Análisis de Reportes de Seguridad Sobre Plataformas LCMS de Tipo Open Source Aplicando Minería de Datos

Diana Paola Castañeda Talero
Universidad Distrital Francisco José de Caldas, Facultad de Ingeniería, Bogotá, Colombia
dipacata@hotmail.com

José Luis Rodríguez Reyes
Universidad Distrital Francisco José de Caldas, Facultad de Ingeniería, Bogotá, Colombia
jlzero88@hotmail.com

Ing. Paulo Alonso Gaona García
Universidad Distrital Francisco José de Caldas, Facultad de Ingeniería, Bogotá, Colombia
pagaonag@udistrital.edu.co

RESUMEN

Son amplios los dominios sobre los cuales la minería de datos permite un análisis profundo de información. Bases de datos como las manejadas por los desarrolladores y administradores de las plataformas LCMS (Learning Content Management System) de tipo Open Source, generan grandes volúmenes de información, lo cual resulta sumamente valioso a la hora de reportar vulnerabilidades y fallos dentro de la plataforma, para definir líneas de acción para posibles correcciones, pero igualmente generan un alto consumo de tiempo para lograr ubicar las más representativas según el grado de impacto de cada una de ellas. En el presente artículo se presenta el resultado del análisis de la información contenida en la base de datos de la herramienta Moodle Tracker, encargada del manejo de los reportes de la plataforma LCMS Moodle, usando como apoyo el paquete de herramientas para minería de datos WEKA, a fin de clasificar y agrupar los posibles puntos vulnerables dentro de la misma.

Palabras clave: Agrupamiento, Clasificación, Sistemas de gestión de contenidos de aprendizaje, Minería de datos, Reglas de asociación.

ABSTRACT

Are broad domains on which data mining allows a thorough analysis of information. Databases such as those run by the developers and managers of the LCMS (Learning Content Management System) Open Source Platforms, generate large volumes of information, which is extremely valuable in reporting vulnerabilities and weaknesses in the platform, to define lines of action for any corrections, but also generate a high consumption of time to achieve the most representative place according to the impact degree of each. This article presents the results of the information's analysis contained in the Moodle Tracker tool database, responsible for handling reports of the LCMS Moodle, using as support the toolkit WEKA data mining, to classify and group potential vulnerabilities within it.

Keywords: Association Rules, Classification, Clustering, Data Mining, Learning Content Management System.

1. INTRODUCCIÓN

El desarrollo de la Inteligencia Artificial (IA) y su cada vez mayor aplicación, ya no sólo en simples proyectos académicos, que redundan en lo teórico y llegan a parecer simples "ejemplos de juguete", sino ahora también en

la industria, en aplicaciones del mundo real, es tan sólo una muestra de las muchas tecnologías que están haciendo una transición de lo teórico a lo práctico; ejemplos concretos de esto son las redes neuronales, los agentes inteligentes y la minería de datos. En el caso de esta última, la minería de datos pretende hacer frente a las complicaciones típicas presentadas al realizar consultas de tipo SQL y permite también la extracción de conocimiento valioso que no es inmediatamente visible. El descubrimiento de conocimiento en bases de datos mediante las herramientas de minería de datos, combina las técnicas tradicionales de búsqueda con numerosos recursos desarrollados en el área de la inteligencia artificial, las matemáticas, la estadística y la teoría de bases de datos (Chen y Han, 1996).

Gracias a las técnicas de minería, el conocimiento que se puede adquirir a partir de la información embebida en las bases de datos, deja de ser de tipo evidente, como el obtenido mediante sentencias SQL, o del tipo multidimensional, como el que se logra con la tecnología de almacenes de datos y estructuras OLAP, y más bien se explora un tipo de conocimiento oculto, que resulta potencialmente útil al tratarse de información no evidente (Villena, 2009).

2. MINERÍA DE DATOS

La minería de datos consiste en la aplicación de una gran cantidad de métodos, para el procesamiento y análisis de datos. Para esto se basa en dos conceptos: "escarbar y explotar". Así, grandes volúmenes de datos son tratados mediante diversos procesos para permitir el descubrimiento de información no evidente, elementos de utilidad y comportamientos interesantes como: cambios, anomalías, estructuras significativas y patrones de comportamiento para aplicarlos a nuevos conjuntos de datos. El objetivo primordial de la minería de datos es el aprovechamiento de las características hombre-máquina, es decir, la mezcla entre flexibilidad, creatividad y conocimiento general con la potencia de cálculo y almacenamiento. Esto para poder realizar una exploración de datos efectiva y veraz, con el fin de construir un sistema computacional que sea capaz de extraer y modelar el conocimiento no visible. La minería de datos resulta especialmente útil en los casos en los que el conjunto de datos a analizar es demasiado grande, por la cantidad de instancias presentes, o complejo, por la cantidad de variables presentes. Además, al hablar de búsquedas automatizadas, también cabe suponer el caso en el que el especialista encargado del tratamiento de estos datos no se encuentre disponible. Así, aparecen características como el descubrimiento automatizado de modelos desconocidos, la aceleración en el procesamiento de los datos y la capacidad de predicción de tendencias, haciendo que la minería de datos se convierta en una tecnología apta para la toma de decisiones tácticas y estratégicas, claro, utilizando el concepto de automatización para la búsqueda e identificación de conocimiento útil. En la figura 1, se observa el proceso correspondiente a la minería de datos.

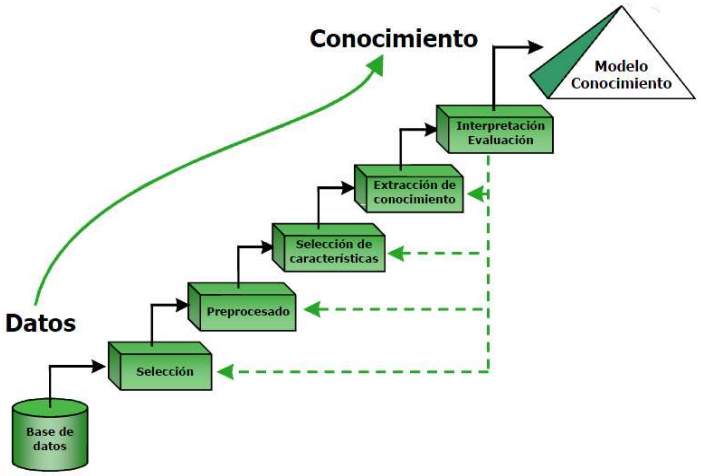

Figura 1. Proceso de Minería de Datos. Tomada de Villena, J. 2009. Inteligencia en Redes de Comunicaciones: Minería de Datos

3. SISTEMAS DE GESTIÓN DE CONTENIDOS PARA EL APRENDIZAJE

Las plataformas o sistemas LCMS se especializan en la gestión de cursos virtuales, usando así el principio de aprendizaje colaborativo. Estos sistemas forman un ambiente virtual de aprendizaje, que facilitan las labores del profesor al permitirle manejar los diferentes ámbitos inmersos en un curso, como por ejemplo, la elaboración de contenido, la formulación de cuestionarios y la charla con los estudiantes (Dans, 2009). Esto significa una nueva oportunidad de aprendizaje para personas que por diversos motivos, como el tiempo o el lugar de residencia, no pueden acceder a cursos presenciales; pero además son una muy buena herramienta complementaria de la educación presencial, ya que se garantiza una pedagogía teórica práctica integral.

3.1 ANÁLISIS DE VULNERABILIDADES REPORTADAS POR PROYECTOS LCMS

A partir de la definición de LCMS vista anteriormente, son muchos los proyectos que cumplen con las especificaciones recomendadas por estándares internacionales como ISO/IEC 9126 (Márquez y Capdevila, 2009) para plataformas de este tipo, sin embargo, para realizar el estudio, era necesario escoger sólo una de ellas, ya que sería complejo tratar de analizar más de una base de datos, teniendo en cuenta el inmenso volumen de información que se maneja. En principio, se consideraron seis de los proyectos más destacados en el ámbito internacional, a saber: .LRN, Sakai, ATutor, Dokeos, Claroline y Moodle. Las seis plataformas se analizaron mediante un estudio, realizado previamente y pendiente de publicación, que evaluaba las plataformas basándose en dos cosas: los mecanismos de seguridad manejados, y los reportes de seguridad generados en un período determinado de tiempo.

De acuerdo al primer ítem de evaluación, se determinó que Moodle era el proyecto con el mayor número de mecanismos de seguridad implementados, 10 de los 12 que se tenían en cuenta para el estudio (figura 2), incluyendo: Módulos de autenticación mediante LDAP, PAM, Kerberos y e-mail; generación de certificados mediante SSL y TS; uso de protocolo HTTPS, manejo de listas de control de acceso, compatibilidad con servidores CAS, manejo de permisos.

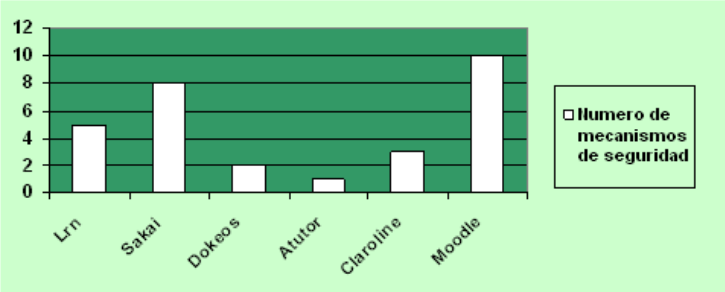

Figura 2. Cantidad de mecanismos de seguridad por plataforma.

En cuanto al segundo ítem de evaluación, la mayoría de las plataformas fueron difíciles de analizar desde el punto de vista del manejo de reportes de seguridad, ya que ni siquiera contaban con un aplicativo que permitiera el acceso a la base de datos que maneja los reportes de seguridad; tan sólo Sakai y Moodle poseían herramientas que manejaban de manera correcta dichos reportes. A partir de estos resultados, se consideró a la plataforma Moodle como LCMS apta para el estudio de minería de datos.

3.2 MOODLE

Moodle es uno de los LCMS más usados actualmente, ya que cuenta con muchas versiones estables. La plataforma fue diseñada para obedecer al enfoque llamado educación social constructivista, por lo que no sigue otras recomendaciones estándar de accesibilidad y uso.

La plataforma cuenta con una base de datos accesible mediante la herramienta Moodle Tracker que se encarga del registro y la gestión de errores, mejoras, y peticiones de nuevas características de Moodle. Cada vez que un administrador o un usuario de Moodle encuentra algún fallo, quiere reportar la implementación de una nueva

característica del sistema, o simplemente quiere dar a conocer características inherentes a algún suceso que haya ocurrido con el sistema, este crea un reporte que se ingresa en la base de datos con el fin de llevar estadísticas de todos y cada uno de ellos. Estas estadísticas permiten identificar, por ejemplo, cuántos y cuáles de los reportes corresponden a fallos, cuáles de estos han sido resueltos, que reportes han sido asignados a alguno de los administradores de Moodle o las fechas en las que más se han presentado fallos.

4. CASO DE ESTUDIO

Al observar las características de Moodle, queda claro que, al tratarse de una plataforma que maneja información valiosa, se hace necesario el contar con mecanismos que ayuden a identificar los posibles fallos de seguridad del sistema, y así evitar que personas malintencionadas la alteren o la eliminen. También podría darse el caso de que delincuentes se hagan pasar como administradores de Moodle, ingresando a los datos contenidos en la base de datos de Moodle Tracker perturbando la información contenida en ella. Aunque las estadísticas arrojadas por la herramienta Moodle Tracker permiten conocer algunas características básicas de los reportes generados, se considera que mediante un análisis exhaustivo utilizando técnicas de minería de datos, es posible encontrar información valiosa para los administradores de la plataforma. Como se mostrará más adelante, los resultados que se obtienen al realizar este tipo de análisis de la base de datos permite, por ejemplo, conocer cuáles son los tipos de reporte más comunes en el sistema, de que tipo son, si han sido resueltos o no, etc., posibilitando a su vez una clasificación de los mismos en varios grupos, todo con el fin de que, cuando se registre un nuevo reporte, este se clasifique de manera automática dentro de uno de los grupos encontrados.

5. ANÁLISIS Y CONSTRUCCIÓN DE MODELOS DE REPRESENTACIÓN

5.1 OBJETIVOS

De acuerdo al enunciado del caso de estudio, se planteó analizar la base de datos Moodle Tracker, seleccionando los atributos más relevantes de los reportes pertenecientes a esta, y determinando algunas reglas de asociación que permitan determinar los principales patrones encontrados en la base de datos, para una posterior clasificación en grupos de los mismos. Como no existen aún estudios anteriores basados en minería de datos para esta fuente, las técnicas para la extracción de conocimiento que se utilizan son descriptivas, más no predictivas. Para el cumplimiento de esto, se plantean objetivos específicos como:

- Determinar un modelo que permita la clasificación de todos los reportes.
- Determinar un modelo de clustering que permita el agrupamiento correcto de los reportes.
- Determinar algunas reglas de asociación para tratar de encontrar patrones dentro de los datos obtenidos.

5.2 SELECCIÓN DE ATRIBUTOS E INSTANCIAS

5.2.1 ATRIBUTOS

La base de datos Moodle Tracker cuenta con un total de 32 atributos o columnas, de los cuales se eligieron 9, los de mayor relevancia para el experimento. Los atributos seleccionados se muestran en la tabla 1.

Se resalta el hecho de que, analizando el atributo componente, algunos de los reportes abarcan situaciones que afectan a más de uno de ellos, en este caso, fue necesario para efectos prácticos asumir cada componente por separado al momento de la preparación de los datos, que se explica más adelante, dando pie a tantas nuevas instancias como fuera necesario. Si por ejemplo un reporte incluía situaciones referentes a 3 componentes, se realizaban 3 instancias independientes con todos los valores iguales para los ocho atributos restantes, obviando por supuesto el valor del atributo componente.

Tabla 1: Atributos Seleccionados Para el Experimento

Atributo	Descripción
tema	Se refiere al tema al cual hace referencia el reporte
estado	Indica el estado actual del reporte
prioridad	Grado de importancia que se le da al reporte
solucion	Determina si ya se atendió el suceso Reportado, o si no se ha hecho, define el por qué
mes_creacion	Mes en el que se reportó el suceso.
version_afectada	Versión de Moodle sobre la cual se registra el error.
version_reparada	Versión de Moodle que corrige la situación reportada.
componente	Nombre del componente de Moodle afectado por la situación reportada.
mes_solucion	Mes en el que se soluciona completamente la situación reportada.

5.2.2 INSTANCIAS

Las instancias o campos de la base de datos corresponden a cada uno de los reportes registrados en esta. Ya que la preparación de los datos es un proceso bastante engorroso y que demanda la mayor cantidad de esfuerzos en el proceso de minería de datos, no se tuvo en cuenta el total de las instancias, sino que solo se tomaron aquellas correspondientes al año 2009, es decir, se tomo una muestra de 3209 instancias de las 20609 existentes en el momento. Las 3209 instancias son reportes generados entre el 1º de enero del 2009 y el 1º de enero del 2010. Este conjunto se amplió hasta llegar a las 4089 instancias, luego de la separación de los reportes con más de un componente asociado, de acuerdo a lo reseñado en la sección anterior.

5.3 COMPUTADOR DE PRUEBA

5.3.1 SOFTWARE

En el mercado existen varios paquetes que manejan las técnicas y algoritmos de minería de datos, tanto de licencia comercial como de licencia GPL (Vilena, 2009). El software elegido para el experimento fue WEKA, software de código libre, en su versión 3.5.3. WEKA, acrónimo de Waikato Environment for Knowledge Analysis, es una colección de librerías desarrolladas en lenguaje JAVA, que implementa bastantes técnicas para el análisis, verificación y evaluación de datos, y que tiene como valor agregado la posibilidad de prestar funcionalidad dentro de desarrollos de software propios.

5.3.2 HARDWARE

Para manejar el volumen de información de la base de datos bastó con una máquina personal común y corriente. Estas son sus especificaciones: CPU Intel Celeron D @ 2.8 GHz; 2Gb de memoria RAM; sistema Operativo Windows XP Professional Service Pack 3; dispositivos de entrada y salida. El tamaño del disco duro es relevante ya que no se trabajo un volumen de datos tan amplio como para que este se tenga en cuenta.

5.4 PREPARACIÓN DE LOS DATOS

Como se explicó anteriormente, el conjunto de datos seleccionado para el experimento estaba conformado por 4089 instancias, las cuales estaban definidas por un total de 9 atributos. Una vez se tenía el conjunto a tratar, era necesario hacer una transformación en los datos, esto para que fueran compatibles con el formato natural de WEKA; así, se hizo necesario pasar la tabla de datos, obtenida en formato .xls, a un archivo de texto simple que posteriormente se convirtió en el archivo .ARFF. Los atributos pueden ser principalmente de dos tipos: numéricos de tipo real o entero (real o integer), y simbólicos (especificando los valores posibles que pueden tomar entre

llaves). Para el caso particular de estudio, se utilizaron tipos de datos simbólicos. Los valores simbólicos que puede tomar cada una de los atributos se consigan en la tabla 2.

Una vez se tienen las cabeceras hechas, se añaden los datos al final de estas (muestreados como se ha comentado anteriormente). Se observa que existen atributos tales como componente, version_afectada, que cuentan con un número muy extenso de posibles valores simbólicos; como posteriormente se mostrará, esto generó alguna repercusión a la hora de clasificar y agrupar las instancias.

Tabla 2. Posibles valores simbólicos para los atributos seleccionadas

Atributo	Descripción
tema	bug,improvement,sub_task,task,new_feature
estado	resolved,open,closed,in_progress,reopened
prioridad	minor,major,trivial,blocker,critical
solucion	fixed,unresolved,not_bug,duplicate,wont_fix,cannot_reproduce,incomplete,deferred
mes_creacion	ene,feb,mar,abr,may,jun,jul,ago,sep,oct,nov,dic.
version_afectada	20,197,19,198,192,1811,21,195,196,194,193,186,18,189,17,191,182,188,184,16,183,169,187,172,176
version_reparada	20,198,197,1812,21,196,195,1810,193,194,189
componente	authentication,themes,gradebook,general,html_editor,blog,backup,wiki_2x,networking,ajax,roles,filters,lib,feedback,languages,questions,blocks,installation,course,administration,usability,forum, forms_library,performance,database,sql_xmldb,portfolio_api,unknown,groups,chat,activity_module,calendar,files_api,accessibility,repository_api,resource,web_services,glossary,quiz,other,assignment,workshop,documentation,scorm,wiki_1x,exercise,lesson,unit_tests,enrolments,rss,hotpot,global_search,progress,tracking,messages,conditional_activities,events_api,my_moodle,licensing,tags,choice,survey,commenting,phpdoc,ims_resource_type,unicode,maths_filters,journal,lams,document_management,security_alert
mes_solucion	ene,feb,mar,abr,may,jun,jul,ago,sep,oct,nov,dic

6. DESARROLLO DE MODELOS Y RESULTADOS EXPERIMENTALES

Una vez los datos fueron seleccionados, codificados y correctamente tratados, se cargan en WEKA para empezar el análisis. A fin de mostrar un buen número de herramientas y utilidades que se encuentran en este, y en la mayoría de paquetes de minería de datos, se aplicaron todas las técnicas descriptivas, es decir, las orientadas a la exploración y el análisis del conjunto de datos tratado a fin de ayudar en el proceso de toma de decisiones. En este grupo de técnicas catalogadas como descriptivas se resaltan los algoritmos de clasificación, de agrupamiento (o clustering) y de generación de reglas de asociación. Es bueno recordar que, la minería de datos también permite el uso de estadística simple; en ese sentido, antes de comenzar con el análisis mediante los algoritmos de minería, se estudiaron las estadísticas arrojadas al iniciar el explorador de WEKA y cargar los datos. Se observó que el 50% de reportes que se ingresan a Moodle Tracker corresponden a fallos (bug) en el sistema, mientras tanto, los reportes generados a partir de tareas asignadas (task) y a la adición de nuevos componentes de la plataforma (new_feature), tan sólo llegan al 3% y 7%, respectivamente. Para la obtención de los porcentajes correspondientes a cada grupo se utilizó un algoritmo de agrupamiento que se explicará más adelante.

6.1 ALGORITMOS DE CLASIFICACIÓN

Tienen como propósito la clasificación de un conjunto de datos a partir del valor de sus atributos. Tales algoritmos buscan propiedades en común entre los conjuntos de instancias, para luego clasificarlos en distintas

clases mediante algún modelo. Al determinar el modelo de clasificación, es necesario utilizar un conjunto de entrenamiento, donde el objetivo es analizar dichos datos de entrenamiento y, mediante un método supervisado, desarrollar un modelo para cada una de las clases, a partir de las características de los datos. Los algoritmos de clasificación más comunes son: algoritmos basados en árboles de decisión (BFTree, C 4.5, J48), basados en reglas (ZeroR, OneR, PART), en redes neuronales, en el paradigma de aprendizaje de Bayes y finalmente, el algoritmo del vecino más cercano. En el experimento realizado se optó por la utilización de los algoritmos PART (reglas) y J48 (árboles), básicamente por sus demostrados bajos tiempos para crear el modelo y clasificar los datos, además de que arrojan resultados bastante acertados (Sierra, 2006). El modo de prueba del experimento en WEKA es "Use training set". Con esta opción, WEKA entrenará el método con todos los datos disponibles y luego lo aplicará otra vez sobre los mismos. Esto se hace ya que no se cuenta con un conjunto de entrenamiento definido al tratarse del primer experimento realizado a la base de datos. El atributo objetivo elegido fue tema, y no se suprimieron atributos para la generación del modelo.

6.1.1 ALGORITMO PART

Como no se modificaron parámetros del algoritmo, cabe tener en cuenta que el mínimo número de objetos para cada clasificación, por defecto es de 2. Los resultados obtenidos son:

- Número de reglas: 323
- Tiempo para crear el modelo: 1.8 segundos.
- Instancias clasificadas correctamente: 2704 (66.1286 %)
- Instancias clasificadas incorrectamente: 1385 (33.8714%)

A partir de la figura 3, se observa que la cantidad de instancias clasificadas correctamente es muy regular; acá se hacen visibles las repercusiones de las que se hablaba en el apartado de Preparación de datos (sección 5.4). Como se advirtió, la considerablemente grande cantidad de valores posibles para algunos de los atributos, sumado a que la cantidad de datos no es lo suficientemente grande como para contrarrestar este efecto, hace que sea más difícil clasificar de manera correcta las instancias evaluadas.

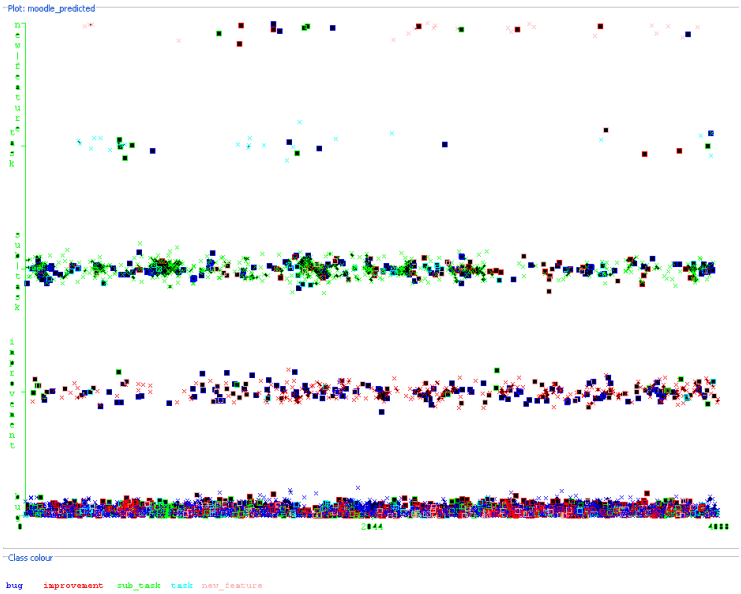

Figura 3. Gráfica correspondiente a la clasificación mediante algoritmo PART (instancias clasificadas correctamente - 66.13%)

6.1.2 ALGORITMO J48

No se modificó ninguno de los parámetros respecto al algoritmo anterior. Los resultados fueron:

- Número de hojas del árbol: 530
- Tiempo para crear el modelo: 0.59 segundos.

- Instancias clasificadas correctamente: 2639 (64.539 %)
- Instancias clasificadas incorrectamente: 1450 (35.461%)

La cantidad de instancias clasificadas correctamente es, al igual que con PART, muy regular. Sin embargo, se puede observar que en este caso, el algoritmo tardó menos tiempo en la construcción del modelo. Aunque WEKA da la posibilidad de mostrar el árbol generado, se omite su visualización en el artículo debido a su tamaño considerablemente grande (530 hojas).

6.2 ALGORITMOS DE AGRUPAMIENTO (CLUSTERING)

Este tipo de algoritmos agrupa los datos no teniendo en cuenta algún tipo de clasificación, sino basándose en la similitud de los valores de sus atributos; los datos normalmente no se encuentran etiquetados. El clustering de los datos es pertinente a la hora de tener una gran cantidad de datos, es por esto que ha tomado gran importancia dentro de la minería de datos. Los datos se agrupan, normalmente, siguiendo un criterio, como por ejemplo, minimizar algún objetivo de acuerdo a alguna medida de similitud. Los algoritmos de agrupamiento más utilizados son: SOM (Self Organizing Maps), que se basa en las redes neuronales y el principio de mapas topológicos; el algoritmo K-Means, realiza agrupamiento por vecinidad, partiendo de un determinado número de prototipos y de un número de ejemplos a agrupar; finalmente está el algoritmo EM, que de manera iterativa pretende realizar una estimación máxima de verosimilitud de los parámetros.

En el experimento particular, el algoritmo, seleccionado para aplicar el agrupamiento sobre los datos fue K-Means (Sierra, 2006), que es tal vez el más utilizado debido a su sencillez de manejo. De nuevo, el modo de prueba del experimento en WEKA es “Use training set”, tomando el atributo tema como objetivo.

6.2.1 ALGORITMO K-MEANS

El algoritmo se aplicó dos veces al mismo conjunto de datos. En el documento solamente se incluye los resultados del primero de ellos, ya que se consideró como relevante. El algoritmo se aplicó, en este primer caso, para un total de 5 clusters (grupos), con el fin de verificar que la información estadística generada anteriormente que describía el conjunto de datos de acuerdo a los valores del parámetro tema, corresponde a los resultados del agrupamiento con el algoritmo de minería. El grupo de datos se clasifica correctamente un 100% de las veces, lo cual se debe a que el número de clusters generados es igual al total de posibles valores simbólicos del atributo. Tal y como se observa, el grupo donde mayo concentración de instancias hay es el cluster 1 (rojo), que corresponde al grupo que tienen a bugs (fallos) como valor para el atributo tema. El porcentaje de instancias clasificadas dentro de cada cluster es obviamente el mismo que se obtiene con el análisis estadístico corriente (50% bugs, 20% improvement, 19% sub_task, 7%new_feature, 3% task). Como comentario adicional, al aplicar el algoritmo por segunda vez, en este caso, tomando tan sólo 3 clusters, el grupo de datos no se clasifica de manera correcta por completo, ya que cabe la posibilidad de que alguna instancia sea incluida en alguno de los 3 grupos disponibles. Según el resultado del algoritmo, los temas característicos que clasifican la totalidad de instancias, y su correspondiente frecuencia de son: bugs 50%, sub_task 30% e improvement 20%. Las instancias pertenecientes a los otros temas se clasifican dentro de estos tres grupos.

Además de estos dos análisis mediante clustering, se realizaron algunos otros, pero teniendo en cuenta no sólo uno, sino dos o más atributos, intentando capturar algunas reglas de asociación básicas que permitieran encontrar patrones en los reportes. Por ejemplo, uno de estos análisis consistió en la creación de 20 clusters, teniendo en cuenta los atributos tema y estado, a fin de encontrar las relaciones más frecuentes dentro del conjunto de datos, sin embargo, el hecho de que se contemplen tantos valores posibles una vez más perjudica la veracidad del agrupamiento. Los resultados obtenidos demostraron una vez más que la mayoría de reportes encontrados corresponden a bugs en el sistema, además, se comprobó que es muy frecuente encontrar los reportes aún sin solución, sobre todo los que pertenecen al tema de bugs. En la figura 4 se observa cómo, para los cinco temas posibles, la mayoría de veces no se tiene aún una solución al reporte (puntos rojos). Hay que anotar que es mucho mejor tratar de definir reglas mediante algoritmos diseñados para esta tarea concreta; para mostarlo, en la siguiente sección se verán las reglas más interesantes obtenidas con ese método, que sin duda muestran de una

manera mucho más clara el grado de ocurrencia de las asociaciones, con grados de confianza elegibles a juicio propio.

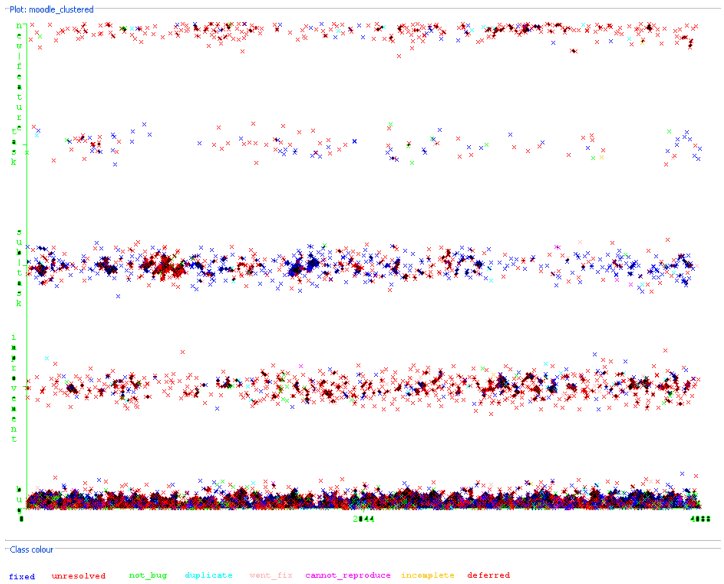

Figura 4. Gráfica correspondiente a la distribución de clusters, analizando en conjunto los atributos tema y solución

6.3 REGLAS DE ASOCIACIÓN

Este tipo de algoritmos centran sus esfuerzos en encontrar reglas, las cuales sean una sentencia probabilística basada en la co-ocurrencia de eventos particulares dentro de una base de datos. De esta forma es aplicable a grandes volúmenes de datos. Existen diferentes algoritmos que hacen uso de las reglas de asociación, pero sin duda el más destacado es el algoritmo Apriori (Sierra, 2006).

6.3.1 ALGORITMO APRIORI

En el experimento, el algoritmo Apriori se basó en el estudio de 3 de los atributos (tema, prioridad y solución), y permitió encontrar varias reglas interesantes. Los parámetros que se configuraron al algoritmo fueron el número máximo de reglas a mostrar (25), y que un margen de confianza de mínimo el 60%.

6.4 RESULTADOS GENERALES DEL EXPERIMENTO

Aunque la clasificación obtenida fue relativamente buena, es claro que los resultados se podrían mejorar dejando a un lado atributos con un conjunto amplio de valores de tipo simbólico, o realizando un análisis para una cantidad de instancias mucho mayor, pero para esto se necesitaría una cantidad de tiempo mucho más considerable, ya que, como se pudo verificar, la selección y el preprocesado de los datos a analizar constituyen el mayor esfuerzo en el proceso de minería. Se pudo comprobar, mediante los algoritmos de agrupamiento que, la mayor cantidad de reportes de seguridad corresponden a fallos en el sistema, que podrías ser posibles puntos vulnerables para la seguridad de la plataforma. Además se encontró que WEKA no realiza una correcta aproximación al momento de trabajar con números decimales, lo cual se comprueba al sumar los porcentajes correspondientes a cada uno de los 5 clusters inicialmente generados (99% y no 100% como por supuesto debería ser). Teniendo en cuenta el análisis de reglas de asociación realizado con el algoritmo Apriori, se pueden generar algunas conclusiones respecto a los reportes:

- Si el reporte está catalogado como bug, de las 2049 instancias de este grupo, el 55% (1127 reportes) tendrán una prioridad de resolución menor que la del resto. Además, cuando se trata de bugs, el 54% de los casos reportados (1109) no tendrán una solución registrada asociada.
- De los reportes que han sido atendidos (solucion = fixed - 1244), el 66% de ellos tenían una prioridad menor asociada (824)

- Cuando ingresan reportes con prioridad menor para su tratamiento (2679), el 61% de ellos aparecerán sin una solución asociada (1638)

Se podría continuar haciendo una lista de reglas que sin duda, en manos de un administrador de la plataforma, le permitiría modificar y mejorar sus políticas a la hora de resolver los fallos en el sistema, pero, con el fin de no extender más los resultados consignados aquí, se considera como suficiente la información hasta acá obtenida.

7. CONCLUSIONES

Se logró verificar que, además de que es factible aplicar la minería de datos para el análisis de los reportes de seguridad de las plataformas LCMS Open Source, la importancia de esta se resalta al momento de generar conocimiento que permita a los administradores generar políticas de seguridad más adecuadas. La minería de datos demostró ser una tecnología mucho mejor que la estadística, cuando se requiere de la generación de modelos de identificación de patrones descriptivos en las fuentes de información.

En el proceso de análisis de datos de fuentes correspondientes plataformas LCMS, los algoritmos de agrupamiento y de reglas de asociación demostraron ser los mejores para la definición de patrones y reglas. Si se deseara realizar el análisis de cualquier otra plataforma de este tipo, se recomienda el uso de algoritmos de este tipo, ya sea los incluidos en el paquete WEKA, o los propuestos por otras herramientas para minería de datos similares.

Mediante WEKA se obtuvieron modelos clasificatorios, de agrupamiento y de asociación que permitieron encontrar patrones comunes en los reportes de seguridad. Se aclara que la fiabilidad de los modelos podría ser muy superior si se analiza una cantidad de instancias considerablemente mayor, y si además se eliminan atributos que manejen rangos muy grandes de valores posibles.

REFERENCIAS

Arboleda, A. y Bedón, C. (2001). "*Sistema de detección de intrusos utilizando inteligencia artificial",* Tesis de Pregrado, Universidad del Cauca, Colombia.

Chen, M., y Han, J. 1996. *"An overview from database perspective*" IEEE Transactions on Knowledge and Data Eng.

Dans, E. (2009). "*Educación online: plataformas educativas y el dilema de la apertura"*. Artículo Monográfico en línea. En: "Cultura digital y prácticas creativas en educación". Revista de Universidad y Sociedad del Conocimiento (RUSC).Vol. 6, n. º 1. UOC.

Hernández, J., Ramírez, M. y Ferri, C. (2004). *"Introducción a la Minería de Datos". Prentice Hall, Madrid. (ISBN: 8420540919).*

Márquez, O., y Capdevila, M. (2009). "*Plataformas de tele-enseñanza de Software Libre"*.

Perversi, I. (2007). *Aplicación de Minería de Datos para la exploración y detección de patrones delictivos en Argentina. Tesis de pregrado en línea, Instituto Tecnológico de Buenos Aires, Argentina.*

Sangüesa, R., y Molina L.C. (2000). *Data Mining, una introducción. Ediciones UOC. 1ª Edición.*

SCORM, Sharable Content Object Reference Model (2010), http://www.scormcourse.com/, 04/12/10. (date accessed)

Sierra, B. (2006). *"Aprendizaje Automático: conceptos básicos y avanzados. Aspectos prácticos utilizando el software Weka",* Prentice Hall, Madrid, 2006 (ISBN: 848322318X).

Villena, J. (2009). *"Inteligencia en redes de comunicaciones: Minería de datos"*

Wilford, I., Rosete, A. y Rodríguez, A. (2009). *Análisis de Información Clínica mediante técnicas de minería de datos,* Artículo en línea. En: "RevistaeSalud.com".Vol. 5, n. º 20. ISSN: 1698-7969

Authorization and Disclaimer