de datos la cual se ha convertido en un estándar. Está descrita como un proceso jerárquico, el cual consiste en un conjunto de tareas descritas en cuatro niveles de abstracción: fase, tareas generales, tareas específicas e instancias de proceso. Es decir, cada fase consta de varias tareas generales, las cuales deben cubrir todas las posibles situaciones. Por otra parte, las tareas específicas representan la descripción de como las acciones de las tareas generales se deben desarrollar en situaciones específicas y las instancias del proceso representan un conjunto de acciones, decisiones y resultados del proceso de minería de dato que se lleva a cabo. Además se organiza de acuerdo con las tareas definidas de los niveles superiores [24].

Esta metodología propone 6 fases [25]: Análisis del problema, Análisis de los datos, Preparación de los datos, Modelado, Evaluación y Despliegue los cuales no tienen un orden fijo ya que en muchas ocasiones durante el desarrollo del proyecto es necesario volver atrás en algunas ocasiones, lo cual se muestra en la figura 2.

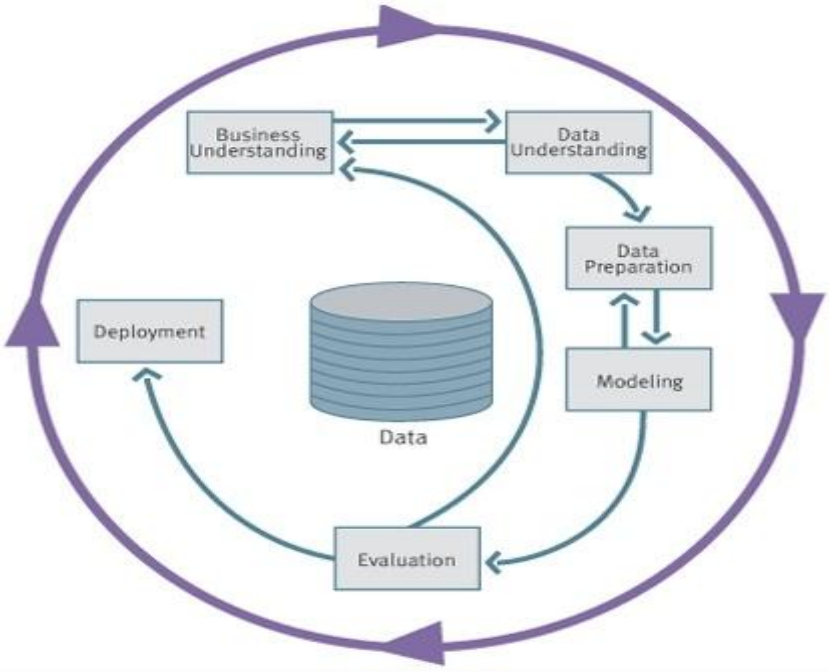

Figura 2 Fases de la Metodología CRISP-MD

[7] http://www.crisp-dm.org/

De manera general en estas seis fases con que cuenta la metodología CRISP-MD primeramente se seleccionan los objetivos y requerimientos del proyecto. Luego se realiza la recolección de los datos, se construye una base de datos a partir de estos, se seleccionan y aplican técnicas de modelado, y por último se evalúa el rendimiento del modelo creado y se continúa con el incremento de conocimiento obtenido de los datos.

Las ventajas de esta metodología son [26]:

- Concibe el proyecto de minería de datos de forma global y estrechamente relacionado al negocio en cuestión.
- Fue diseñada de forma neutra a la herramienta que se utilice para el desarrollo del proyecto.
- Es de distribución libre y se encuentra en constante perfeccionamiento por parte de la comunidad internacional.
- Presenta una precisa y sólida distribución de tareas de carácter general con sus resultados, así como una guía para su desarrollo.
- Muchas de las metodologías que se pueden encontrar en la actualidad se basan en este estándar.
- Es la que cuenta con mayor aceptación por parte de los desarrolladores de procesos de extracción de conocimientos a partir de datos.

Metodología SEMMA

La metodología SEMMA se define como el proceso de selección, exploración y modelado de grandes cantidades de datos con el fin de descubrir patrones de negocios desconocidos [21].

Esta metodología propone cinco fases: Muestreo, Exploración, Modificación, Modelado y Valoración. La ejecución de dichas fases no se plantea de forma rígida, pues no es necesario terminar una fase para comenzar otra. Además es necesario señalar que esta metodología no toma en cuenta los objetivos y requerimientos del negocio y tampoco la explotación de los resultados obtenidos.

En la figura 3 se representa la dinámica del sistema.

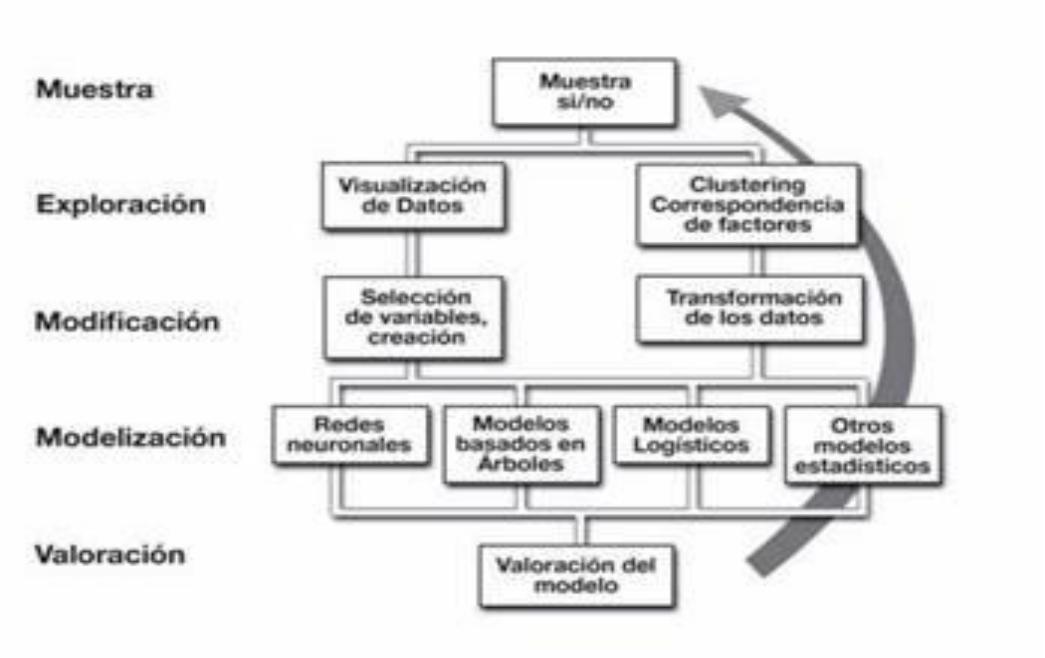

Figura 3 Fases de la Metodología de SEMMA

Es importante señalar que la metodología SEMMA se centra en las características técnicas del proceso ya que desde la primera fase se comienza realizando el muestreo de los datos, sin realizar un análisis previo del problema empresarial, para luego llevar a cabo su transformación en un problema técnico. Esta metodología en cuanto a su relación con herramientas comerciales, solamente es abierta en aspectos generales, debido a que se encuentra estrechamente vinculada a los productos de la compañía que la desarrollo.

1.2.6 Herramientas de Minería de Datos

En la actualidad existe una gran cantidad de herramientas tanto libres como comerciales para el desarrollo de proyectos de minería de datos. Entre las herramientas comerciales más usadas se encuentran IBM SPSS Modeler (anteriormente Clementine), Statsoft Statistica, SAS Enterprise Miner. Estas herramientas comerciales fundamentalmente abarcan métodos estadísticos y de visualización combinados con algoritmos bastantes rápidos. Generalmente disponen de entornos gráficos y suelen permitir a los usuarios realizar múltiples tareas.

IBM SPSS Modeler[8] posee una interfaz visual que permite interactuar de forma rápida con los datos. Permite un fácil acceso a los usuarios y preparar los datos en formato numérico, de texto o web para su modelado. Además se puede construir y comparar modelos rápidamente y a la vez distribuirlos eficientemente en tiempo real a las personas encargadas de la toma de decisiones. Tiene la capacidad de integrarse directamente a las bases de datos como ninguna otra herramienta. Soporta la metodología estándar para proyectos de minería de datos denominada CRISP-DM. Posee potentes herramientas de visualización y una gran variedad de técnicas de aprendizaje automático para clasificación, regresión, clustering y discretización [27].

Statsoft Statistica[9] ofrece un sistema efectivo y amigable de herramientas destinado al proceso de minería de datos y proporciona una eficaz plataforma para tareas de descarga de recursos intensivos de construcción de modelos, exploración basada en la web o ventanas de estaciones de trabajo de clientes y configuraciones centrales de consultas, análisis, plantillas de reportes y modelos [28].

SAS Enterprise Miner[10] es una herramienta proporcionada por SAS Institute que agiliza el análisis de minería de datos creando modelos descriptivos y predictivos de alta precisión basados en el análisis de una gran cantidad de datos. Facilita la implementación del modelo y proceso de calificación. Además posee una arquitectura distribuida en donde toda la funcionalidad del sistema es accesible mediante una potente interfaz gráfica de usuario. Su diseño está inspirado en la metodología SEMMA desarrollada por el mismo instituto [29]. Tanto el programa cliente como servidor de SAS Enterprise Miner puede ser trasladado a diferentes plataformas como Windows, Linux, Digital Unix, Solaris, etc.

Hoy en día existen una gran cantidad de herramientas de distribución libre las cuales han brindado aportes importantes en el campo de la investigación. Entre las más usadas se destacan las siguientes:

R es una herramienta para el análisis de datos basada en el programa estadístico S-Plus y con un manejo de las matrices y variables equivalente a MATLAB. Es muy útil para el análisis estadístico, transformación y manipulación de los datos. Además está compuesto de múltiples librerías para realizar gráficos y análisis estadísticos, regresiones lineales y no lineales, modelado, clusterizado y análisis de series temporales. Cuenta con una excelente asesoría técnica llevada a cabo generalmente por algunos de los principales profesores e investigadores en estadística en el mundo [17].

[8] http://www.spss.com/
[9] http://www.statsoft.cl/
[10] http://www.sas.com/

Weka[11] consiste en un conjunto de librerías en Java que contiene una colección de algoritmos de minería de datos los cuales permiten realizar tareas como preprocesamiento y filtrado, agrupamiento, reglas de asociación y visualización [30]. Es un software desarrollado bajo la licencia GPL como código abierto e incluye interfaz gráfica compuesta por diversos entornos, desarrollada por un grupo de investigadores de la Universidad de Waikato de Nueva Zelanda. Se destaca por la cantidad de algoritmos que presenta así como la eficacia de los mismos. Aunque la herramienta está implementada en Java no presenta problemas de portabilidad mientras que el sistema disponga de la máquina virtual adecuada.

Weka tiene 4 entornos de trabajo, el primero Simple CLI es un entorno de consola para con java invocar directamente los paquetes de Weka, el segundo es una interfaz gráfica conocida como Weka Explore en la cual se pueden ejecutar y configurar los algoritmos con los que cuenta esta herramienta, otro es el Experimenter el cual es un entorno centrado en la automatización de tareas de manera que se facilite la relación de experimentos a gran escala y el último KnowledgeFlow que permite generar proyectos de minería de datos mediante la generación de flujos de información [31].

Weka emplea el formato ARFF (Atribute-Relation File Format) como soporte de datos, en el que cada uno de los ficheros consta de una lista de instancias con los mismos atributos. Los tipos de datos que Weka permite son los numéricos, cadenas de caracteres, nominal y fecha [30].

Orange[12] es una herramienta para el análisis predictivo, la cual consta de una serie de componentes que implementan algoritmos de minería de datos, así como operaciones de preprocesamiento y representación gráfica de datos. Dichos componentes pueden ser manipulados desde programas desarrollados en Python o a través de un entorno gráfico bastante cómodo. Los algoritmos están programados en C++ y están interconectados con Python [32].

RapidMiner[13] es una herramienta para el análisis y minería de datos la cual permite a través de un entorno gráfico el desarrollo de procesos de análisis de datos mediante encadenamiento de operadores y se usa en investigación y en aplicaciones empresariales [33].

[11] http://www.cs.waikato.ac.nz/ml/weka/
[12] http://www.ailab.si/orange/
[13] http://www.rapid-i.com/

1.2.7 Aplicaciones de la Minería de Datos

En la actualidad existen un gran número de organizaciones enfrascadas en proyectos de minería de datos, su uso se hace cada vez más extensivo y sus productos de apoyo a la toma de decisiones han alcanzado una gran aceptación. Estas tecnologías pueden aplicarse a cualquier tipo de organización que disponga de grandes cantidades de datos y desee explotarlos para mejorar el servicio que presta.

Algunos ejemplos de aplicaciones ya clásicos en el campo de la gestión de la información son: la determinación de autores más productivos en ciertos temas, la contabilización de títulos y palabras claves en las publicaciones y el rastreo de referencias.

En el marketing se ha utilizado para la identificación del patrón de comportamiento de compra de los consumidores y la obtención de asociaciones a través de las características demográficas de los clientes [34]. En la medicina se ha utilizado para predecir la efectividad de procedimientos quirúrgicos, exámenes médicos y medicamentos [35].

Se pueden encontrar ejemplos en todo tipo de áreas en las cuales ha tenido un gran éxito la minería de datos: financieras, seguros, científicas (medicina, farmacia, psicología, etc.), políticas económicas o demográficas, educación, procesos industriales, policiales, etc. A continuación se incluyen ejemplos de las áreas antes mencionadas [20]:

Aplicaciones financieras y banca:

- Obtención de patrones de uso fraudulento de tarjetas de crédito.
- Determinación del gasto en tarjeta de crédito por grupos.
- Cálculo de correlaciones entre indicadores financieros.
- Identificación de reglas de mercado de valores a partir de históricos.
- Análisis de riegos en crédito.

Análisis de mercado y comercio:

- Análisis de la cesta de compra (compras conjuntas, secuenciales, ventas cruzadas).
- Evaluación de campañas publicitarias.
- Análisis de la fidelidad de los clientes.
- Segmentación de clientes.
- Estimación de stocks, de costes, de ventas.

En el ámbito médico la minería de datos tiene aplicación en varios campos [36]:

- En el ámbito clínico contribuye a la identificación y diagnóstico de patologías, así como al

descubrimiento de posibles interrelaciones entre diversas enfermedades.

- A nivel de medicina preventiva, ayuda en la detección de pacientes con factores de riesgos para sufrir una patología.
- A nivel de gestión hospitalaria se emplea para obtener predicciones temporales que permiten optimizar los recursos disponibles y priorizar el uso de diversos tratamientos para una misma patología.

Educación:

- Selección o capacitación de estudiantes.
- Detección de abandonos o de fracasos.
- Estimación de tiempo de estancia en la institución.

Telecomunicaciones:

- Establecimiento de patrones de llamadas.
- Modelo de carga en redes.
- Detección de fraudes.

Procesos industriales:

- Detección de piezas con trabas. Modelos de calidad.
- Predicción de fallos y accidentes.
- Estimación de composiciones óptimas en mezclas.
- Extracción de modelos de coste.
- Extracción de modelos de producción.

Otras áreas:

- Recursos humanos: selección de empleados.
- Tráfico: modelos de tráficos a partir de fuentes diversas.
- Deportes: estudio de la influencia de jugadores y de cambios.
- Policiales: identificación de posibles terroristas.
- Web: análisis de comportamiento de los usuarios, detección de fraude en el comercio electrónico.

Como se mencionó anteriormente, la minería da datos se ha empleado en diferentes áreas. A continuación se podrán ejemplos prácticos de las aplicaciones de la misma [37]:

- El FBI utiliza la minería de datos para analizar las bases de datos comerciales para detectar terroristas. Esto fue anunciado en julio del 2002 cuando el director del FBI comunicó que el departamento de Justicia comenzaría a introducirse en la enorme cantidad de datos comerciales, entre los que se revelan los hábitos y costumbres de la población, con el fin de identificar potenciales terroristas con antelación.
- Falcon Fraud Manager es un sistema inteligente que examina las transacciones, propietarios de tarjetas y datos financieros para intentar y eliminar el fraude financiero. En un principio estaba pensado para detectar fraude en tarjetas de crédito, pero actualmente también se utiliza para las tarjetas comerciales, de combustible y de débito. Es una sofisticada combinación de redes neuronales para analizar el pago mediante tarjeta y detectar los más remotos casos de fraude.
- La BBC (British Broadcasting Corporation) emplea un sistema para predecir el tamaño de las audiencias televisivas de un programa determinado, así como la hora óptima de emisión. Utiliza redes neuronales y árboles de decisión para aplicar los criterios que participan según el programa que hay que presentar y el contenido del mismo.
- El Second Palomar Observatory Sky Survey coleccionó durante seis años tres terabytes de imágenes que contenían alrededor de 2 millones imágenes de objetos en el cielo, con el objetivo de formar un catálogo con las mismas, por lo que se creó una herramienta conocida como SKYCAT (Sky Image Cataloguing and Analysis Tool) la cual emplea técnicas de agrupamiento y árboles de decisión para poder clasificar estos objetos en estrellas, planetas, sistemas y galaxias. Los resultados de este análisis han ayudado a los astrónomos a descubrir dieciséis señales radiales lejanas conocidos como quásares con corrimiento hacia el rojo que los incluye entre los objetos más lejanos de universo y más antiguo, los cuales son muy difíciles de encontrar y permiten entender el origen del universo.
- El AC Milán utiliza un sistema inteligente para prevenir lesiones y optimizar el acondicionamiento de cada atleta para lo que usa redes neuronales, lo cual alerta al médico del equipo de una posible lección de determinado atleta. El sistema fue creado por Computer Associates International, el cual se sustenta de los datos de cada jugador, relacionados con su rendimiento, alimentación y respuesta a estímulos externos, estos datos se obtienen de 24 sensores que son conectados al cuerpo del jugador mientras lleva a cabo determinadas actividades y que trasmiten señales de radio que luego son almacenadas en una base datos. Esto trae gran ventaja para este club ya que evita que compren jugadores que presenten una alta posibilidad de lesión.

- Los equipos de la NBA también utilizan aplicaciones inteligentes para apoyar a su cuerpo de entrenadores, un ejemplo de esto es el software Advanced Scout el cual emplea técnicas de minería de datos para detectar patrones estadísticos y eventos extraños. Posee una interfaz gráfica amigable para analizar el juego de los equipos de la NBA. Este software emplea todos los registros guardados de los diferentes eventos en cada juego como: pases, encestes, rebotes y doble marcaje con el objetivo de ayudar a los entrenadores a aislar los eventos que no pueden detectar cuando están viendo el juego o están viendo una retransmisión del mismo.
- Last.fm es una radio vía Internet y un sistema de recomendación de música que crea perfiles y estadísticas sobre gustos musicales, basándose en los datos enviados por los usuarios registrados. Esta radio permite seleccionar las canciones según las preferencias personales o de otro usuario. Es de código abierto y las recomendaciones de música son calculadas usando un algoritmo colaborativo de filtrado, así los usuarios pueden explorar las listas que aparecen en otros usuarios con gustos similares.
- Flickr es un servicio utilizado extensamente como dispositivo de fotos. Es un sitio web para organizar fotografías digitales. Este emplea técnicas de clustering para agrupar las imágenes por etiquetas, puede crear un perfil de usuario y encontrar gente alrededor del mundo con gustos similares y agregarlos a tu lista de contactos. Además almacena una colección sobre las mejores fotos que se van colocando en el servidor diariamente, así consiguen que estas sean de gran calidad según las visitas recibidas y las notas de otros usuarios.

1.3 Minería Web

Según Jaideep Srivastava, Robert Cooley, Mukund Deshpande y Pang-Ning Tan la minería web consiste en aplicar las técnicas de la minería de datos para descubrir y extraer de forma automática información de los documentos y servicios de la web. Mediante la minería web se puede realizar un estudio y análisis del comportamiento, acceso y modo de funcionar de la web. Además desempeña un papel fundamental en la prevención del proceso de gestión, mantenimiento, mejora y explotación de los sitios web. Se puede definir la minería web como el descubrimiento y análisis de información que involucra el uso de técnicas de la minería de datos orientadas al descubrimiento y extracción automática de información teniendo en cuenta el comportamiento y preferencias de los usuarios [38].

La minería web tiene diversas aplicaciones como son: la personalización, que consiste en presentar a cada usuario la información que más le interesa de acuerdo con su perfil, la mejora del sistema aumentando el rendimiento a partir del análisis del tráfico, la detección de errores y la detección de

intrusiones o fraudes. Permite la modificación del sitio, es decir, reestructurar los contenidos a partir del comportamiento de los usuarios, así como proporcionar datos acerca de cómo se puede mejorar la forma de escribir los elementos o recursos que se publican en Internet, de forma que sean más atractivos para el usuario y si los temas que se tratan en estos son interesantes o no. Además brinda la posibilidad de saber si la estructura del sitio web es la más adecuada y qué se puede hacer para mejorarla.

Al navegar por sitios web los usuarios dejan registros de todas sus acciones, algunas de las huellas que dejan son las direcciones IP y los navegadores que se almacenan automáticamente en los servidores como una bitácora de acceso (registro web). Las herramientas de la minería web analizan y procesan estos registros web para producir información significativa acerca de los usuarios y sus tendencias como por ejemplo qué páginas son accedidas desde otras páginas y qué usuarios acceden a qué páginas. También se utilizan programas de estadísticas como los anteriormente mencionados awstats, webtrends o clicktracks para sacar conocimientos de cómo se puede mejorar un sitio, ya que ofrece información estructurada y significativa acerca de la navegación como [39]:

- Cantidad de visitas por horas, por día, por mes.
- Horas pico y horas de baja audiencia.
- Páginas más visitadas.
- Páginas de entrada y salida más frecuentes del sitio.
- Uso del buscador, ranking de palabras claves usadas para llegar.

Las herramientas utilizadas en la minería web son sistemas inteligentes que trabajan tanto del lado del servidor como del lado del cliente, para poder cubrir toda la información que se crea con el uso de Internet. A continuación se abordarán un poco más sobre estos tipos de herramientas [40]:

Las herramientas que se incorporan al propio servidor son programas que procesan en tiempo real los datos que están almacenados en los archivos de registro web. Al correr en el servidor el acceso a la información del tráfico tanto gráfica como estadística se realiza a través de una interfaz en línea. De forma general este tipo de soluciones vienen incluidas en ofertas de alojamiento web, ya sea en servidores dedicados o compartidos. Entre estas herramientas se encuentran: OmniAnalyzer[14], AWStats, Deep Log Analizar V 3.1[15], Advanced Log Analyzer, WebLog Expert[16].

[14] http://www.hypersoft.com/analyser.htm/
[15] http://www.deep-software.com/
[16] http://www.weblogexpert.com/

Las herramientas que trabajan en máquinas personales son software que se instalan de manera independiente en las computadoras de escritorio, con el objetivo de analizar los archivos de registro web, pero no en tiempo real, por lo que para la misma hay que descargar primeramente los registros web y luego procesarlos. Esto brinda una gran ventaja porque sin requerir acceso a Internet se pueden desarrollar informes a fondos sobre estadísticas en poco tiempo. Como ejemplo de este tipo de software se encuentran algunos comerciales como: DB Miner[17] y Speed Tracer, y públicos como SYstat[18] y Analog.

Cada una de las herramientas de minería web tiene su propósito específico como es: el análisis sobre el uso de la tecnología, el nivel del conocimiento que se maneja en una organización, estadísticas de ventas y usabilidad. Además cada uno tiene requerimientos técnicos como: la capacidad de memoria, sistema operativo y capacidad en el disco duro, así como brindan diferentes resultados [41].

1.3.1 Clasificación de Minería Web

En la actualidad la World Wide Web es el repositorio de información más grande y diverso que existe, del cual se puede extraer una gran cantidad de conocimiento relevante y útil. Este proceso de minería web no es un problema sencillo puesto que muchas páginas web contienen datos multimedia (texto, imágenes, vídeos y audio). Además los datos pueden residir en diversos servidores o en archivos como los que contienen los registros web. Otros aspectos que dificultan la minería web son cómo determinar a qué páginas se debe acceder y cómo seleccionar la información que va a ser útil para la extracción de conocimiento. Toda esta diversidad hace que la minería web se divida en tres tipos de actividades diferentes dependiendo de que parte de la web se esté explotando las cuales son: Minería Web de Contenido, Minería Web de Estructura y Minería Web de Uso [42].

[17] http://www.dbminer.com/
[18] http://www.systat.com/

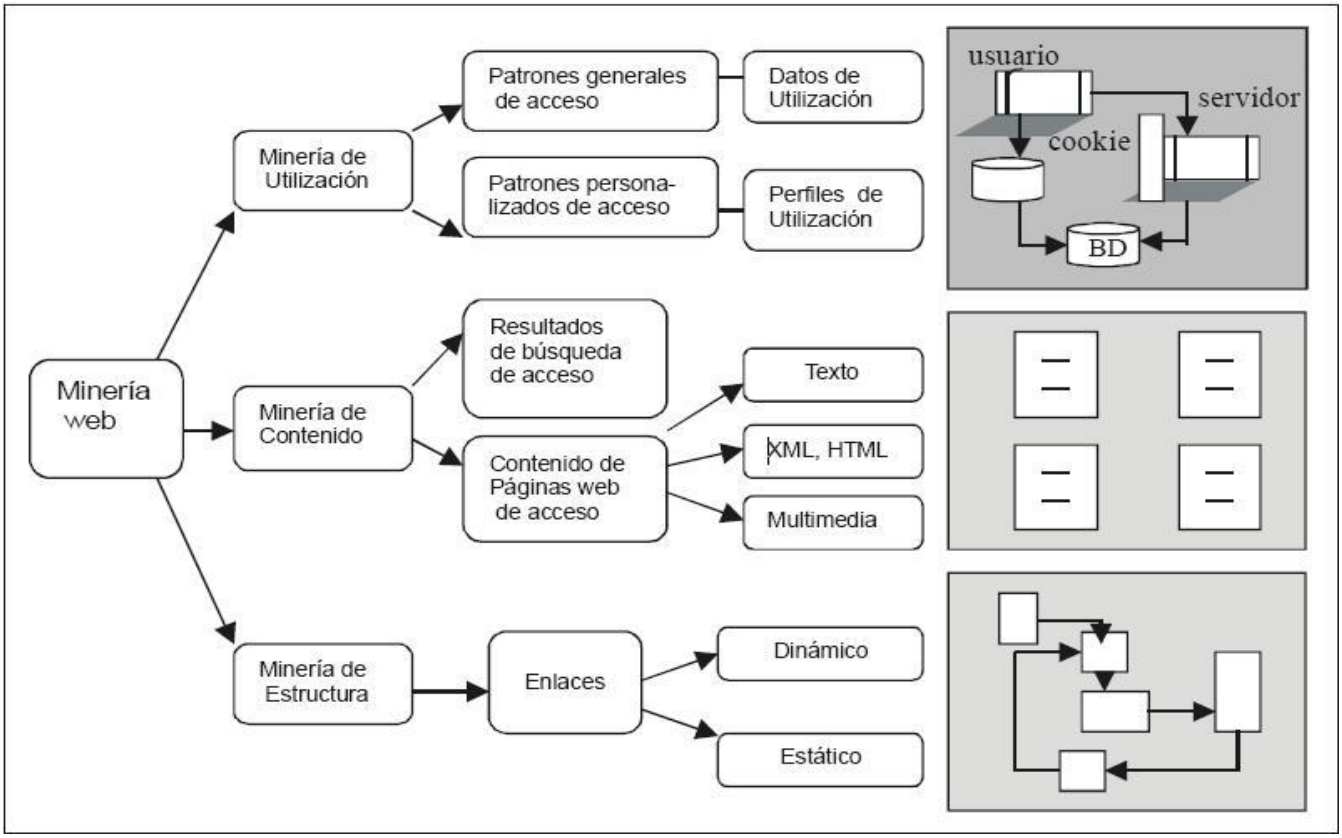

Figura 4 Categorías de la Minería Web

La minería web de estructura se enmarca en los procesos cuyo objeto es extraer información sobre la topología de la web, es decir los enlaces entre las páginas [43]. Muchos autores no incluyen este tipo de minería web en su clasificación.

La minería web de contenido engloba los procesos cuya finalidad es extraer información a partir de la información contenida en un determinado portal, esta puede ser en cualquier tipo de formato que contenga dicho portal.

La minería web de uso se basa en el proceso de extracción de patrones interesantes de la información del tráfico en la web, es decir, información acerca de los comportamientos de los usuarios en la web. Este tipo de minería es el objeto del presente trabajo por lo que será analizada exhaustivamente.

De manera general se puede señalar que para llevar a cabo la minería web de estructura se utiliza el conjunto de hiperenlaces que unen las diferentes páginas web, así como para realizar la minería web de contenidos se parte del texto almacenado en dichas páginas. Pero estas van a conformar una representación invariable de la web por lo que se puede llegar a realizar aproximaciones de la realidad

que no sean del todo confiables en cualquier momento, ya que tanto el contenido como los enlaces de las páginas cambian constantemente.

1.3.2 Minería Web de Contenido

La minería web de contenido es el proceso que consiste en la extracción de conocimiento del contenido de documentos o sus descripciones [4]. Su objetivo fundamental es extraer información útil de los contenidos de un sitio o página web. Los contenidos pueden dividirse en dos grandes grupos: el contenido de los resultados de búsquedas en la web y los contenidos incluidos en las páginas web. Estos contenidos pueden incluir cualquier tipo de información entre los que se encuentran: los datos de páginas HTML, datos multimedia, datos XML y los textos. Para ser más específicos, el contenido de un sitio web se encuentra de forma no estructurada, semi-estructurada y estructurada [44]. La minería web de contenido se clasifica en [35]:

- Text Mining: Trata fundamentalmente sobre las técnicas de recuperación de información, técnicas estadísticas y lingüísticas.
- Hypertext Mining: No se refiere solamente a los enlaces entre documentos, sino también a enlaces intradocumentos. Se realiza con la ayuda del grafo de referencia.
- Multimedia Mining: Es un campo poco desarrollado, principalmente se realiza sobre bibliotecas de imágenes.
- Markup Mining: Se utilizan las marcas que contienen información (las marcas HTML contienen tablas negritas, secciones, cursivas; las marcas XML contienen aún más información).

La minería web de contenido considera procesos selectivos e intensivos de recuperación de información, extracción de información y minería de texto para obtener información valiosa y específica en los contenidos de los documentos publicados en la web. Este tipo de minería según Raymon Kosala y Hendrick Blockeel [45] puede ser vista desde dos puntos de vista, desde la Recuperación de Información y desde la Base de Datos.

La Minería Web de Contenido desde el punto de vista de la Recuperación de la Información y la Extracción de la Información.

En la actualidad grandes empresas del mundo web que poseen servicios de máquinas de búsquedas como google, directorios jerárquicos como yahoo, entre otros tipos de sistemas de filtrado colaborativo emplean la recuperación de información. Esta se diferencia de la extracción de la información en que recupera documentos relevantes de una colección y sin embargo la extracción de información recupera

la información relevante de estos documentos, por lo que una técnica complementa a la otra y utilizándolas las dos se podrá obtener una información de gran valor.

Inicialmente la indexación de texto para hacer más simple la búsqueda de documentos útiles en una colección, era la idea inicial de la recuperación de la información. Actualmente la recuperación de la información incluye: modelado, clasificación y categorización, arquitectura de sistema, interfaces de usuario, visualización de datos, filtrado y lenguaje.

La minería de Texto o Text mining mencionada con anterioridad principalmente hace referencia a la extracción de la información y conocimiento interesante, no trivial desde documentos. La Text Categorization, Text Clustering, Association Analysis y Trend Prediction se encuentran entre las principales categorías de la Minería de Texto.

La Text Categorization se basa en que dada determinada clasificación, cada documento de una categoría es clasificado dentro de una o más de una clase apropiada. Entre los algoritmos de la Text Categorization se encuentran: K-nearest, neighbot-algorithm y naive bayes algorithm.

El objetivo de Text Clustering es dividir en un conjunto de clústeres una colección de documentos de manera tal que la similitud extra-clúster se maximice y la intra-clúster se minimice, esto se puede emplear a los documentos extraídos de una máquina de búsqueda. Se clasifica en dos tipos: clustering jerárquico y clustering particional [21].

Minería Web de Contenido desde el punto de vista de Base de Datos (BD).

El objetivo principal de la minería web de contenido desde el punto de vista de Base Datos es representar los datos a través de grafos etiquetados.

En los últimos años la cantidad de datos semi-estructurados y estructurados publicados en la web han crecido considerablemente, lo cual se ha visto reflejado en las páginas ocultas “hidden Web”, generadas automáticamente por las consultas hechas por los usuarios a partir de los datos guardados en las base datos. A raíz de esto, surgen herramientas y aplicaciones para la extracción de información importante en las páginas ocultas.

Actualmente se emplea para esta extracción de información los llamados “wrappers” los cuales no son más que procedimientos para la extracción de contenido [44].

Los wrappers extraen información de páginas web u otras fuentes semi-estructuradas escritas en un formato específico. Un wrapper sobre fuentes web es considerado un software que acepta consultas de usuarios de datos en la Web y después de extraer la información relevante retorna los resultados [46]. Entre las ventajas que presenta la construcción de wrappers en la Web se encuentran [38]:

- Consultas en las fuentes similares a las realizadas sobre bases de datos.
- Todas las fuentes sobre la que se construyó el Wrapper pueden ser procesadas usando un lenguaje común de consulta.
- El acceso integrado se puede hacer usando un mediador que integra la información desde las diversas fuentes.

1.3.3 Minería Web de Estructura

La minería web de estructura trata de mostrar cómo están relacionados los hiperenlaces entre las distintas páginas teniendo en cuenta dos tipos de enlaces: estáticos y dinámicos. Además proporciona información acerca de si los usuarios encuentran la información que buscan, si la estructura del sitio es ancha o demasiado profunda y si los elementos están colocados en los lugares adecuados dentro de la página [39]. Este tipo de minería hace un análisis de la navegabilidad de los sitios y de cómo se interconectan con otros. Para realizar la minería web de estructura es necesaria la utilización de grafos, lo cual permite reflejar el movimiento entre enlaces al navegar de una página a otra y así tener una mejor visión del conocimiento obtenido.

La minería web de estructura es el proceso que analiza la estructura de la información usada, que describe el contenido de la web. La estructura de la información de la web puede ser clasificada como: intra-página o inter-página. La estructura de información inter-página se puede analizar a través de los hiperenlaces. El enlace de estructura puede representarse por un grafo, en el cual los nodos son los documentos web y las aristas los hiperenlaces, precisamente de esta relación entre los nodos y las aristas se puede obtener información muy útil. Por otra parte la estructura de información intra-página se refiere a las estructuras internas de los documentos de la web, los cuales están usualmente representados por árboles [47].

La estructura presenta una gran cantidad de información para descubrir y ser analizada. Se pueden considerar dos tipos de descubrimiento de páginas o topologías llamadas Hubs y Autoridades. Una autoridad puede considerarse como páginas altamente referenciadas en un tema específico como se muestra en la figura 5. Mientras que un Hub puede definirse como el conjunto de páginas comparables para muchas relaciones de autoridad como se muestra en la figura 6. Los Hubs y las autoridades

tienen una relación muy fuerte entre ellos, debido a que un hub adquiere un mayor peso cuando se acopla a una autoridad y una autoridad adquiere un mayor peso cuando se asocia a muchos hubs [48].

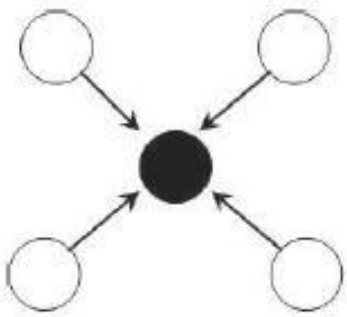

Figura 5 Representación de Autoridad

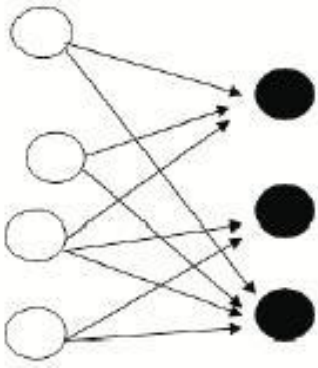

Figura 6 Representación de Hub

Para modelar la topología de la web se utilizan algoritmos como el PageRank y HITS (Hiperlinks-Induced Tepic Search). El PageRank asigna valores a las páginas web para poder ordenarlas según su categoría o importancia, basándose en el grafo de la web. La idea principal consiste en que las páginas web a las que apuntan otras páginas serán más importantes que aquellas que tienen pocos enlaces entrantes, además se tiene en cuenta la importancia de la página que te apunta. De manera tal que una página será importante si tiene muchos enlaces o si tiene pocos enlaces de páginas importantes.

El HITS es un algoritmo interactivo que alinea las páginas en dos tipos que guardan una relación de mutua dependencia: autoridades y hubs. La idea principal consiste en que cuando alguien establece un enlace a la página es porque la considera interesante y las personas con intereses comunes tienden a referirse a las autoridades sobre un tema dentro de una misma página [49].

1.4 Minería Web de Uso

La minería web de uso captura y modela los patrones de comportamiento y los perfiles de los usuarios al interactuar con un sitio web [50]. Tales patrones son usados para entender las características del comportamiento de los usuarios y así mejorar la estructura del sitio o crear una experiencia personalizada para los visitantes. Este tipo de minería tiene múltiples aplicaciones que van desde la mejora del diseño del sitio web, hasta la optimización de las relaciones entre clientes y responsables del sitio en cuestión.

1.4.1 Etapas de la Minería Web de Uso

Este proceso contempla cuatro etapas fundamentales: la recolección de la información, preparación y transformación de los datos, el descubrimiento de patrones de uso y el análisis de patrones de uso (Figura 7).

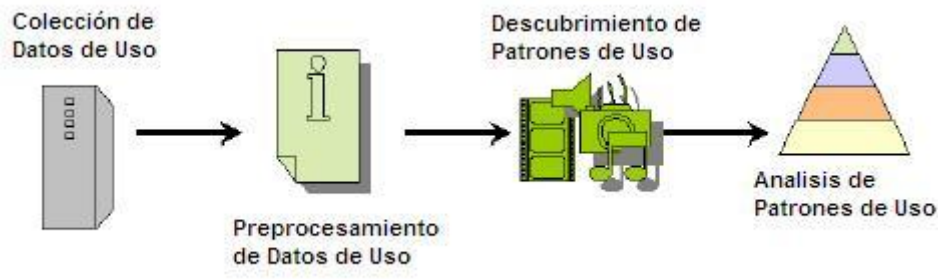

Figura 7 Etapas de la Minería Web de Uso

La recolección de la información se puede ejecutar a nivel del servidor web que hospeda las páginas, a nivel del servidor proxy del lado cliente o en el nivel de los agentes del cliente web [51]. En estos se generan constantemente un gran volumen de datos debido a los requerimientos de los usuarios que son almacenados en los Registros de Acceso Web.

Una vez obtenida la información se realiza la preparación de los datos o preprocesamiento de los registros web y probablemente esta sea la parte más difícil en todo el proceso de minería web por la complejidad y el tiempo que requiere. Cuenta con varias etapas entre las que se encuentran: combinación de fuentes de información, limpieza de datos, identificación de páginas visitadas, identificación de usuarios, identificación de sesiones, identificación de transacciones y la elaboración de inferencias sobre las referencias a enlaces perdidos durante el almacenamiento de los archivos en la memoria caché de los servidores proxy o dispositivos de administración de contenidos [51].

En el proceso de transformación de la información se crean grafos, tablas y estructuras de datos que son requeridas por algoritmos utilizados en el descubrimiento de patrones de uso. Cuando la información ha sido transformada se inicia el descubrimiento de patrones mediante algoritmos que pueden hacer uso de reglas de asociación, agrupamiento de ítems, clasificación, rastreo de rutas para crear patrones secuenciales y modelado de dependencias [52].

1.4.1.1 Recolección de los Datos de Uso

La recolección de los datos implica fuentes de información de diversa naturaleza como son: los datos de los registros de servicios del servidor web, los datos de los registros de servicios del servidor proxy y los datos de los registros de servicios de la máquina cliente.

Los servidores web almacenan la información de acceso en los registros de acceso web, los cuales se consideran el recurso más importante para el proceso de la minería web de uso. Los datos se pueden almacenar en un Formato Común de Registro (CLF) o en un Formato Extendido de Registro (ELF), los datos almacenados en CLF contienen los siguientes campos [53]:

- Número IP o nombre del host remoto que realiza el acceso.
- Nombre del usuario que accede remotamente.
- Nombre de usuario bajo el cual se ha identificado.
- Fecha y hora en la que el usuario realiza la solicitud del servicio.
- La solicitud como se realizó exactamente por el cliente.
- El código de estado HTTP que se devolvió al cliente.
- La cantidad de información (en bytes) que se transfiere.

Los datos guardados en ELF contienen además de la información anterior los siguientes campos:

- El servidor del que proviene el cliente
- La información de identificación que el navegador del cliente incluye en sí mismo.

Un servidor proxy permite el acceso a la Web a varios equipos a través de una única dirección IP y posee un cortafuego el cual restringe el acceso para una red protegida. Por tanto, el registro de acceso web del servidor proxy puede ser utilizado como fuente de datos para caracterizar el comportamiento de navegación del usuario.

Los datos del registro de acceso web residen en el cliente, así que este es el mejor recurso para descubrir patrones del cliente, donde las cookies son una potente herramienta empleada por los

servidores web para almacenar y recuperar información acerca de sus visitantes.

Una página web que se muestra en un navegador es un objeto complejo, compuesto por numerosos marcos y cada uno de ellos muestra contenidos de un fichero HTML diferente, los cuales a su vez contienen referencias a varios ficheros de imágenes. Para construir cada una de estas páginas cada uno de estos objetos tienen que ser solicitados a uno o varios servidores web. En caso que el contenido necesario esté disperso por varios servidores esto trae como consecuencia que al realizar un link a dicha página se creen múltiples accesos al sitio, es decir, uno para cada objeto que contiene la página y son registrados en ficheros diferentes. Por tanto, antes de analizar los registros web para la extracción de conocimiento acerca del comportamiento de los usuarios, es necesario realizar un grupo de acciones que permitan utilizarlos de una forma eficiente.

1.4.1.2 Preprocesamiento de los registros web

Esta etapa comienza con la combinación de diferentes fuentes de información de la misma naturaleza y continúa con la limpieza de los datos en la cual se eliminan todos los registros innecesarios o redundantes. Cada página visualizada por el usuario genera más de una petición al servidor por ejemplo: un fichero HTML, las imágenes, gráficos, vídeos, las hojas de estilo CSS y los ficheros de Java Script. Generalmente, la única información relevante es la petición que hace referencia al fichero HTML, pues fue la que explícitamente realizó el usuario, por consiguiente se debe eliminar toda la información restante.

Otra de las acciones que se deben llevar a cabo en esta etapa es la eliminación de los usuarios falsos que son aquellos usuarios que no son útiles para la investigación, como por ejemplo, los administradores y desarrolladores de los sitios web. Estos usuarios no son de gran utilidad a la hora de mejorar un sitio web mediante la individualización de los servicios por lo que es necesario encontrar aquellos usuarios irreales o falsos y eliminarlos del fichero de registros web.

Para poder aplicar la minería web de uso es imprescindible contar con los datos a nivel de secciones de usuarios, pero para poder reconocer o identificar estas secciones es necesario anteriormente realizar una identificación de usuarios [43]. Un dato importante que identifica a un usuario es la dirección IP, pero esta información no resulta suficiente pues existen mecanismos intermedios como son los servidores proxy que ocasionan que distintos usuarios reales aparezcan con la misma dirección, por ejemplo en una empresa varios trabajadores pueden acceder a un sitio web con la misma dirección IP.

El método ideal para identificar fácilmente a los usuarios que acceden a un sitio web sería que estos lleven a cabo su registro en el sitio y se autentiquen mediante un nombre y una contraseña cada vez que accedan a él. Pero la web es anónima y los usuarios entran y salen de los sitios web sin más testigos que la información almacenada en los servidores web, por lo que es necesario aplicar varias estrategias para identificar a un usuario a partir de varios accesos a un determinado sitio. En los ficheros de registros web, además de guardarse las direcciones IP desde las cuales se acceden al sitio también se guarda una cadena de caracteres que identifica el navegador utilizado, es decir, que si se accede a una página desde una misma dirección IP pero se utilizan diferentes navegadores puede corresponder a diferentes usuarios.

Otra manera de identificar usuarios diferentes consiste en analizar el camino que siguen dentro de la estructura de contenidos, si se accede a diferentes páginas que no tienen ningún enlace una con la otra desde la misma dirección IP entonces se puede estar en presencia de usuarios diferentes. Esta técnica resulta muy difícil de implementar y además es poco fiable ya que los usuarios pueden dar saltos que no sigan con la navegación de un sitio, por ejemplo usando direcciones almacenadas como “favoritos” [30].

En ausencia de un formulario de registros de usuario muchos sitios utilizan las cookies conocidas también como huellas, para intentar reconocer los usuarios que han accedido al sitio, pero para que las cookies cumplan su función es necesario que el usuario esté utilizando el mismo navegador desde la misma computadora y que no haya deshabilitado o eliminado las cookies [52]. Además cuando dos empleados de la empresa acceden al mismo sitio, la cookie del primer usuario que se envió desde el servidor es reconocida cuando el segundo usuario es el que accede, esto provoca un error de identificación. Otro problema sería en el caso cuando un mismo usuario accede a un mismo sitio desde diferentes lugares. Sin embargo esto no implica que no se puedan utilizar las cookies para la identificación de usuario, se emplearían mayormente cuando un usuario se encuentre previamente registrado, a los cuales se le realiza algún tipo de información o se personaliza la apariencia del portal [43].

La identificación de sesiones no es más que el proceso por el cual se determina que una serie de servicios solicitados por un mismo usuario pertenece a una única visita a un portal web [42]. Consiste en dividir los accesos de un determinado usuario en sesiones de navegación, agrupando los pertenecientes a una misma sesión de trabajo y que se han realizado de forma ininterrumpida por parte del usuario. Las sesiones también pueden ser llamadas como visitas. El método más sencillo es

utilizar un tiempo límite de modo que si el tiempo transcurrido entre dos peticiones consecutivas supera ese límite se considera que ha iniciado una nueva sesión, generalmente este tiempo es de 30 minutos [30]. La mejor forma de aplicar esto es que cuando un determinado usuario visite por primera vez el sitio se cree un identificador de sesión, esto puede realizarse alterando las URLs o mediante las cookies.

Basándose únicamente en los datos de los ficheros de registros web, cualquier método de determinación de secciones no puede determinar realmente cuanto tiempo un usuario se encuentra en determinado portal ya que pudo haber minimizado el navegador y no esté realmente consultando el servicio que en el fichero se está registrando.

El siguiente paso es la determinación de la secuencia de navegación cuyo objetivo es detectar la navegación de los usuarios por páginas que no quedan registradas en los registros de los servidores debido a la utilización de las memorias temporales (caché) en los navegadores. Tener la página guardada en la caché del navegador trae como consecuencia que no se solicite al servidor web y por tanto la solicitud no queda registrada en los ficheros de registros web.

Algunos de los efectos de la caché sobre los ficheros de registros de los servidores son [43]:

- El número de páginas solicitadas es bajo.
- Las solicitudes de diferentes usuarios con un proxy en común no pueden distinguirse a través de la dirección IP ya que todos ellos acceden a través de la dirección IP del servidor proxy.
- Las secuencias guardadas son incompletas. Por ejemplo, suponiendo que un usuario visita la página A, luego accede a una B y de esta vuelve nuevamente a la A, para después acceder a una C. En el fichero de registros web solamente se guarda el acceso a la página C, seguido del acceso a la página B, ya que el retorno a la página A no se guarda porque se encuentra almacenada en la caché.

Retomando el ejemplo anterior, si no existe ningún vínculo de la página B a la página C, entonces se puede inferir que se retornó después de ser visitada la página B a la página A. Este tipo de proceso es llamado determinación de la secuencia y se basa en completar el camino seguido por un usuario al visitar un portal web.

El uso de páginas dinámicas facilita el proceso de determinación de la secuencia, pues cada página dinámica creada es única y por tanto no se almacenan en la caché de los navegadores. Muchos sitios en la actualidad las utilizan, generalmente los sitios web destinados al comercio electrónico.

Por último es necesario dejar los datos preprocesados en un formato que sea utilizable en el siguiente paso de la minería web de uso, que sería el descubrimiento de patrones para la extracción de conocimiento útil.

1.4.1.3 Descubrimiento de patrones de Uso

Una vez que los datos están organizados por usuarios y sesiones y son manejables se pasa a la extracción de conocimiento. Este paso se realiza aplicando algoritmos provenientes de diferentes campos de estudio como son: la Estadística, la Minería de Datos y el Reconocimiento de Patrones a la información procesada anteriormente.

El **análisis estadístico** es el método más usual para extraer información acerca de los usuarios de un sitio web, que permite llevar a cabo análisis acerca del tráfico web, incluyendo servicios más solicitados, tamaño medio de los ficheros transferidos y número de visitantes de un sitio web. De manera general estos análisis a pesar de ser poco elaborados pueden ser muy útiles a la hora de estudiar el rendimiento del sistema, comprobar la seguridad del mismo y facilitar la modificación del sitio.

Las técnicas que se emplean para el análisis estadístico se pueden clasificar en las técnicas descriptivas y en las basadas en inferencias. Las descriptivas son utilizadas para sintetizar datos mediante el cálculo de promedios, media, moda, varianza, desviación estándar y desviación absoluta [52]. Las basadas en inferencias realizan suposiciones sobre información desconocida haciendo uso de probabilidades de frecuencias e inferencia Bayesiana.

Las **reglas de asociación** consisten en descubrir correlaciones entre un conjunto de páginas que son visitadas en una misma sesión. Básicamente se asigna un valor de importancia a cada página dependiendo del número de accesos registrados a esta y para que esa página sea tenida en consideración, su valor de importancia debe ser mayor que uno definido con anterioridad. Las reglas obtenidas representan el comportamiento de los usuarios y esta información es muy importante a la hora de decidir la estructura del sitio y para predecir que páginas visitará el usuario con mayor probabilidad lo que permite realizar una precarga y servirlas de manera más rápida [51].

El **agrupamiento** consiste en reunir información de características similares, por ejemplo: agrupar páginas similares o detectar usuarios con patrones de comportamiento afines. Los algoritmos de agrupamiento se enmarcan dentro de los sistemas basados en aprendizaje no supervisados, es decir, que no se conoce la clase a la que pertenecen los patrones de entrenamiento [43]. Esto se basa en

determinar la organización de los patrones en grupos o clústeres que permiten descubrir semejanzas y diferencias entre los patrones, así como extraer conclusiones sobre el problema [54].

La **clasificación** busca determinar los patrones de navegación que siguen los usuarios y crear categorías de acuerdo a su comportamiento, es decir, que dado un patrón de usuario este puede clasificarse en diferentes categorías. Para esto es común la utilización de redes neuronales, sistema de lógica difusa, modelos vectoriales y algoritmos genéticos.

La **extracción secuencial de patrones** trata de descubrir patrones dentro de una misma sesión, básicamente, series de páginas que son visitadas siguiendo una determinada frecuencia [30]. Con la utilización de este tipo de técnica pueden encontrarse patrones periódicos que resultan muy útiles para determinar las tendencias del comportamiento de los usuarios para detectar puntos de cambio y también para el análisis de similitudes. Además se pueden predecir futuras visitas, lo cual es muy útil en la colocación de anuncios dirigidos a determinados grupos de usuarios.

El **modelado de dependencias** tiene como objetivo principal construir un modelo que represente dependencias significativas entre variables del entorno web, por ejemplo los diferentes estados por los que pasa un usuario que realiza una compra en una tienda virtual, además de las probabilidades de pasar de un estado a otro [51].

1.4.1.4 Análisis de patrones de Uso

Un proceso importante dentro de todo el proceso de la minería web de uso es el análisis de patrones y este va a depender de los objetivos que se quieran alcanzar. Debe ser lo suficientemente efectivo como para descubrir información no visible a simple vista, presentar resultados de manera que sea fácil identificar las acciones que realizan los usuarios y sus tendencias. Para llevar a cabo este paso se pueden utilizar técnicas de visualización como gráficos o uso de colores para destacar determinados resultados, de filtrado de información y herramientas de minería de datos.

1.4.2 Herramientas para la Minería Web de Uso

Hoy en día existen varias herramientas para el análisis de registros web para extraer información significativa de la interacción del usuario mientras navega por la web. Por lo que se hará un breve análisis de algunas como: WUM, WebMiner y Weka.

WUM es una herramienta basada en Java para el análisis de los patrones de navegación de usuarios en un sitio web. Consta con un entorno integrado (ver figura 5) para la preparación de registros web,

tratamientos de sesiones y visualización gráfica de resultados. Permite importar varios formatos en la fase de preprocesamiento de registros web como son: Common y Extended, los cuales son variantes del formato de archivos de registros web ECLF. Además posibilita realizar tareas como: la exclusión de acceso que contenga determinadas cadenas, situación de literales y utilización de conceptos en vez de URL. Luego de la importación de los registros web, WUM pasa a la extracción automática de sesiones de usuario por tiempo máximo en una página o por tiempo máximo de sesión [30].

WUM también utiliza registros web preprocesados en el formato WUMprep los cuales se obtienen a través de la herramienta WUMprep, que es una herramienta complementaria de WUM y se basa en automatizar las tareas de preprocesamiento. La eliminación de accesos secundarios irrelevantes como imágenes, la eliminación de accesos duplicados o redundante, la detección de sesiones, la eliminación de accesos de procesos automáticos y la sustitución de las URL por conceptos, son las tareas que se pueden realizar en dicha herramienta. Los registros web obtenidos en WUMprep son muy similares a los registros web originales, se diferencian en que los primeros incluyen solamente los accesos relevantes, sustituyen la URL por conceptos que se hayan definido y cada acceso ya viene con un prefijo que es el número de la sesión a la que pertenece.

Después que se cargan los registros web y son creadas las sesiones en WUM, se pasa a la extracción de conocimiento y resultados a partir de estas sesiones como son: las representaciones gráficas de estas sesiones en grafo o árbol, la consulta de sesiones usando MINT el cual es un lenguaje propio de sintaxis parecida a la de SQL, la creación y visualización de registro web agregado y la creación de informes de resumen en formato HTML [30].

Entre sus debilidades se pueden encontrar que no se realiza la identificación de perfiles de usuarios, no hace agrupamiento de perfiles de usuarios, no tiene en cuenta el domino del trabajo de usuario, su arquitectura no tiene bien diferenciada las fases del proceso de minería web de uso y no realiza un filtrado colaborativo [55].

WebMiner es un sistema que implementa la arquitectura propuesta por Cooley la cual divide el proceso de minería web de uso en dos partes, la primera incluye el proceso del dominio de la transformación de los datos web en formatos que se ajusten a la transacciones lo cual incluye preprocesamiento, identificación de transacción e integración de componentes de datos y la segunda incluye técnicas de minería de datos y reconocimientos de patrones [56].

Este presenta modelos de datos y transacciones para varias tareas de la minería web de uso como el descubrimiento por asociación de reglas y patrones secuenciales para los datos web. Aplica técnicas de descubrimiento de conocimientos y hace un análisis de patrones secuenciales. Además de que

propone como trabajos futuros el desarrollo de agentes autónomos que analicen el descubrimiento de reglas de clasificación y un mecanismo de consultas que pueda ser manipulado en el pre descubrimiento.
Esta herramienta posee las mismas debilidades de WUM anteriormente mencionadas además de que no presenta un enfoque centralizado [56].

Weka, mencionada en el epígrafe 1.2.6, utilizando sus algoritmos de minería de datos aplicados a la minería web de uso fue la herramienta que se escogió para el desarrollo del presente trabajo, ya que tiene bien marcada todas las fases del proceso de minería de datos y contiene una gran cantidad de algoritmos implementados que ayudarán en la tarea de extraer los patrones de acceso y navegación en los sitios web. Se utilizará la versión 3.6 con la cual se analizarán los datos y mostrarán de manera gráfica la correlación entre ellos.

2. Diseño del Módulo de Procesamiento Inteligente de Datos

2.1 Arquitectura de Airesweb

Airesweb es un software de analítica web, que tiene como objetivo principal brindar la posibilidad de que tanto administradores de sitios web como directivos, obtengan datos reales sobre la actividad realizada por los usuarios sobre sus páginas, ayudándoles en la toma de decisiones, tanto comerciales como técnicas.

Existen diferentes tipos de analizadores de registros. Los primeros analizadores examinaban los registros de los servidores web, pero en los últimos años ha surgido una tendencia a analizar el tráfico web insertando un código de seguimiento en cada página que se quiera monitorizar. Este código envía la información recopilada hacia un servidor en el que se procesan los datos. Airesweb utiliza esta forma de medición y recuperación de los datos. El sistema crea sus propios ficheros de registros a partir de la información recopilada de los clientes, por el código de seguimiento. En el servidor hay un componente que recibe las peticiones y escribe la línea del registro, una línea con un formato estándar definido para el sistema.

En la figura 8 se muestra arquitectura del sistema Airesweb.

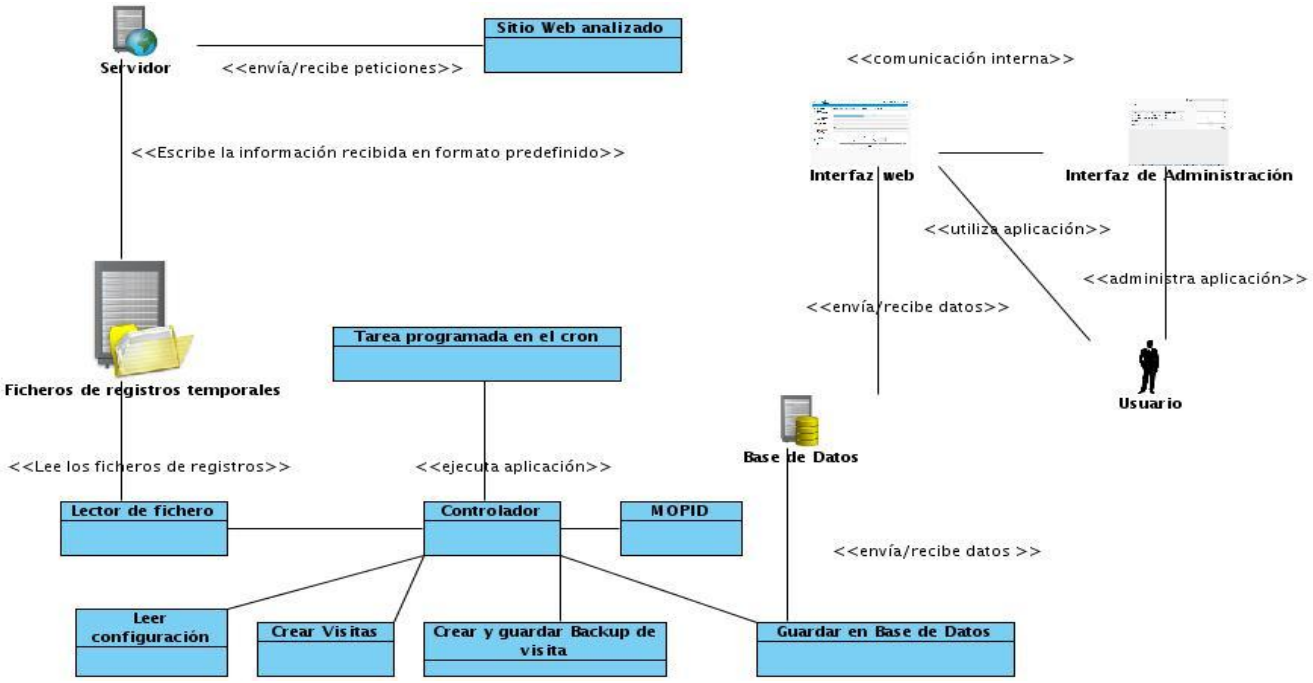

Figura 8 Arquitectura de Airesweb

El software está compuesto por tres módulos fundamentales: el módulo de instalación, que se encarga de realizar un instalador para el usuario, el módulo de procesamiento de datos o Core, que realiza el procesamiento de los archivos de registros y los guarda en una base de datos y por último el módulo de presentación, el cual lee la información y la muestra mediante una interfaz web.

Módulo de Procesamiento de Datos

El módulo de procesamiento de datos procesa la información que contienen los archivos generados por el tráfico web, esta información es analizada y guardada en una base de datos para luego ser mostrada por la interfaz web.

El Controlador Principal tiene un módulo que lee la configuración, donde están los datos necesarios para que el sistema localice los archivos de registros temporales. El lector de fichero es el encargado de leer los registros temporales, pasar los mismos al controlador, quien crea la visita, guarda las visitas que no se han completado y escribe en la base de datos la información compilada.

Módulo de Presentación

El módulo de presentación se utiliza para gestionar la información generada por el sistema. Además permite la gestión de los usuarios y de los sitios monitorizados por la aplicación. En la interfaz se muestran los datos almacenados. Esta información es presentada mediante tablas y gráficas para lograr una mejor interpretación por parte del usuario final. Además se observan un total de 31 reportes, entre los que se encuentran:

- Visitantes únicos: Muestra un indicador de la cantidad de nuevos usuarios que contiene el sitio analizado.
- Visitantes únicos distribuidos por países: Muestra la demografía de los visitantes, esta información se puede utilizar para ajustar el sitio analizado a los intereses de los visitantes que más acceden en dependencia de donde provienen.
- Visitantes únicos distribuidos por tiempo: Muestra un indicador de la hora, día o mes en el que existe más tráfico en el sitio analizado.
- Las páginas visitadas separadas por nivel: Esta información puede ser de gran ayuda para determinar si una página que está en un nivel de navegación determinado está siendo accedida de la forma que desea el administrador del sitio y a partir de esto se puede decidir si cambiar el nivel de navegación de la misma.
- Las visitas al sitio, analizando a la vez el tiempo de duración de las mismas, muestra un

indicador del tiempo en que los usuarios navegan en el sitio web analizado.

- Acceso desde buscadores al sitio y las palabras claves con las que accedieron, principales navegadores y sistemas operativos, los bytes transferidos por los visitantes, entre otras estadísticas de utilidad.

2.1.1 Integración de MOPID a Airesweb

MOPID va a estar insertado en la arquitectura de Airesweb como un módulo más, él cual será manejado por el Controlador Principal. Es decir, a MOPID lo llama el Controlador y le pasa como parámetro los ficheros de registros a analizar, MOPID devolverá las reglas de asociación y patrones secuenciales encontrados y el Controlador se encargará de escribirlos en la Base de datos, para que después puedan ser consultados desde la interfaz web.

2.2 Ficheros

Los datos con los que se van a trabajar se obtendrán de los ficheros temporales que genera el sistema Airesweb. De estos ficheros se tomarán los campos de interés para la investigación para luego aplicar las técnicas de minería de datos.

El primer paso para empezar a trabajar en Weka es definir el origen de los datos, ya que esta soporta diferentes fuentes. Entre estas fuentes se encuentra la extracción de información a partir de una base de datos. Para conectarse a ella hay que especificar la dirección de la misma, así como el usuario y la contraseña. Otra vía es mediante una URL donde se especifica la dirección donde se encuentra el fichero a procesar. Y por último a partir de un fichero de texto donde se debe especificar la ruta donde se encuentra y seleccionarlo. El formato por defecto de los ficheros que Weka utiliza es el ARFF, pero también soporta otros tipos como son el CSV y C4.5. En la presente investigación se trabajará con los ficheros de texto, cuyos datos deben estar dispuestos como se especifica en el epígrafe 2.2.2.

2.2.1 Fichero de registros temporales

Los ficheros de registros temporales generados por Airesweb están compuestos por distintos campos entre los que se encuentran: el código que representa la identificación del usuario, la fecha en la que el servidor recibió la petición, la dirección IP, la URL, la URL de referencia (urlref), la resolución, la información que identifica al navegador conocido como “User-Agent”, entre otros como se muestra en la tabla siguiente:

115f96f5647b2fba9022120ed0ea91fa [15/Mar/2010:16:31:15 -0400] - 10.33.13.240 http://10.33.13.13:8080/index.php/default http://10.33.13.13:8080/index.php 1024x768 0 0 0 0 0 1 0 0 0 Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2) Gecko/20100115 Firefox/3.6

Tabla 2.1 Ejemplo de fichero de registro temporal generado por Airesweb.

Estos ficheros de registros temporales serán transformados a ficheros ARFF en MOPID, teniendo en cuenta solamente los siguientes campos: sesión, fecha, dirección IP, URL, URL de referencia, así como el sistema operativo y el navegador encontrados en el User-Agent. La identificación de la sesiones se realizará como fue explicado en el capítulo anterior, para lo cual se utilizará el código de identificación del usuario y la fecha.

2.2.2 Fichero ARFF

El formato ARFF está estructurado claramente en 3 partes: cabecera, declaración de atributos y sección de los datos.

En la cabecera se define el nombre de la relación, de la forma: @relation <nombre-de -relación>, expresado como una cadena de texto. Si dicho nombre, contiene algún espacio será necesario indicarlo entrecomillado.

En la sección de declaración de los atributos, se declararan todos los atributos que contendrá el archivo junto al tipo de dato por ejemplo: @attribute <nombre-de-atributo> <tipo>. Weka acepta diversos tipos de atributos como son:

- STRING: Expresa cadenas de texto.
- NOMINAL: Expresa entre llaves separados por coma los posibles valores que puede tomar el atributo.
- NUMERIC: Expresa números reales.
- INTEGER: Expresa números enteros.
- DATE: Expresa fechas para ello debe ir precedido de una etiqueta de formato entrecomillada la cual está compuesta por caracteres separadores y unidades de tiempo: dd Día, MM Mes, yyyy Año, HH Horas, mm Minutos, ss Segundos.

Por último en la sección de los datos se declaran los que componen la relación, separando entre comas los atributos y con saltos de líneas las relaciones. En caso de que algún dato sea desconocido se expresa con el signo de cerrar interrogación “?” , y con el símbolo de “%” se indicará que desde ese símbolo hasta el fin de línea, corresponde a un comentario.

En la figura 9 se muestra un ejemplo completo de un fichero ARFF:

Ejemplo.arff

```
@relation pagians_vistas

@attribute sesion string
@attribute fecha date "dd-MM-yyyy HH:mm:ss"
@attribute ip string
@attribute url string
@attribute urlref string
@attribute so string
@attribute browser string

@data
1,"10-03-2010 22:12:46",10.8.121.102,http://feu10.uci.cu/,?,Windows,Firefox/3.6
1,"10-03-2010 22:18:32",10.33.13.70,http://feu10.uci.cu/,?,Linux,Firefox/3.5.8
2,"16-03-2010 10:37:37",10.33.13.10,http://10.33.13.10:8080/index.php/default,http://10.33.13.10:8080/index.php/,Linux,Firefox/3.5.8
2,"16-03-2010 10:39:35",10.8.13.10,http://10.33.13.10:8080/index.php/default,http://10.33.13.10:8080/index.php/,Linux,Firefox/3.5.8
2,"16-03-2010 10:48:58",10.33.13.13,http://10.33.13.10:8080/index.php/default,http://10.33.13.10:8080/index.php/,Linux,Firefox/3.5.8
```

Figura 9 Ejemplo de un fichero en el formato ARFF.

2.3 Algoritmos de búsqueda de reglas de asociación

El descubrimiento de reglas de asociación busca relaciones o asociaciones entre conjuntos de ítems, donde un ítem es un par atributo-valor. Una regla de asociación está compuesta por dos conjuntos de ítems llamados premisa y conclusión, los cuales se unen mediante una flecha que va desde la premisa hacia la conclusión por ejemplo A→B, siendo A y B conjuntos de ítems [57]. La conclusión siempre contiene un solo par atributo-valor.

Soporte y confianza de una regla de asociación [58]

El soporte de la regla se define como la proporción de transacciones que contienen tanto la premisa como la conclusión. Es decir, el número de veces que se encontró la regla de asociación entre la cantidad de relaciones de datos analizados.
Soporte (A→B) = soporte (A U B)

Por otra parte la confianza es la probabilidad condicional de que un registro que contenga A también contenga B.
Confianza (A→B) = P(A/B)= soporte (A ∩ B) / soporte(A)

Los algoritmos de asociación permiten la búsqueda automática de reglas que relacionan conjuntos entre sí. Son algoritmos no supervisados ya que no existen relaciones conocidas a priori con las que comprobar la validez de los resultados, sino que se evalúa si esas reglas son estadísticamente significativas mediante los valores del soporte y la confianza.

La búsqueda exhaustiva de reglas consideraría todas las posibles combinaciones de atributos ubicándolos como premisas y conclusiones, luego se evaluaría el soporte y la confianza de cada regla y se descartarían todas las asociaciones que no cumplen con los valores mínimos establecidos de estos [58]. Lo descrito anteriormente resulta un proceso computacionalmente muy costoso ya que el número de reglas de asociación encontradas aumenta rápidamente. Debido a esto los algoritmos existentes aplican técnicas heurísticas para reducir en gran medida el posible número de conjuntos que se tendrían en cuenta, de acuerdo a la estimación de si podrán o no ser frecuentes.

2.3.1 Algoritmo Apriori

El algoritmo Apriori soluciona el problema reduciendo el número de conjuntos considerados ya que genera solamente los conjuntos que cumplen con la condición de tener un soporte menor o igual al soporte mínimo definido por el usuario previamente.

El algoritmo emplea un proceso iterativo de acceso donde los k-itemsets son usados para buscar los (k+1)-itemsets, es decir primeramente se busca el conjunto de ítems frecuentes de longitud uno L1 y a partir de este se busca el conjunto de longitud dos L2, luego el conjunto dos es usado para buscar L3 y así sucesivamente [59]. El descubrimiento de cada conjunto requiere de una búsqueda completa en la base de datos por lo que para mejorar la eficiencia de la generación de conjuntos frecuentes el algoritmo se basa en una propiedad que reduce el espacio de búsqueda.

La propiedad plantea que todo subconjunto no vacío de un conjunto de ítems frecuentes tiene que ser frecuente también. Si un conjunto de ítems X no satisface el soporte mínimo entonces no es frecuente y si a este conjunto X se le adiciona un ítem entonces el conjunto resultante ocurrida con menor o igual frecuencia que X [60].

Apriori es el principal algoritmo de asociación implementado en Weka, el cual solo puede buscar reglas a partir de atributos simbólicos, es decir atributos nominales mencionados anteriormente, por esta razón es necesario convertir todos los atributos de tipo numérico o cadena a atributos nominales mediante la aplicación de diferentes filtros.

Como los valores de los campos de los registros web relevantes para esta investigación son en su gran mayoría de tipo string o date es necesario aplicarle los filtros de preprocesamiento de Weka conocidos como StringToNominal y Discretize para convertirlos a nominales antes de utilizar el algoritmo.

El filtro StringToNominal convierte los atributos de tipo string a nominal, tiene un solo parámetro en el que se especifica el rango de los atributos que serán procesados. La fecha en Weka es convertida a Tiempo Unix interpretado como un atributo numérico por lo que se emplea el filtro Discretize encargado de transformar atributos numéricos en simbólicos, como parámetros toma los índices de los atributos a discretizar (attributeIndices) y el número de particiones en que se quiera dividir los datos (bins), además se puede seleccionar si las particiones se realizan por la frecuencia de los datos y no por el tamaño de estas mediante el parámetro useEqualFrequency.

En este algoritmo no se necesita analizar las sesiones para la búsqueda de reglas de asociación por lo que se aplica el filtro Remove para eliminar el atributo sesión. Este filtro tiene como parámetro attributeIndices mediante el cual se especifica el rango de atributos a eliminar.

El algoritmo Apriori para mejorar su eficiencia en la búsqueda de conjuntos de ítems frecuentes elimina los atributos que tengan sus valores desconocidos en todos los resultados pero esto en la presente investigación no es conveniente ya que todos los atributos utilizados son relevantes y es necesario que se tengan en cuenta. Para solucionar este problema se aplica el filtro ReplaceMissingValues que reemplaza todos los atributos desconocidos por la moda en caso de que sea un atributo nominal o la media aritmética si es numérico.

Una vez que los datos están listos ya se puede aplicar el algoritmo con sus parámetros por defecto. En la figura 10 se aprecia como resultado un conjunto de 10 reglas de asociación obtenidas a partir del procesamiento de un conjunto de registros web generados por Airesweb.

```
Apriori
=======

Minimum support: 0.25 (108 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 15

Best rules found:

 1. fecha='(1268737757200-inf)' 175 ==> so=Linux 175    conf:(1)
 2. ip=10.33.13.240 110 ==> so=Linux 110    conf:(1)
 3. ip=10.33.13.240 110 ==> browser=Firefox/3.6 110    conf:(1)
 4. so=Linux browser=Firefox/3.6 110 ==> ip=10.33.13.240 110    conf:(1)
 5. ip=10.33.13.240 browser=Firefox/3.6 110 ==> so=Linux 110    conf:(1)
 6. ip=10.33.13.240 so=Linux 110 ==> browser=Firefox/3.6 110    conf:(1)
 7. ip=10.33.13.240 110 ==> so=Linux browser=Firefox/3.6 110    conf:(1)
 8. browser=Firefox/3.6 120 ==> ip=10.33.13.240 110    conf:(0.92)
 9. browser=Firefox/3.6 120 ==> so=Linux 110    conf:(0.92)
10. browser=Firefox/3.6 120 ==> ip=10.33.13.240 so=Linux 110    conf:(0.92)
```

Figura 10 Resultados del algoritmo Apriori de minería de datos después de ejecutarlo con el archivo generado mediante los datos originales.

Analizando estos resultados se puede apreciar de manera general que reglas como el 100% de los usuarios que acceden al sitio desde la dirección IP 10.33.13.240 utilizan como sistema operativo GNU/Linux, no aportan información significativa acerca del comportamiento de los usuarios en la web. Por tanto es necesario proveer los parámetros adecuados para obtener una buena cantidad de reglas de asociación confiables que puedan brindar información interesante sobre el tráfico web. Estos parámetros se configuran teniendo en cuenta el tipo y la cantidad de datos que se esté manejando. Los parámetros son:

- Car: Brinda la posibilidad de seleccionar si se quiere hacer las reglas de asociación de forma general tomando todos los atributos como clase (conclusión) o tomando un solo atributo como clase.
- ClassIndex: En caso de seleccionar un solo atributo como clase en el parámetro anterior en este se especifica que atributo se desea tomar. Por defecto trae -1 que significa que se va a tomar como clase el último atributo.
- Soporte Mínimo: Se establece el soporte mínimo que la regla debe cumplir. Un soporte mínimo muy alto no arrojaría ningún resultado.

- Delta: Iterativamente se decrece el valor del soporte máximo hasta llegar al soporte mínimo por este parámetro.
- Tipo de métrica: Esta opción brinda la posibilidad de seleccionar el tipo de métrica que se quiere utilizar, en este caso se indica la confianza para poder determinar el porcentaje de la calidad de las reglas.
- Mínimo de la métrica: Mediante este parámetro se introduce la confianza mínima que las reglas a obtener tengan que cumplir. Este valor debe ser pequeño para poder obtener reglas que no se descubran a simple vista y así obtener información interesante.
- Número de reglas: Indica el número máximo de reglas a obtener, es decir se utiliza como criterio de parada para detener la ejecución si se llega a este número de reglas cumpliendo las restricciones anteriores.
- Soporte máximo: Se establece un soporte máximo a partir del cual se va a empezar a decrecer hasta llegar al soporte mínimo establecido.

A continuación se analizarán los resultados de la ejecución del algoritmo con diferentes parámetros.

Primeramente si se cambia el parámetro car, el cual por defecto tiene el valor false, por true y el valor de classIndex, se obtendrá en la reglas resultantes como conclusión los atributos más significativos como son el atributo url y el atributo urlref.

Al contar con una gran cantidad de datos podrían no encontrarse reglas de asociación que incluyan los atributos url y urlref ya que los valores en dichos campos no tiene una gran ocurrencia, por lo que se deberá disminuir el soporte mínimo para que se encuentren más reglas de asociación. Además la confianza inicialmente está muy elevada, provocando que solo se muestren las reglas que tengan una confianza de 0.9, pudiéndose encontrar otras reglas de interés con una confianza menor.

Y para obtener un mayor número de reglas se cambiará el atributo número de reglas. Con los valores de los parámetros siguientes:

Car: True
ClassIndex: 3 (Correspondiente al atributo url)
Soporte Mínimo: 0.05
Tipo de métrica: Confianza
Confianza Mínima: 0.5
Número de Reglas: 200

Se obtuvieron los siguientes resultados:

```
Apriori
=======

Minimum support: 0.05 (12 instances)
Minimum metric <confidence>: 0.5
Number of cycles performed: 19

Best rules found:

 1. urlref=http://10.33.13.10:8080/index.php/ 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
 2. ip=10.33.13.10 urlref=http://10.33.13.10:8080/index.php/ 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
 3. urlref=http://10.33.13.10:8080/index.php/ so=Linux 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
 4. urlref=http://10.33.13.10:8080/index.php/ browser=Firefox/3.5.8 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
 5. ip=10.33.13.10 urlref=http://10.33.13.10:8080/index.php/ so=Linux 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
 6. ip=10.33.13.10 urlref=http://10.33.13.10:8080/index.php/ browser=Firefox/3.5.8 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
 7. urlref=http://10.33.13.10:8080/index.php/ so=Linux browser=Firefox/3.5.8 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
 8. ip=10.33.13.10 urlref=http://10.33.13.10:8080/index.php/ so=Linux browser=Firefox/3.5.8 13 ==> url=http://10.33.13.10:8080/index.php/default 13    conf:(1)
```

Figura 11 Resultados del algoritmo Apriori después de ejecutarlo con los parámetros modificados.

Se encontraron un total de 8 reglas de asociación, debido a que el soporte mínimo seleccionado tiene un valor muy elevado en comparación con la cantidad de datos distintos que existen. Una vez obtenidos los resultados es necesario interpretarlos y para esto hay que tener en cuenta el nivel de confianza obtenido con cada regla. De manera general se evidencia que el 100% de las visitas efectuadas a la página *http://10.33.13.10:8080/index.php/default* se realizaron desde la página *http://10.33.13.10:8080/index.php*. Modificando los valores de los parámetros siguientes:

ClassIndex: 4 (Correspondiente al atributo urlref)
Soporte Mínimo: 0.02
Confianza Mínima: 0.4
Número de Reglas: 200

Se obtuvieron 72 reglas de asociación entre las que se encuentran:

```
Apriori
=======

Minimum support: 0.02 (5 instances)
Minimum metric <confidence>: 0.4
Number of cycles performed: 20

Best rules found:

 1. url=http://10.33.13.10:8080/index.php/report?id_site=4&name_site=AiresWeb+Offficial+Site 8 ==> urlref=http://10.33.13.10:8080/index.php/default 8    conf:(1)
 2. url=http://10.33.13.10:8080/index.php/report?id_site=4&name_site=AiresWeb+Offficial+Site so=Linux 8 ==> urlref=http://10.33.13.10:8080/index.php/default 8    conf:(1)
 3. url=http://10.33.13.13:8080/index.php/report?id_site=4&name_site=AiresWeb+Offficial+Site 6 ==> urlref=http://10.33.13.13:8080/index.php/default 6    conf:(1)
 4. url=http://10.33.13.10:8080/index.php/report?id_site=31&name_site=hayhsdjasd 6 ==> urlref=http://10.33.13.10:8080/index.php/default 6    conf:(1)
 5. url=http://10.33.13.13:8080/index.php/default/ayuda_2 6 ==> urlref=http://10.33.13.13:8080/index.php/default 6    conf:(1)
 6. fecha='(1268744660000-1268804245000]' url=http://10.33.13.13:8080/index.php/default/ayuda_2 6 ==> urlref=http://10.33.13.13:8080/index.php/default 6    conf:(1)
 7. ip=10.33.13.13 url=http://10.33.13.13:8080/index.php/default/ayuda_2 6 ==> urlref=http://10.33.13.13:8080/index.php/default 6    conf:(1)
53. ip=10.33.13.10 url=http://10.33.13.10:8080/index.php/default 19 ==> urlref=http://10.33.13.10:8080/index.php/ 13    conf:(0.68)
54. url=http://10.33.13.10:8080/index.php/default browser=Firefox/3.5.8 19 ==> urlref=http://10.33.13.10:8080/index.php/ 13    conf:(0.68)
55. fecha='(-inf-1268834037500]' ip=10.33.13.10 url=http://10.33.13.10:8080/index.php/default 19 ==> urlref=http://10.33.13.10:8080/index.php/ 13    conf:(0.68)
56. fecha='(-inf-1268834037500]' url=http://10.33.13.10:8080/index.php/default browser=Firefox/3.5.8 19 ==> urlref=http://10.33.13.10:8080/index.php/ 13    conf:(0.68)
57. ip=10.33.13.10 url=http://10.33.13.10:8080/index.php/default so=Linux 19 ==> urlref=http://10.33.13.10:8080/index.php/ 13    conf:(0.68)
58. ip=10.33.13.10 url=http://10.33.13.10:8080/index.php/default browser=Firefox/3.5.8 19 ==> urlref=http://10.33.13.10:8080/index.php/ 13    conf:(0.68)
```

Figura 12 Resultados del algoritmo Apriori después de ejecutarlo con los parámetros modificados.

Tras el análisis de los resultados se puede observar que:

- El 68% de las visitas realizadas a la página *http://10.33.13.10:8080/index.php/default*, desde la dirección IP 10.33.13.13, utilizando como sistema operativo Linux se efectuaron a través de la página *http://10.33.13.10:8080/index.php*.
- El 100% de las visitas realizadas a la página *http://10.33.13.13:8080/index.php/default/ayuda_2* desde la dirección IP 10.33.13.13 se efectuaron a través de la página *http://10.33.13.13:8080/index.php/default*.

Como se ha podido observar después de aplicar el algoritmo Apriori con distintos valores de sus parámetros, este arroja diversos resultados de utilidad para Airesweb, por lo que debe ser aplicado varias veces modificado los valores de los parámetros.

Otro tipo de algoritmo Apriori es el algoritmo Predictive Apriori, que es muy similar al algoritmo Apriori original, pero su diferencia radica en que el Predictive Apriori combina el soporte y la confianza en una única medida y el soporte mínimo se aumenta hasta alcanzar el soporte máximo [61]. Al combinar el

soporte mínimo y la confianza en una sola medida se requiere un mayor tiempo y recursos para el procesamiento de los datos.

Tertius también es un tipo de algoritmo Apriori, el cual busca las reglas de asociación de acuerdo con una medida de confirmación, pero este utiliza condiciones OR en lugar de AND en la premisa de la regla, es decir que se tienen variados elementos en la premisa y por tanto el análisis tendría una menor precisión [62].

Como al utilizar el algoritmo Predictive Apriori se obtuvieron reglas de asociación muy similares a las encontradas por el Apriori empleando más tiempo el primero en encontrar los resultados y además al Tertius tener una menor precisión solo se utilizará el Apriori para la búsqueda de reglas de asociación.

2.4 Algoritmo de búsqueda de patrones secuenciales

Las reglas de asociación no tienen en cuenta el orden de las transacciones, sin embargo en muchas aplicaciones el orden de las mismas es de gran utilidad. Por ejemplo, en la minería web de uso es útil para encontrar patrones de navegación de un sitio web a partir de secuencias de visitas de los usuarios a las páginas web. Para estas aplicaciones, las reglas de asociación no son apropiadas, por lo que es necesario emplear los patrones secuenciales.

Sea ***I= {i1, i2,…,in}*** un conjunto de elementos, una secuencia sería una lista ordenada de los conjuntos de elementos. Se puede denotar una secuencia ***s*** como **< *a1, a2,…,am* >** donde ***ai*** es un conjunto de elementos también conocido como elemento de ***s***. Un elemento de la secuencia se puede expresar como ***{x1, x2,…,xr }*** donde ***xj*** $\in$ ***I*** y es un elemento [42].

La búsqueda de patrones secuenciales se basa en encontrar todas las secuencias que tienen un soporte mayor que el soporte mínimo especificado por el usuario. Al igual que en las reglas de asociación, cada secuencia de este tipo se llama secuencia frecuente o patrón secuencial. Dado un conjunto S de secuencias de datos de entrada el soporte de una secuencia frecuente es la fracción de las secuencias de datos S que contienen dicha secuencia.

Por ejemplo, utilizando el análisis de la cesta de mercado, cada secuencia en este contexto representa una lista ordenada de las operaciones de un cliente en particular. Una transacción es un conjunto de elementos que el cliente compra a la vez, llamado tiempo de transacción. La tabla 2.2 muestra los valores de una base de datos sobre las operaciones de compra realizadas, las cuales están ordenadas por el identificador del cliente. En la tabla 2.3 se observa la secuencia de los datos también llamado

secuencia del cliente y por último en la tabla 2.4 se muestra los patrones secuenciales resultantes con un soporte mínimo del 25 %.

Id del Cliente	Tiempo de Transacción	Elementos Comprados
1	Julio 20, 2009	30
1	Julio 25, 2009	90
2	Julio 9, 2009	10, 20
2	Julio 14, 2009	30
2	Julio 20, 2009	10, 40, 60, 70
3	Julio 25, 2009	30, 50, 70, 80
4	Julio 25, 2009	30
4	Julio 29, 2009	30, 40, 70, 80
4	Agosto 2, 2009	90
5	Julio 12, 2009	90

Tabla 2.2 Conjunto de transacciones

Id del Cliente	Secuencia de datos
1	<{30} {90}>
2	<{10, 20} {30} {10, 40, 60, 70}>
3	<{30, 50, 70, 80}>
4	<{30} {30, 40, 70, 80} {90}>
5	<{90}>

Tabla 2.3 Secuencias de datos producidas por las transacciones de la tabla 1.1

	Patrones secuenciales con soporte mínimo ≥ 25%
1-secuencia	<{30}>, <{40}>, <{70}> , <{80}>, <{90}>
2-secuencia	<{30} {40}>, <{30} {70}>, <{30}, {90}>, <{30, 70}>, <{30, 80}>, <{40, 70}>, <{70, 80}>
3-secuencia	<{30} {40, 70}>, <{30, 70, 80}>

Tabla 2.4 Patrones Secuenciales

Para calcular el soporte de la secuencia < {30}> se divide la cantidad de veces que aparece en la secuencia de datos entre el número total de secuencia de datos, en este caso sería 4/5=0.8 lo que significa el 80%.

2.4.1 Algoritmo GeneralizedSequentialPatterns

GeneralizedSequentialPatterns (GSP) es un algoritmo que descubre patrones secuenciales generalizados. El algoritmo realiza múltiples búsquedas sobre los datos. Primeramente determina el soporte de cada uno de los elementos, es decir las veces que se repite el elemento entre la cantidad

de transacciones. Al final de la primera búsqueda el algoritmo conoce que elementos son los más frecuentes utilizando el soporte mínimo, los cuales arroja como resultado de la primera iteración. Estos resultados son la semilla para establecer nuevas secuencias frecuentes, es decir cada iteración comienza con una semilla, para generar una nueva secuencia potencialmente frecuente conocida como secuencia candidata. Cada secuencia candidata tiene un elemento más que la secuencia semilla, por lo cada secuencia candidata en cada iteración tiene el mismo número de elementos. El soporte de cada secuencia candidata se encuentra durante las búsquedas sobre los datos, para determinar cuales en realidad son las más frecuente cumpliendo con el soporte mínimo. El algoritmo termina cuando no existen secuencias frecuentes que cumplan con el soporte mínimo o no se encuentren nuevas secuencias candidatas [63].

Al igual que el algoritmo A priori, el algoritmo GSP implementado en Weka solo puede buscar patrones secuenciales a partir de atributos simbólicos, por tanto es necesario realizar el mismo procedimiento de filtrado en los datos originales que se realizó en el algoritmo Apriori con los filtros StringToNominal y Discretize. También se emplea el filtro Remove para eliminar algunos atributos que no son necesarios para la búsqueda de patrones secuenciales como son: la fecha, el sistema operativo y el navegador.

Una vez que los datos estén correctamente filtrados, se puede aplicar el algoritmo GSP pero es necesario definirle los parámetros de manera tal, que aporten información significativa para la investigación. Los parámetros son:

dataSeqID: Número del atributo que representa el identificador de la secuencia.
filterAttributes: Número de atributo utilizado para comenzar la búsqueda de secuencias frecuentes. Si se toma -1 utilizará todos los atributos como salida de la primera iteración.
minSupport: Soporte mínimo que debe cumplir la secuencia.

El algoritmo como valores por defecto de los parámetros, trae un soporte mínimo de 0.9 y un filterAttributes de -1, lo cual no proporciona ningún resultado significativo ya que el soporte mínimo es muy elevado. Por tanto es necesario cambiarle los valores de los parámetros de la siguiente manera:

dataSeqID: 0
filterAttributes: 3
minSupport: 0.5

En la figura 9 se puede observar las secuencias obtenidas con los parámetros definidos anteriormente.

```
GeneralizedSequentialPatterns
=============================

Number of cycles performed: 8
Total number of frequent sequences: 257

Frequent Sequences Details (filtered):

- 1-sequences

[1] <{http://10.33.13.13:8080/index.php/default}> (2)
[2] <{http://10.33.13.10:8080/index.php/default}> (2)
[3] <{http://10.33.13.10:8080/index.php/report}> (2)

- 2-sequences

[1] <{10.33.13.240,http://10.33.13.13:8080/index.php/default}> (2)
[2] <{10.33.13.240,http://10.33.13.10:8080/index.php/report}> (2)
[3] <{http://10.33.13.13:8080/index.php/default,http://google.com}> (2)
[4] <{http://10.33.13.10:8080/index.php/report,http://yahoo.es}> (2)

- 3-sequences

[1] <{10.33.13.240,http://10.33.13.13:8080/index.php/default,http://google.com}> (2)
[2] <{10.33.13.240,http://10.33.13.10:8080/index.php/report,http://yahoo.es}> (2)
```

Figura 13 Resultados del algoritmo GSP después de ejecutarlo con los parámetros modificados.

Después de analizar los resultados se puede observar que:

- El 40% de los usuarios visitaron la página *http://10.33.13.13:8080/index.php/default* y la página *http://google.com* desde la dirección IP 10.33.13.240.
- El 40% de los usuarios visitaron la página *http://10.33.13.10:8080/index.php/report* y la página *http://yahoo.es* desde la dirección IP 10.33.13.240.

2.5 Algoritmo de análisis de caminos HotSpot

El algoritmo HotSpot encuentra un conjunto de reglas que se muestran mediante una estructura de árbol, e cual representan relaciones entre las páginas web. Al igual que los algoritmos Apriori y GSP, los atributos deben ser de tipo nominal, por lo que se hace necesario realizar el preprocesado de los atributos anteriormente mencionado, mediante el filtro StringToNominal y Discretize. Además se utiliza el filtro Remove ya que solamente son de interés los atributos correspondientes a las sesiones, las url y las url de referencia. Este algoritmo cuenta con una serie de parámetros:

- maxBranchingFactor: Número máximo de nodos a considerar en cada ampliación del árbol.
- minImprovement: Mejora mínima en el valor objetivo a fin de considerar la adición de una nueva rama.
- Soporte: Representa el soporte mínimo que deben cumplir las reglas. El valor del soporte es un número entre 0 y 1, que se interpreta como el porcentaje de la cantidad total de datos.
- Target: Indica el atributo de interés, es decir el atributo que será la raíz del árbol.
- targetIndex: Indica el valor del atributo de interés.

Para los siguientes valores de los atributos se obtienen los resultados que se muestran en la figura # 10:

maxBranchingFactor: 20

minImprovement: 0.1

Soporte: 0.001

Target: first

targetIndex: first

```
Hot Spot
========
Total population: 887 instances
Target attribute: sesiones
Target value: 1 [value count in total population: 17 instances (1.92%)]
Minimum value count for segments: 1 instances (0.1% of total population)
Maximum branching factor: 20
Minimum improvement in target: 1%

sesiones=1 (1.92% [17/887])
  urlref = http://airesweb.f10.uci.cu/index.php/report/demografia?l=1 (100% [3/3])
  url = http://airesweb.f10.uci.cu/index.php/site/edit/id/5 (100% [2/2])
  url = http://airesweb.f10.uci.cu/index.php/report?id_site=5&name_site=GIDI (100% [2/2])
  urlref = http://airesweb.f10.uci.cu/index.php/site/edit/id/5 (100% [2/2])
  urlref = http://airesweb.f10.uci.cu/index.php/report?id_site=5&name_site=GIDI (100% [2/2])
  urlref = http://airesweb.f10.uci.cu/index.php/site/new (100% [1/1])
  urlref = http://airesweb.f10.uci.cu/index.php/report?select_param=actual (100% [1/1])
  urlref = http://airesweb.f10.uci.cu/index.php/default/ayuda_1 (100% [1/1])
  url = http://airesweb.f10.uci.cu/index.php/default/ayuda_1 (100% [1/1])
  url = http://airesweb.f10.uci.cu/index.php/report/demografia?l=1 (100% [1/1])
  url = http://airesweb.f10.uci.cu/index.php/site/new (100% [1/1])
  url = http://airesweb.f10.uci.cu/index.php/report?select_param=actual (100% [1/1])
  urlref = http://airesweb.f10.uci.cu/index.php/default/ayuda_2 (100% [1/1])
  url = http://airesweb.f10.uci.cu/index.php/default/ayuda_2 (100% [1/1])
  urlref = http://airesweb.f10.uci.cu/index.php/default (83.33% [5/6])
  |   url = http://airesweb.f10.uci.cu/index.php/report?id_site=5&name_site=GIDI (100% [2/2])
  |   url = http://airesweb.f10.uci.cu/index.php/default/ayuda_1 (100% [1/1])
  |   url = http://airesweb.f10.uci.cu/index.php/site/new (100% [1/1])
  url = http://airesweb.f10.uci.cu/index.php/default (66.67% [8/12])
  |   urlref = http://airesweb.f10.uci.cu/index.php/report/demografia?l=1 (100% [3/3])
  |   urlref = http://airesweb.f10.uci.cu/index.php/site/edit/id/5 (100% [1/1])
  |   urlref = http://airesweb.f10.uci.cu/index.php/report?id_site=5&name_site=GIDI (100% [1/1])
  |   urlref = http://airesweb.f10.uci.cu/index.php/default/ayuda_2 (100% [1/1])
  urlref = http://airesweb.f10.uci.cu/index.php/register (50% [1/2])
```

Figura 14 Resultados del algoritmo HotSpot con los parámetros modificados para la sesión 1.

Como se puede observar en la figura 14, el algoritmo no analiza en conjunto todos los datos de todas las sesiones, lo hace de forma separada, tomando el valor especificado para buscar el árbol de relaciones. Por ejemplo, para esta sesión se puede apreciar que el usuario accedió a la página *http://airesweb.f10.uci.cu/index.php/report?id_site=5&name_site=GIDI*, a partir de la página *http://airesweb.f10.uci.cu/index.php/report/demografia?l=1* siguiendo por la página *http://airesweb.f10.uci.cu/index.php/default*.

Realizando nuevamente el algoritmo con los mismos parámetros, con excepción del valor del atributo targetIndex, que en este caso sería 2, el cual representa la sesión de usuario número 2, se obtuvieron los siguientes resultados (Figura 15):

```
Hot Spot
========
Total population: 887 instances
Target attribute: sesiones
Target value: 2 [value count in total population: 3 instances (0.34%)]
Minimum value count for segments: 1 instances (0.1% of total population)
Maximum branching factor: 20
Minimum improvement in target: 1%

sesiones=2 (0.34% [3/887])
  urlref = http://airesweb.f10.uci.cu/index.php/register (50% [1/2])
  url = http://airesweb.f10.uci.cu/index.php/default (25% [3/12])
  |   urlref = http://airesweb.f10.uci.cu/index.php/register (50% [1/2])
  |   urlref = http://airesweb.f10.uci.cu/index.php/default (50% [1/2])
  urlref = http://airesweb.f10.uci.cu/index.php/default (16.67% [1/6])
```

Figura 15 Resultados del algoritmo HotSpot con los parámetros modificados para la sesión 2.

Después de analizar los resultados se evidencia un comportamiento muy diferente para cada sesión de usuario, lo cual imposibilita obtener patrones generales de comportamiento de los usuarios. Por tanto no se utilizará el algoritmo para la implementación del prototipo funcional debido a que no aporta información de interés a la investigación.

2.6 Implementación del prototipo funcional

De los algoritmos implementados en Weka analizados anteriormente, se seleccionaron para la implementación del prototipo funcional el Apriori y el GSP. Para esto, se tuvo en cuenta que con dichos algoritmos, se obtienen patrones no triviales referentes al comportamiento de los usuarios necesitados por el sistema Airesweb para realizar un procesamiento inteligente de la información.

El prototipo funcional se implementó en el lenguaje de programación Java utilizando el API que brinda la herramienta Weka y empleando el entorno de desarrollo integrado Netbeans 6.5. El API está compuesto por dos librerías conocidas como weka-src.jar, que contiene el código fuente y weka.jar, que contiene las clases compiladas, por lo que la primera se emplea para cuando se quiere modificar el código original de las clases implementadas en Weka y la segunda para utilizar estas clases sin hacerle ninguna modificación. En el prototipo funcional solamente se utilizó la librería weka.jar, ya que en el mismo solo se van a utilizar los filtros y algoritmos implementados en la herramienta sin modificación, para demostrar que los resultados obtenidos son de gran utilidad en esta investigación.

La construcción del prototipo funcional se basa en la aplicación de la minería web para obtener patrones interesantes de acceso web. Esencialmente, el proceso consiste en transformar los ficheros de registros web en ficheros ARFF, dejando solo la información de interés como anteriormente se explicaba. Luego se le realiza un preprocesamiento a los datos contenidos en los ficheros mediante la aplicación de varios filtros. Por último, se ejecutan los algoritmos seleccionados y el resultado obtenido se analiza para mostrarle al usuario información relevante sobre el comportamiento de los usuarios al acceder a su sitio web. Este proceso se muestra en la siguiente figura:

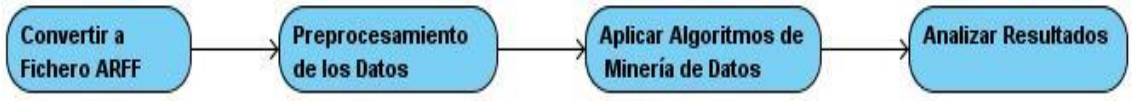

Figura 16 Proceso de Extracción de Patrones

Primeramente se lee el fichero de registro web y se convierte a fichero ARFF utilizando el método Convertir_Fichero, al cual se le pasa la dirección donde se encuentra el mismo. Este método transforma la información que se encuentra en el fichero original para que sea aceptada en Weka como formato ARFF, incorporándole inicialmente las dos primeras partes, la cabecera y la declaración de atributos, para luego con los datos de los campos relevantes, conformar la parte de la sección de los datos. Después se guarda la información en un fichero con el mismo nombre que el inicial pero con la extensión ".arff".

Cuando el fichero esta correctamente creado se carga como un objeto de la clase Instances implementada en la librería weka.jar, para así poder realizar el preprocesamiento de los datos contenidos en dicho fichero. Este preprocesamiento consiste en la aplicación de varios filtros según el algoritmo a emplear. En general se utilizan cuatro clases de la librería: StringToNominal, Discretize, Remove, ReplaceMissingValues, a las que es necesario especificarle las opciones o parámetros que le

corresponden a cada una, así como el objeto que contiene los datos a los cuales se le van a aplicar estos filtros.

Para la aplicación de los algoritmos se usan las clases Apriori y GeneralizedSequentialPattern. A cada una de estas clases se le declaran una serie de opciones en el formato “parámetro”, “valor”.

Opciones de la clase Apriori:

- -N (Número de Reglas)
- -T (Tipo de Métrica)
- -C (Confianza Mínima)
- -D (Delta)
- -U (Soporte Máximo)
- -M (Soporte Mínimo)
- -A (Car)
- -c (ClassIndex)

Opciones de la clase GeneralizedSequentialPattern:

- -S (Soporte Mínimo)
- -I (DataSeqID)
- -F (filterAttributes)

Por último se analizan los resultados obtenidos después de ejecutar los algoritmos. En este paso se utilizan dos métodos llamados AnalizarResultadosApriori y AnalizarResultadosGSP, los cuales se encargan de mostrar al usuario los patrones obtenidos de manera que sean entendibles para estos.

2.7 Propuesta de MOPID

La herramienta Weka al trabajar con grandes cantidades de datos, provoca que los algoritmos se tarden en encontrar resultados, haciendo este procedimiento lento e ineficiente. Para optimizar el tiempo que tardan estos algoritmos en encontrar los resultados, es necesario implementar los mismos en MOPID en vez de utilizar el API de Weka. Se propone para ganar en rapidez realizar el módulo en el lenguaje de programación C/C++, debido a que generalmente es un lenguaje rápido en cuanto a ejecución. Además se lograría una mejor integración con el sistema Airesweb ya que varios módulos que lo componen, incluso con los que MOPID tiene que interactuar están desarrollados en dicho lenguaje.

MOPID, como se muestra en la figura 17, tendrá como clases principales la clase Controladora, que se encargará de manejar todas las operaciones que realizará dicho módulo, la clase Sesión la que guardará los datos de cada una de las sesiones y por último la clase Algoritmo. De esta última heredan las clases de Algoritmo Apriori y Algoritmo GeneradorDePatronesSecuenciales en las cuales se buscarán las reglas de asociación y los patrones secuenciales.

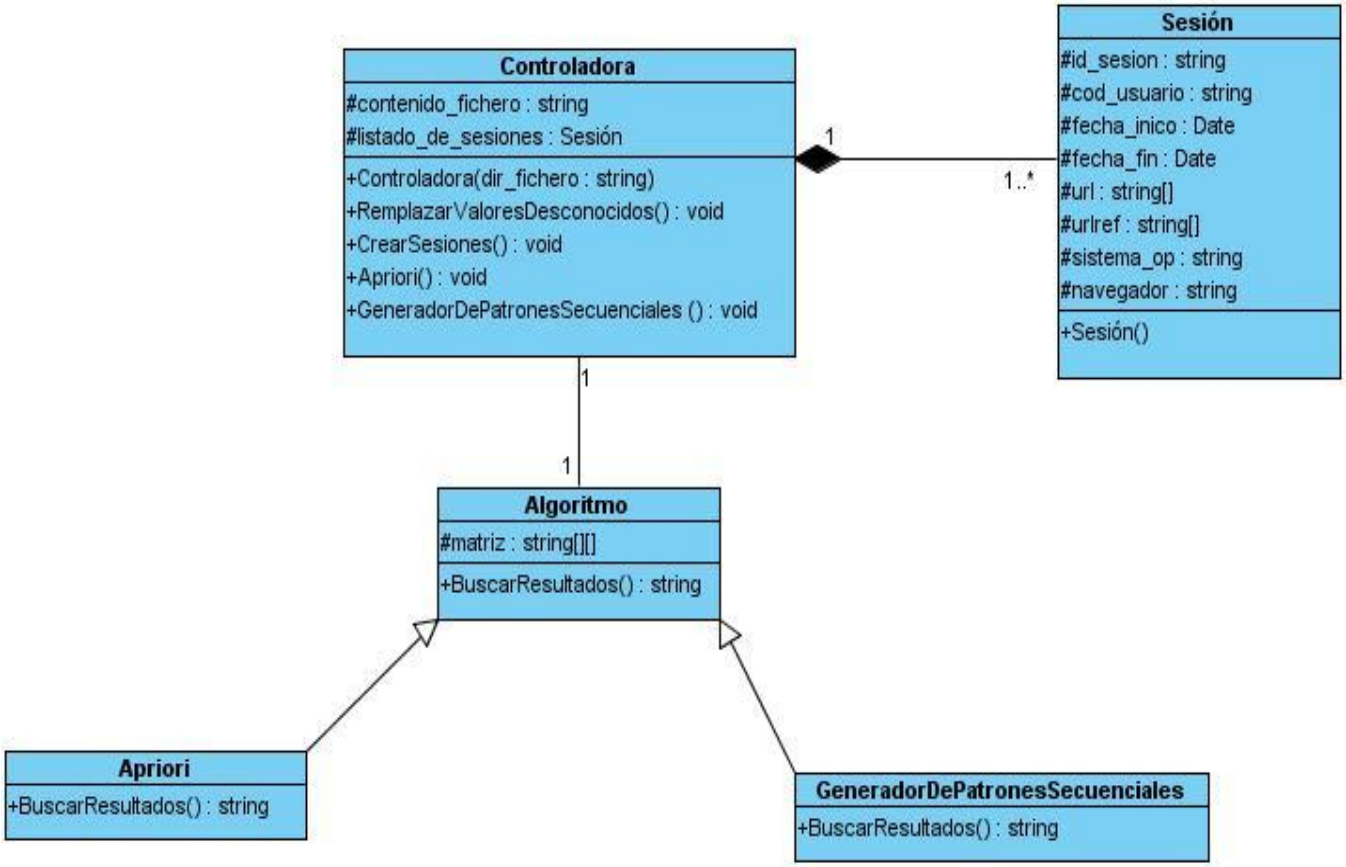

Figura 17 Diagrama de Clases de MOPID

El Controlador Principal de Airesweb le pasará a la clase Controladora los ficheros de registros web, a los que se les va a aplicar los algoritmos. Esta clase es la encargada de crear los objetos de la clase Sesión, aplicándole primeramente el método RemplazarValoresDesconocidos para remplazar los datos desconocidos por la moda, ya que son de tipo cadena. Una vez creadas todas las sesiones, mandará a ejecutar las clases Apriori y GeneradorDePatronesSecuenciales obteniendo las reglas de asociación y los patrones secuenciales. Por último la clase Controladora analizará los resultados obtenidos y después le enviará al Controlador Principal el resultado de este análisis, para que lo guarde en la Base de Datos de Airesweb y sean consultados y mostrados mediante la interfaz web.

Las instancias de la clase Sesión guardarán todos los datos de una sesión de usuario. Esta tendrá como atributos el identificador de la sesión, el código del usuario, la fecha y hora de que inició la sesión, la fecha y hora en que la finalizó, un arreglo con todas las direcciones URL que ha visitado ese usuario, otro arreglo paralelo con las direcciones de URL de referencia, el sistema operativo y el navegador que utilizó el usuario. Este proceso de identificación de sesión se hará como anteriormente se describió. Una sesión tendrá las acciones de un mismo usuario en un intervalo de 30 min. El identificador de la misma es un número que empezará desde el 1 hasta que se hallan identificado todas las sesiones.

Antes de realizar el algoritmo Apriori se tiene que parametrizar los datos que se guardaron en las sesiones. Con este fin primeramente se crea una matriz de sesiones llamada MS, siendo una matriz mxn donde m es la cantidad de filas que representa el número de sesiones creadas y n es la cantidad de columna que representa el número total de páginas visitadas por los usuarios, es decir el elemento MS[i,j] representa la página j en la sesión i. Si un usuario visita la página j en una sesión i, entonces el valor MS[i,j] sería un 1, en caso de que la página j no fue visitada en la sesión i, entonces el valor sería 0.

Esta matriz es la que se le pasa por parámetro a la clase Apriori así como los valores del soporte y confianza mínimos iniciales para realizar el algoritmo Apriori. En este caso se propone que se comience con un soporte de 0.02 debido a que se trabajará con grandes cantidades de datos y con una confianza de 0.4 para obtener diversos resultados de interés. Estos valores fueron seleccionados de manera empírica, a partir de las pruebas realizadas al algoritmo. Si el algoritmo no arroja suficientes resultados los valores del soporte y confianza mínimos iniciales serán disminuidos en la clase Controladora para luego volverlo a realizar.

En este algoritmo se realizarán varias iteraciones por los datos con que se cuentan, el cual se realizará de la siguiente forma:

- Primeramente se buscarán las URL individuales con la representación de los índices de la matriz MS, con lo cual se obtendrá los primeros grupos de premisas de las reglas de asociación que cumplan con el soporte mínimo. Por ejemplo suponiendo que se tiene solo 12 sesiones y el soporte mínimo es de 0.5, lo que representa el 50% de todas las sesiones, entonces las URL seleccionadas como premisas serán las que aparezcan en al menos 6 sesiones diferentes o filas de la matriz.
- Con este conjunto de URL como premisas, se busca en el resto de las columnas de la matriz la conclusión de cada regla, es decir para determinada página se busca otra página que también

fue visitada en esa misma sesión, pasando esta última a ser la conclusión de la regla de asociación. Luego se calcula la confianza de las reglas de asociaciones obtenidas para desechar aquellas reglas que no cumplan con la confianza mínima. El valor de la confianza de una regla sería la cantidad de sesiones en que fueron visitadas tanto las premisas como las conclusiones entre la cantidad de sesiones en que se visitaron las premisas nada más. Por ejemplo suponiendo que determinada página seleccionada como premisa fue visitada en 6 sesiones y la página seleccionada como conclusión solo se visitó en 4 de estas sesiones entonces la confianza de la regla sería de 4/6=0.67.

- Con las reglas de asociación anterior se forman conjuntos candidatos de páginas visitadas siempre que sean válidos, es decir que las páginas del conjunto sean visitadas en la misma sesión. Luego de estos conjuntos candidatos solo se convertirían a premisas si cumplen con el soporte mínimo. Se volvería a realizar una pasada por los datos buscando para cada uno de esto conjuntos la página que se tomaría como conclusión y a estas reglas se le calcularía la confianza para desechar las que no cumplan con la confianza mínima. Esto se repetiría hasta que no se encuentren más premisas que cumplan con el soporte mínimo establecido por el usuario o se hayan analizado todos los datos con los que se cuentan.
- El formato de las reglas de asociación será de la siguiente forma: primero estarán entre corchetes [] el conjunto de páginas que integran las premisas. Cada página estará separada de la otra mediante una coma. Al conjunto de premisas le seguirá del signo (→) indicando que la página a continuación del mismo es la conclusión de la regla. Después de cada regla separada de un espacio se especificará la confianza de la misma.

Una vez que se obtienen las reglas de asociación en la clase Controladora, son analizadas obteniéndose los patrones interesantes en forma de cadena de caracteres los cuales son guardados en la Base de Datos.

Para realizar el algoritmo GeneradorDePatronesSecuenciales es necesario crear una matriz de transición llamada MT, siendo una matriz nxn, es decir una matriz cuadrada con la misma cantidad de filas que columnas, donde n son todas las páginas visitadas por los usuarios, por lo que la MT[i,j] representa el número de los usuarios que han visitado la página j después de visitar la página i. Después de creada la matriz MT, se crea otra matriz la cual se deriva de la primera, solo que en las celdas MT[i,j] que tenga un valor menor que el soporte mínimo establecido se actualizarán con el valor cero.

La matriz obtenida y el soporte mínimo que deben cumplir los patrones secuenciales se pasan como parámetro a la clase GeneradorDePatronesSecuenciales. Al igual que en Apriori, si con el soporte mínimo inicial no se obtienen resultados suficiente, se disminuye el mismo y se vuelve a realizar el algoritmo, este al igual que en el Apriori tendrá un valor de 0.02.

Con esta matriz se generan los caminos frecuentes, los cuales se almacenan en una estructura de árbol, donde cada nodo o hoja del árbol representa una página web, la cual tiene como datos el identificador de la página así como la fecha en que fue visitada por primera y última vez. La raíz de un árbol sería la página de donde se empieza a navegar, y el primer hijo, la página que visitó el usuario después de haber visitado la raíz. El árbol termina cuando todas las páginas sean hojas y una página es hoja cuando de ella no se visita ninguna otra página. Cada vez que se analice una página esta se marca como analizada, por lo que cuando no se pueda seguir más por un camino o árbol se busca la primera página que no ha sido aún analizada. Con estos árboles se buscan los patrones secuenciales frecuentes. Esto se hace de la forma siguiente: se recorren los árboles desde la raíz hasta la hoja, calculando el soporte del camino el cual sería el mínimo soporte de todas las páginas, cada vez que se encuentre una hoja entonces se puede crear un patrón secuencial, donde los antecedentes serían todas las páginas visitadas antes de llegar a la hoja y el consecuente sería la hoja encontrada calculando la confianza del mismo, que no es más que el soporte de consecuente entre el soporte de antecedente.

El formato de los patrones secuenciales será de la siguiente forma primero estarán entre corchetes [] el conjunto de páginas que integran los patrones secuenciales donde cada página estará separada de la otra mediante una coma. Al conjunto de antecedentes le seguirá del signo (→) indicando que la página a continuación del mismo es la consecuente del patrón secuencial. Después de cada patrón secuencial separado de un espacio se especificará la confianza del mismo.

Los resultados obtenidos de este algoritmo son analizados en la clase Controladora obteniéndose en forma de cadena de caracteres los patrones no triviales los cuales luego son guardados en la Base de Datos.

2.8 Evaluación de los Resultados

Cuando un portal web contiene una enorme cantidad de información y además es visitado por un gran número de personas como es el caso de los sitios de la prensa, se hace necesario que dicha información esté bien estructurada, de manera tal que se pueda acceder a ella fácilmente. Esto constituye un aspecto muy importante para el desarrollo exitoso del portal. El módulo propuesto en esta investigación, proporciona el análisis necesario para determinar el uso real que se está haciendo

del sitio web en cuestión, ofreciéndole información al administrador que le podría ayudar en la toma de decisiones técnicas y organizativas sobre el sitio.

El análisis de asociaciones y secuencias son de gran utilidad a la hora de plantearse una posible mejora o rediseño del sitio, así como a la hora de localizar posibles fallos o errores en su creación. Las reglas de asociación obtenidas mediante el prototipo funcional implementado son usadas para mostrar correlaciones entre páginas accedidas durante las mismas sesiones, además indican una posible relación entre páginas visitadas simultáneamente aunque no se encuentran conectadas de forma directa, y relaciones entre grupos de usuarios sin intereses específicos.

La tabla siguiente muestra las reglas de asociación obtenidas a partir del procesamiento de un conjunto de 100 transacciones.

No Regla	Reglas de Asociación
Regla 1	airesweb.f10.uci.cu/index.php/register➔airesweb.f10.uci.cu/index.php/default
Regla 2	airesweb.f10.uci.cu/index.php/default➔airesweb.f10.uci.cu/index.php/default/ayuda_1
Regla 3	airesweb.f10.uci.cu/index.php/default➔airesweb.f10.uci.cu/index.php/report
Regla 4	airesweb.f10.uci.cu/index.php/default/ayuda_1➔airesweb.f10.uci.cu/index.php/default/report
Regla 5	airesweb.f10.uci.cu/index.php/report➔airesweb.f10.uci.cu/index.php/report/content_1

Tabla 2.5 Reglas obtenidas durante el proceso.

No de Regla	Soporte	Confianza
Regla 1	0.4	0.9
Regla 2	0.28	0.74
Regla 3	0.10	0.1
Regla 4	0.15	0.8
Regla 5	0.08	0.05

Tabla 2.6 Medidas para las reglas obtenidas.

Analizando los resultados se puede señalar que las reglas más representativas son la regla 1 y 2 con un 40% y 28% respectivamente. Esto significa que el comportamiento más común de los usuarios en este conjunto de datos es registrarse en el sitio y luego visitar la página *ayuda_1*. También se puede apreciar que el vínculo hacia la página *report* en la página *default*, no capta la atención de los usuarios que visitan el sitio, ya que solo el 10 % de los usuarios que acceden a *default* van hacia la página *report*. Sin embargo no sucede lo mismo en el vínculo de la página *ayuda_1* a la página *report* pues 80% de los usuarios que acceden a la página *report* lo hacen a través de *ayuda_1*, dando lugar a la interrogante de si está bien organizada la arquitectura de información. Esto se debería comprobar visitando la página en específico y viendo si es interés marcado que en la página *default* se siga el vínculo hacia la página *report*, de ser así, se debería revisar la posición en el sitio de dicho vínculo. La decisión final no solo se debe basar en analizar fríamente los números, también se debe conjugar todas las variables posibles en el entorno del sitio web, para lograr la optimización de la navegación de los usuarios.

Otro aspecto a tener en cuenta es que solo el 5% de los usuarios que accedieron a la página *report* accedieron desde la misma, a la página *content_1*, por lo que se debería hacer un análisis del contenido de la página *content_1* para comprobar si esta es de interés para los usuarios que visitan *report*. Esto permitiría encontrar posibles errores a la hora de la creación de la página en cuestión.

Los patrones secuenciales obtenidos mediante el prototipo funcional implementado son usados para determinar el comportamiento de los usuarios con respecto al tiempo. Con esa información se puede predecir las futuras visitas al sitio web y así poder organizar mejor los accesos al mismo.

La siguiente tabla muestra los patrones secuenciales obtenidos a partir del procesamiento de un conjunto de 100 transacciones.

No. Patrón	Patrones Secuenciales	Tiempo	Soporte
Patrón 1	[airesweb.f10.uci.cu/index.php/default,airesweb.f10.uci.cu/index.php/default/report, airesweb.f10.uci.cu/index.php/report/demografia_1]	7:00 a 11:59	0.7
Patrón 2	[http://airesweb.f10.uci.cu/index.php/default,airesweb.f10.uci.cu/index.php/options_1,http://airesweb.f10.uci.cu/index.php/default]	7:00 a 11:59	0.5
Patrón 3	[airesweb.f10.uci.cu/index.php/default,airesweb.f10.uci.cu/index.php/default/ayuda_2]	12:00 a 20:00	0.1

Tabla 2.7 Patrones secuenciales obtenidos

Analizando los resultados obtenidos se puede observar una tendencia de que el 70% de los usuarios visitaron la página *demografía_1* en el horario de 7 a 11:59 de la mañana, siguiendo una secuencia desde la página *default* pasando por la página *report*. De esta información se puede deducir que en este horario se publicó en las páginas una información interesante para los usuarios. Esto se puede aprovechar para ubicar en ese horario en las páginas antes mencionadas el contenido de interés que los administradores del sitio precisan que conozcan los usuarios. También se puede apreciar solo el 10% de los usuarios visitaron la página *ayuda_2* desde la página *default*, en el horario de las 12:00 del mediodía a las 8:00 de la noche, por lo que debería realizarse un análisis sobre el contenido que ofrece dicha página en ese horario o si es de mucho interés cambiarlo a una página más visitada.

La creación de MOPID dotará al sistema Airesweb de valiosos resultados. La evaluación detallada de los mismos será de utilidad para los administradores de los sitios a la hora de organizar y mejorar el contenido de las páginas web, y optimizar la navegación en ellas, aumentando así, las funcionalidades y eficiencia de Airesweb.

Conclusiones

Después de estudiar las soluciones existentes a nivel internacional de software de analítica web y sus módulos de procesamiento de datos y procesamiento inteligente de datos, y ninguna contar con las características necesarias para añadirlas al sistema Airesweb se decidió sentar las bases teóricas para el desarrollo de un módulo propio. Con el estudio de las técnicas y los algoritmos de minería de datos y el desarrollo del prototipo funcional se llegaron a las siguientes conclusiones:

- El estudio del proceso de la minería de datos así como de sus algoritmos más utilizados a nivel mundial permitió seleccionar las herramientas y tecnologías con las cuales se probarían estos algoritmos.
- El desarrollo del prototipo funcional permitió comprobar el correcto funcionamiento de los algoritmos seleccionados para la realización de MOPID.
- A partir del estudio de la arquitectura del sistema Airesweb se definió como realizar la integración de MOPID al mismo.
- La evaluación de los resultados obtenidos permitió comprobar cual algoritmo era el más eficiente para cada caso específico.

Como conclusión general, puede afirmarse que se le dieron cumplimiento a todos los objetivos planteados al inicio del trabajo y se verifico la validez de la idea a defender materializada en la solución propuesta.

Recomendaciones

El mundo de la analitica web es uno de lo más cambiantes en el panorama actual de Internet, por lo que contar con una herramienta que pueda estar a la altura de los principales analizadores es una gran ventaja. La implementación de este módulo debería ser una de las principales prioridades para el equipo de trabajo de Aires, por lo tanto se recomienda:

- Implementar en C/C++ los algoritmos propuestos en el trabajo para una mejor interacción con el sistema Airesweb.
- Seguir las especificaciones técnicas y las bases teóricas descritas en el trabajo para el desarrollo completo del módulo.
- Estudiar otros algoritmos de minería de datos y minería web para encontrar otros posibles resultados.
- Probar los algoritmos utilizando la herramienta Weka para tener una idea de como será su comportamiento para que el desarrollo sea más exacto.

Referencias Bibliográficas

1. AVALON. Recursos de internet. [En línea] 2008. [Citado el: 2 de Diciembre de 2009.] Disponible en: <http://www.avalonps.com/rec_conocimientos.asp>.

2. Documento Básico Daedalus. Minería Web. [En línea] Noviembre de 2002. [Citado el: 3 de Diciembre de 2009.] Disponible en: <http://www.daedalus.es>.

3. **Román, Julio, García Vegas, F y González, José C.** *Uso de la Minería Web para mejorar los servicios al ciudadano.* 2002.

4. **Molina, Luis Carlos.** *Data Mining: Torturando a los datos hasta que confiesen.* Universitat Oberta de Catalunya : FUOC, 2002.

5. Google Analytics. [En línea] [Citado el: 15 de Enero de 2010.] Disponible en: <http://www.google.com/intl/es_ALL/analytics/>.

6. WebTrends Analytics. [En línea] [Citado el: 15 de Enero de 2010.] Disponible en: <http://www.isoftland.com/index2.php?option=com_content&do_pdf=1&id=23>.

7. AWStats official web site. [En línea] [Citado el: 17 de Enero de 2010.] Disponible en: <http://awstats.sourceforge.net/>.

8. Hallvarsson halvarsson. [En línea] [Citado el: 22 de Enero de 2010.] Disponible en: <http://www.halvarsson.se/en/Services/Web-consulting/Web-Analytics/Omniture-SiteCatalyst/>.

9. Omniture SiteCatalyst. [En línea] [Citado el: 28 de Enero de 2010.] Disponible en: <http://www.omniture.com/es/products/web_analytics/sitecatalyst>.

10. Estadística web. [En línea] [Citado el: 29 de Enero de 2010.] Disponible en: <http://www.desarrolloweb.com/articulos/2045.php>.

11. Clicktracks - Web Analytics. Análisis de las estadísticas web. [En línea] [Citado el: 30 de Enero de 2010.] Disponible en: <http://www.searchmarketing.pt/WebAnalytics?_Locale=es>.

12. **Fayyad, Usama, Piatetsky-Shapiro, Gregory y Padhraic.** *From Data Mining to Knowledge Discovery in Databases.* Menlo Park CA. : IAAA Press/The MIT Press, 1996.

13. **Piatetski-Shapiro, G y Frawley, J.** *Knowledge Discovery in Databases.* Cambridge, MA. :AAAI/MIT Press, 1991.

14. **Thuraisingham, B.** *Data Mining. Technologies, Techniques, Tools and Trends.* CRC Press LLC, 1999.

15. **Montero Navarro, Miguel.** *Extracción de conocimiento en bases de datos astronómicas.* Sevilla, junio 2009.

16. **Pyle, Dorian.** *Data Preparation For Data Mining.* San Francisco, California : Morgan Kaufmann Publishers, 1999.

17. **Martínez de Pison Ascasibar, J.** *Optimización mediante técnicas de minería de datos del ciclo de recorrido de una línea de galvanizado.* : Universidad de La Rioja, 2003. ISBN 84-688-2870.

18. **Vallejos, Sofía.** *Minería de Datos.* Argentina : Universidad Nacional del Nordeste, 2006.

19. **Herrera Varela, R.** *Bibliomining: minería de datos y descubrimiento de conocimiento en bases de datos aplicados al ámbito bibliotecario.* Diciembre 2006.

20. **Hernández Orallo, J., Ramírez Quintana, M. J. y Ferri Ramírez, C.** *Introducción a la minería de datos.* Madrid. Universidad Politécnica de Valencia : PEARSON EDUCACIÓN, 2004. ISBN 84-205-4091-9.

21. **Westphal, Christopher y Blaxton, Teresa.** *Data Mining Solutions. Methods and Tools for Solving Real-World Problems.* USA : John Wiley & Sons, 1998.

22. **Méndez Cáceres, Lesley.** *Un Algoritmo Basado en la Programación Genética para la Extracción de Reglas de Asociación.*

23. **Fu, L.** *Neural Networks in Computer Intelligence.* New York : McGraw Hill, 1994.

24. Metodología CRISP-DM para minería de datos. [En línea] 2007. [Citado el: 2 de Febrero de 2010.] Disponible en: <http://www.dataprix.com/modelo_crisp-dm>.

25. **Acosta Sánchez, Rolando, Vázquez Martín, Laura y Brito Sarasa, Raycos.** *Empleo de Minería de Datos para la Obtención de Patrones en el Sistema.* 2007.

26. **Acosta Sánchez, R., y otros.** *Minería de Datos para la predicción de causas de diabetes. Preprocesado de datos.* 2008.

27. IBM SPSS Modeler Professional. [En línea] [Citado el: 4 de Febrero de 2010.] Disponible en: <http://www.spss.com/software/modeling/modeler-pro/>.

28. STATISTICA Enterprise. [En línea] [Citado el: 4 de Febrero de 2010.] Disponible en: <http://www.statsoftiberica.com/es/soluciones/empresa/sol_empr_satentreprise.html#Sol>.

29. Data mining with SAS Enterprise Miner. [En línea] [Citado el: 4 de Febrero de 2010.] Disponible en: <http://www.sas.com/technologies/analytics/datamining/miner/>.

30. **Sanchez Sanchez, J.** *Análisis de accesos a un servidor web de contenidos dinámicos.* 2006.

31. **Hernandez Orallo, J y Ramírez, Ferri.** *Curso de Doctorado Extracción Automática de Conocimiento en Bases de Datos e Ingeniería del Software.* Madrid : Universidad Politécnica de Valencia.

32. Orange. [En línea] [Citado el: 4 de Febrero de 2010.] Disponible en: <http://www.ailab.si/orange/>.

33. RapidMiner oficial web site. [En línea] [Citado el: 4 de Febrero de 2010.]Disponible en: <http://rapid-i.com/>.

34. **Acosta Aguilera, M.** *Minería de Datos y Descubrimiento de Conocimiento.* 2004.

35. **Esquivel Gámez, Ismael.** *Tratamiento automático de noticias empresariales en la Web mediante minería textual.* Mexico : Universidad Popular Autónoma del Estado de Puebla, 2007.

36. **Zamarrón Sanz, C., y otros.** *Aplicación de la Minería de Datos al Estudio de las Alteraciones Respiratorias durante el sueño.* Hospital Clínico Universitario de Santiago. 2006.

37. **Virseda Benito F, Román Carrillo R.** *Minería de datos y aplicaciones.* Universidad Carlos III. 2007.

38. **Ortega Morán, Juan F.** *Cálculos de modos y tiempos de desplazamientos en una ciudad usando fuentes públicas.* 2008.

39. Web mining en el diseño de sitios web. [En línea] [Citado el: 12 de Febrero de 2010.] Disponible en: <http://www.webtaller.com/maletin/articulos/web-mining-diseno-sitios-web.php>.

40. **Fuentes Reyes, S. C.; Ruiz Lobaina, M.** Minería de Texto: Aplicación de Web Mining. [En línea] Disponible en: <http://www.idict.cu/UserFiles/File/Trabajos%20de%20Jornada%20Bibliotecaria/Fuentes%20Reyes,%20Sady%20Carina%20_%20Web%20Mining.pdf>.

41. **Fuentes Reyes, S. y Ruiz Lobaina, M.** *Minería web: un recurso insoslayable para el profesional de información.* 2007.

42. **Liu, Bing.** *Web Data Mining.* New York : Springer Berlin Heidelberg, 2007. ISBN-10 3-540-37881-2.

43. **Martin Guerrero, J.** *Determinación de tendencias en un portal web utilizando técnicas no supervisadas. Aplicación a sistemas de recomendaciones basados en filtrado colaborativo.* Valencia : Universidad de Valencia, 2004.

44. **De Gyves, F.** *Web Mining: Fundamentos Básicos.* [En línea] [Citado el: 23 de Febrero de 2010.] Disponible en: <http://zarza.usal.es/~fgarcia/doctorado/iweb/05-07/Trabajos/WMINING.pdf.>.

45. **Kosala, R. y Blockeel, H.** *Web Mining Research: A Survey.* : ACM SIGKDD Explorations. Newsletter of the Special Interest Group on Knowledge Discovery and Data Mining, 2000.

46. **Araque Cuenca, Hurtado Torres, Samos Jiménez.** *Extracción de Información de Fuentes de Datos Heterogéneas e Incorporación al Data Warehouse.* : Universidad de Granada, 2000.

47. **Chang, G., Healey, M.J., McHugh, J.A.M. y Wang, J.T.L.** *Mining the world wide web: An information search approach.* Norwell, MA : Kluwer Academic Publishers, 2001.

48. **Escobar Jeria, Víctor.** *Minería Web de Uso y Perfiles de Usuario: Aplicaciones con Lógica Difusa.* Granada : Editorial de la Universidad de Granada, 2007. ISBN: 978-84-338-4707-2.

49. **J., Kleingberg.** *Authoritative source in hyperlinked environment.* : Journal of the ACM, 1999.

50. **Mobasher, Bamshad.** *Web Usage Mining and Personalization.* : CRC Press LLC, 2004.

51. **Srivastava, Jaideep, y otros.** *Web Usage Mining: Discovery and Applications of Usage Paterns from Web Data.* : SIGKDD Explorations, ACM, Enero, 2000.

52. **Gonzalo Torres, Antonio.** *Minería web y personalización: Revisión bibliográfica y propuesta de un marco de referencia.* 2005.

53. **González Sánchez, G., Delfín Ávila, S. y Lluís de la Rosa, J.** *Preprocesamiento de bases de datos masivas y multi-dimensionales en la minería de uso web para modelar usuarios: comparación de herramientas y técnicas con un caso de estudio.* 2005.

54. **Theodoridis, S. y Koutroumbas, K.** *Pattern recognition.* : Academic Press., 1999.

55. **Spiliopoulou, M. y Faulstich, L. C.** *WUM: A Web Utilizattion Miner. Proceedings of the international workshop on the web and databases.* Valencia , Marzo de 1999.

56. **Mobasher, B, y otros.** *Web Mining: Pattern Discovery from World Wide Web Transactions.* : Proceedings of the 9th IEEEInternational conference on tools with artificial intelligence (ICTAI'97), Noviembre, 1997.

57. **Ye, Nong.** *The Handbook of data mining.* . : HUMAN FACTORS, E. Arizona State University., 1999.

58. **Reyes Saldaña, José y García Flores, R.** El proceso de descubrimiento de conocimiento en bases de datos. *Ingenierías.* Enero-Marzo 2005, Vol. VIII, 26.

59. **Armas Santos, Arazay.** *Herramientas para la detección de errores usando reglas de asociación.* Villa Clara, 2009.

60. **Han, Jiawei y Kamber, Micheline.** *Data Mining: Concepts and Techniques.* : The Morgan Kaufmann Series in Data Management Systems, 2001.

61. **Mondragón Becerra, R.** *EXPLORACIONES SOBRE EL SOPORTE MULTI-AGENTE BDI EN EL PROCESO DE DESCUBRIMIENTO DE CONOCIMIENTO EN BASES DE DATOS.* Diciembre 2007.

62. **Serrano González, Miguel García.** *Evolución de proyecciones lineales para aprendizaje automático.* Julio 2009.

63. **Srikant, Ramakrishnan y Agrawal, Rakesh.** *Mining Sequential Patterns: Generalizations and Performance Improvements.* 1996.

Bibliografía

1. **Riquelme, J, Ruiz, Roberto y Gilbert, K.** *Minería de Datos: Conceptos y Tendencias.* 2006.

2. **Zaïane Osmar, R.** *Principles of Knowledge Discovery in Data.* Canadá : Universidad de Alberta, 2007.

3. **Brachman, R. y Anand, T.** *The process of Knoviledge discovery in databases: A human centered approach, Advances in Knowledge Discovery and Data Mining.* AAAIIIMIT Press, 1996.

4. **Pascual, D., Sanchez, S. y Pla, F.** *Algoritmos de agrupamiento.* 2007.

5. **González Gómez, J.** *Generalidades de la Minería de Datos.* 2007.

6. **Carmona del Jesus, Cristóbal, y otros.** *Aplicación de un algoritmo de extracción de reglas difusas para minería de uso web .* Mieres – Langreo, Septiembre 2008.

7. **Ruiz Sánchez, Roberto.** *Heurísticas de Selección de Atributos para Datos de Gran Dimensionalidad.* Sevilla : Universidad de Sevilla, Junio 2006.

8. **Román, Ulises y Alarcón, Luis.** *Minería de Uso de Web para Predicción de Usuarios en la Universidad.* Universidad Nacional Mayor de San Marcos, 2005. ISSN: 1816-3823.

9. **Alcívar Zambrano, Patricio, Idrovo Chiriboga, Fanny E. y Macas Pizarro, Víctor H.** *Sistema de análisis de patrones de navegación usando Minería Web.* Guayaquil- Ecuador, 2007.

10. **Bouckaert, Remco, y otros.** *Weka Manual or Version 3-6-0.* New Zealand : University of Waikato, 2008.

11. **Castaño P., Andres P.** Minería de uso web para para la identificación de patrones. *Vector.* Enero - Diciembre 2009, Vol. 4.

12. **Daza Portocarrero, Luis A.** *Métodos para mejorar la calidad de un conjunto de datos para descubrir conocimiento .* Universidad de Puerto Rico, Julio 2007.

13. **Servente, Magdalena.** *Algoritmos TDIDT aplicados a la Minería de Datos Inteligente.* Universidad de Buenos Aires, 2002.

14. **Corso, Cynthia y Alfaro, Sofía.** *Algoritmos de Data Mining aplicados a la enseñanza basada en la Web.* Córdoba, Argentina.2009.

15. **Baeza-Yates, Ricardo y Poblete, Bárbara.** *Una herramienta de minería de consultas para el diseño del contenido y la estructura de un sitio web.* 2005. ISBN: 84-9732-449-8.

16. **Wilford Rivera, Ingrid, Rosete Suárez, Alejandro y Rodríguez Díaz, Alfredo.** *Análisis de Información Clínica mediante técnicas de Minería de Datos. Estudio Experimental.* 2008.

17. **Gallardo Campos, Margarita.** *Aplicación de tecnicas de clustering para la mejora del aprendizaje.* Leganés : Uiversidad Carlos III de Madrid, 2009.

18. **Escobar Jeria, Víctor.** *Minería Web de Uso y Perfiles de Usuario: Aplicaciones con Lógica Difusa.* Granada : Editorial de la Universidad de Granada, 2007. ISBN: 978-84-338-4707-2.

19. **Srikant, Ramakrishnan y Agrawal, Rakesh.** *Mining Sequential Patterns: Generalizations and Performance Improvements.* 1996.

20. **Liu, Bing.** *Web Data Mining.* New York : Springer Berlin Heidelberg, 2007. ISBN-10 3-540-37881-2.

21. **García Jiménez, María y Álvarez Sierra, Aránzazu.** *Análisis de Datos en WEKA – Pruebas de Selectividad.* 2007.

Figuras Relacionadas

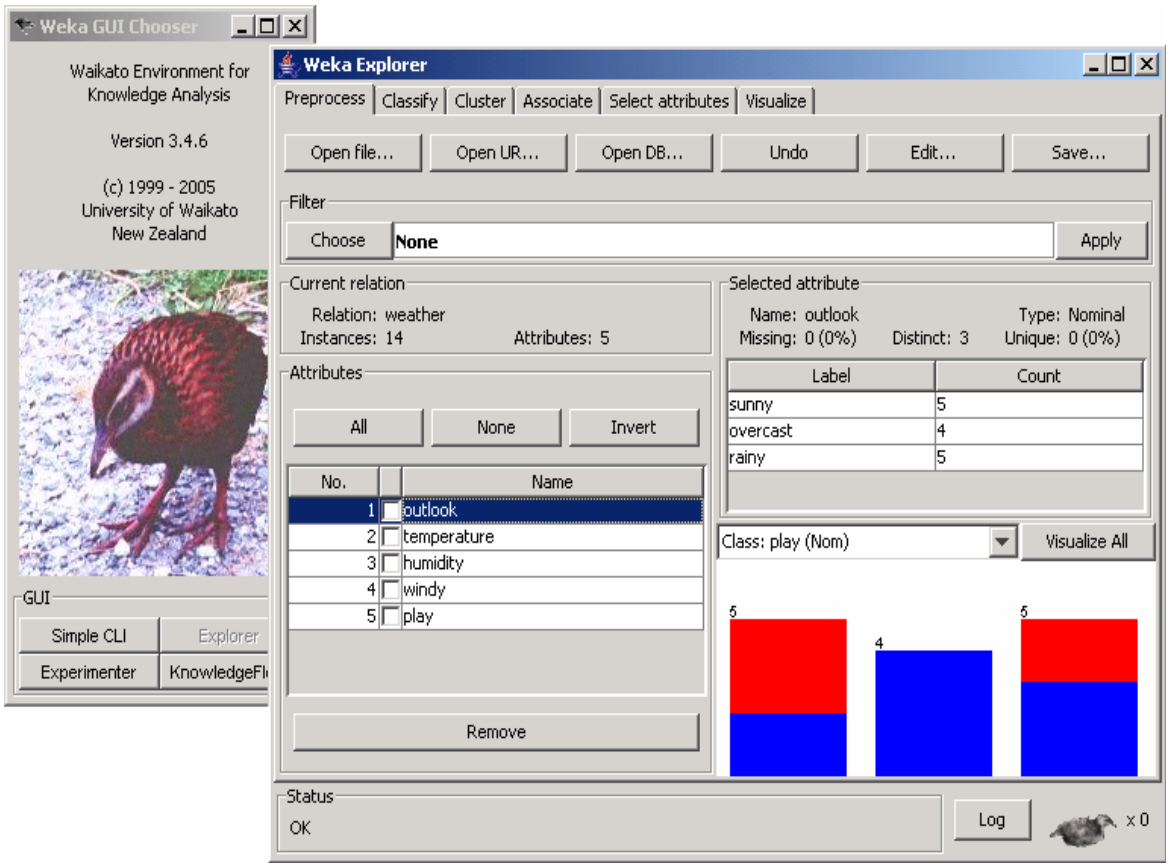

Figura A.1. Entorno gráfico de Weka Explorer, con opciones de preprocesamiento.

Figura A.2. Entorno gráfico de WUM.

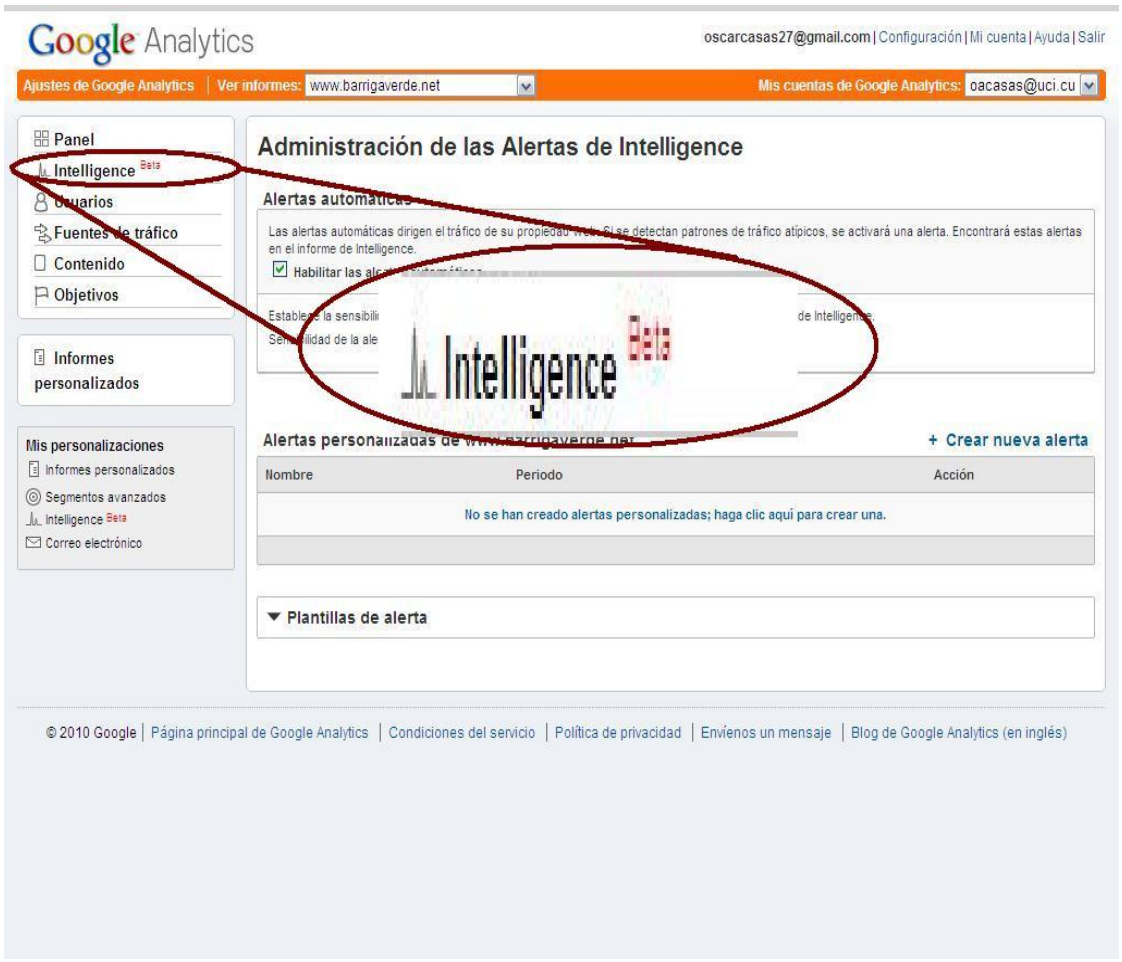

Figura A.3. Interfaz web de Google Analytics.

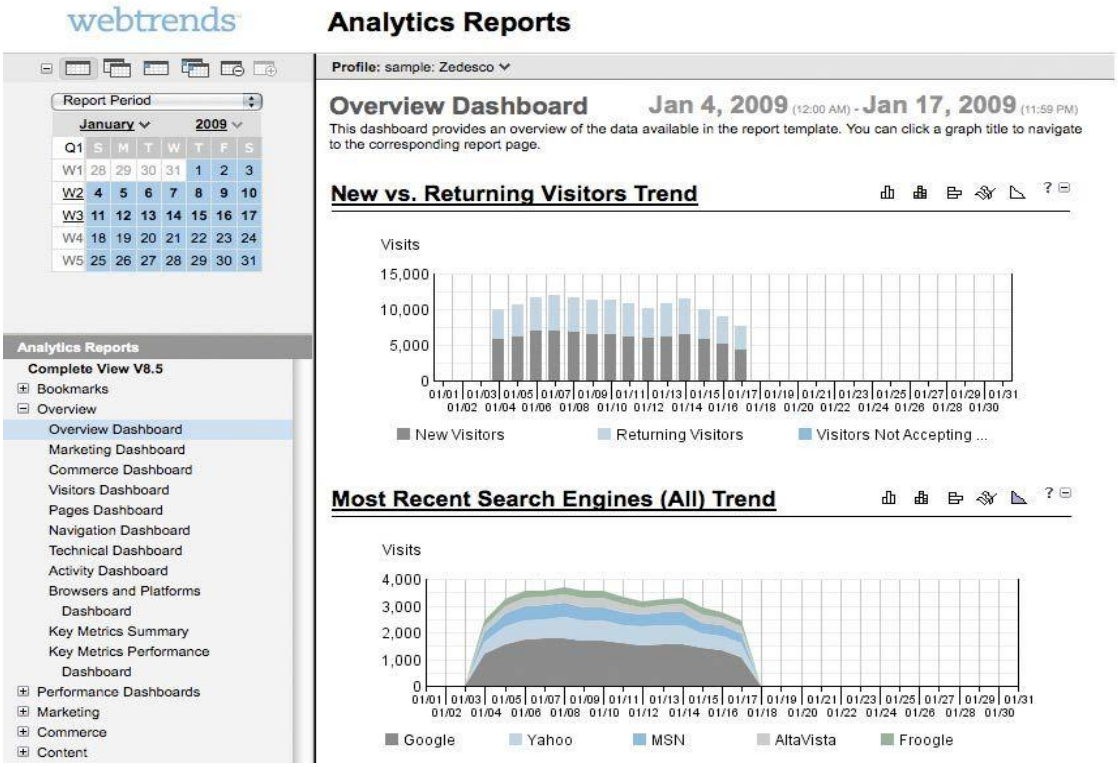

Figura A.4. Interfaz web WebTrend.

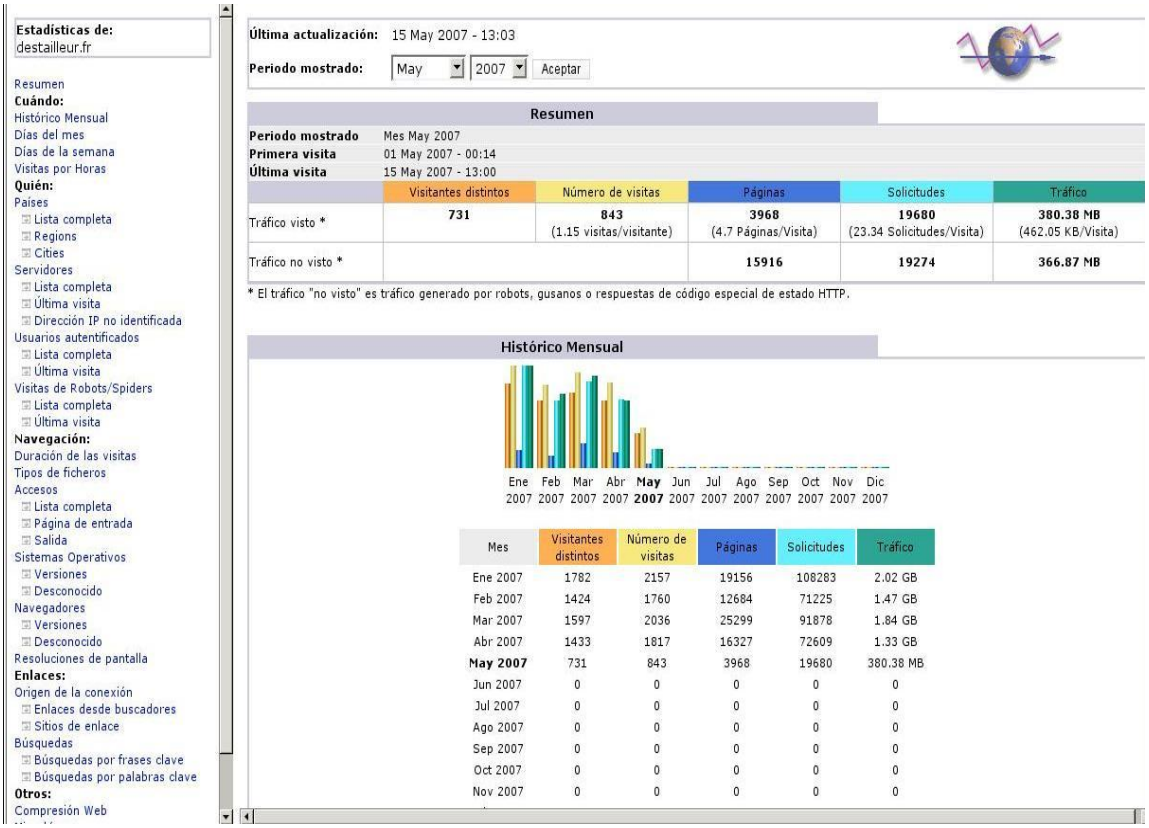

Figura A.5. Interfaz web de AWStats.

Figura A.6. Interfaz web de Omniture SiteCatalyst.

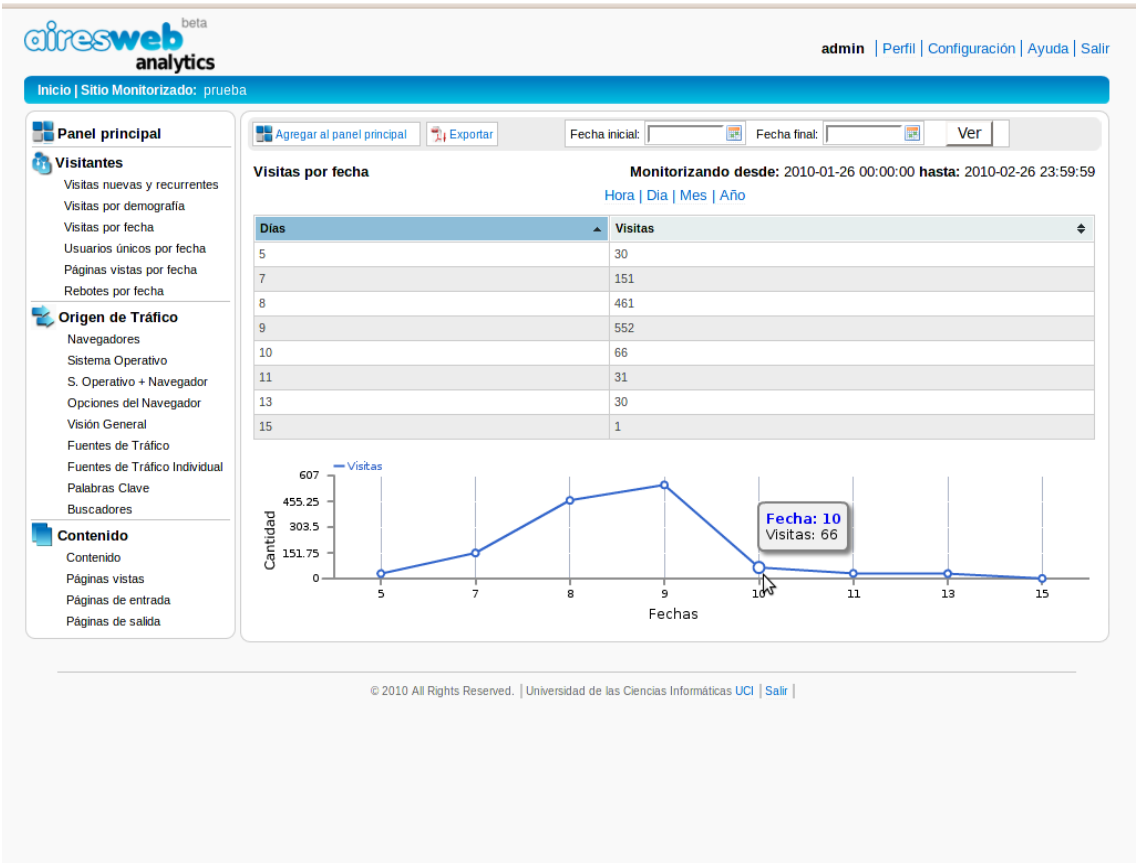

Figura A.7. Interfaz web de Airesweb.

Glosario de Términos

Airesweb Analizador Inteligente de Registros de Servidores Web.

API Interfaz de Aplicación.

Apriori Algoritmos de búsqueda de reglas de asociación.

BD Base de Datos.

CIPRE Centro de Información para la Prensa.

GSP Algoritmo de búsqueda de patrones secuenciales.

Minería de datos Proceso mediante el cual se convierten los datos en conocimiento.

Minería web Proceso mediante el cual se convierten los datos almacenados en los servidores web en conocimiento.

Minería web de uso Proceso mediante el cual se convierten los datos relativos al comportamiento de los usuarios en los sitios web.

MOPID Módulo de Procesamiento Inteligente de Datos.

Registro web Comúnmente llamado Log, es un registro oficial de eventos durante un periodo de tiempo en particular.

Sesgo Propiedad de una muestra estadística que hace que los resultados sean representativos para toda la población.

UCI Universidad de las Ciencias Informáticas, la cual fue creada, por el comandante en jefe de la Revolución cubana, al calor de la batalla de ideas, en el año 2002.

Weka Waikato Environment for Knowledge Analysis.