Aplicación de algoritmos evolutivos de descubrimiento de subgrupos en e-learning: un caso de estudio analizando cursos de Moodle

C. Romero, P. González, S. Ventura, M.J. del Jesus, F. Herrera

Resumen— **En este trabajo se describe la aplicación de técnicas de descubrimiento de reglas de descripción de subgrupos utilizando algoritmos evolutivos sobre los datos de utilización del sistema de enseñanza a distancia Moodle. El objetivo consiste en extraer reglas que describan relaciones entre el uso que hacen los alumnos de las distintas actividades y módulos que proporciona el sistema de e-learning y la calificación final que obtienen en las asignaturas. Para ello se comparan los resultados obtenidos por diferentes algoritmos clásicos de descubrimiento de subgrupos frente a la propuesta de dos algoritmos evolutivos difusos, uno mono-objetivo y otro multiobjetivo. Los resultados muestran la adecuación de los algoritmos evolutivos en este problema.**

Palabras clave—**Descubrimiento de subgrupos, enseñanza a distancia, algoritmos evolutivos, reglas difusas.**

I. INTRODUCCIÓN

En la actualidad existe un incremento de la aplicación de técnicas de minería de datos a los sistemas de enseñanza a distancia basados en web o sistemas de e-learning [17]. Su principal objetivo es descubrir información útil para el profesor con el objetivo de hacer una evaluación continua y un mejor mantenimiento de los cursos, además de descubrir información útil para la mejora del rendimiento o aprendizaje de los propios alumnos.

Uno de los modelos de minería de datos ampliamente utilizados es la obtención de reglas de asociación [2], reglas descriptivas que permiten descubrir relaciones entre atributos de un conjunto de datos que superan unos determinados umbrales.

El descubrimiento de subgrupos es un caso específico de obtención de reglas descriptivas que tiene como objetivo descubrir propiedades características de subgrupos para los cuales se ha fijado una característica concreta (representada como consecuente de la regla) [11]. Se construyen reglas *sencillas* (con una estructura comprensible y con pocas variables), *altamente significativas* y con *alto soporte* (que cubran muchas instancias de la clase objetivo).

C. Romero y S. Ventura pertenecen al Dpto. de Informática, Universidad de Córdoba, 14071 Córdoba. E-mail: {cromero,sventura}@uco.es. P. González y M.J. del Jesus pertenecen al Dpto. de Informática, Universidad de Jaén, 23071 Jaén. E-mail: {pglez,mjjesus}@ujaen.es. F. Herrera pertenece al Dpto. de Ciencias de la Computación e IA, Universidad de Granada, 18071 Granada. E-mail: herrera@decsai.ugr.es

Una descripción de subgrupo tiene la forma de regla

$$Condición \rightarrow Etiqueta\_de\_Clase$$

donde la propiedad de interés para el descubrimiento de subgrupos es el valor de la variable (*Etiqueta_de_Clase*) que aparece en el consecuente de la regla, y el antecedente de la regla (*Condición*) es una conjunción de características (parejas atributo-valor) seleccionadas de entre las características que describen a las instancias de entrenamiento.

La extracción de reglas de asociación se ha aplicado con éxito en sistemas de e-learning para descubrir relaciones o asociaciones entre distintas páginas web visitadas, actividades realizadas, puntuaciones obtenidas, etc. Uno de trabajos pioneros en la aplicación de técnicas de minería web a los sistemas de e-learning es [20] en el que se propone el uso de agentes [21] para recomendar actividades de aprendizaje en línea o atajos en un curso web basándose en los historiales de acceso de los alumnos, mejorando de este modo el proceso de aprendizaje en línea. Otro trabajo que utiliza técnicas de minería de reglas de asociación y filtrado colaborativo es [18] para descubrir patrones de navegación útiles y proponer un modelo de navegación. En [9] se propone el uso de métodos tales como la regresión lineal en combinación con reglas de asociación para obtener modelos de transferencia de aprendizaje de estudiantes a partir de los ficheros *log* de interacción en sistemas tutores inteligentes. En [19] se describe la utilización de reglas de asociación difusas para descubrir relaciones entre patrones de comportamiento de los estudiantes, incluyendo el tiempo de acceso, números de páginas leídas, preguntas contestadas, mensajes leídos y enviados, etc. En [16] se utiliza algoritmos evolutivos como técnica de descubrimiento de información útil para los autores de este tipo de cursos, con el objetivo de poder realizar mejoras a nivel de contenido, estructura de y adaptación de los cursos. Otro trabajo que también emplea algoritmos evolutivos es el realizado en [12], que realiza un análisis de asociación para predecir el rendimiento de los estudiantes. Estos mismos autores también utilizan técnicas de

clustering de recursos web valorados y descubrimiento de reglas de asociación interesantes mediante algoritmos genéticos para optimización de minería de datos [13] con el objetivo de clasificar a los estudiantes para predecir su clasificación final basándose en las características extraídas de los ficheros *log*.

En este trabajo se propone la aplicación de técnicas de descubrimiento de subgrupos utilizando algoritmos evolutivos sobre los datos de utilización del sistema de enseñanza a distancia Moodle implantado en la Universidad de Córdoba. Utilizaremos dos algoritmos evolutivos de inducción de reglas desarrollados en el ámbito del descubrimiento de subgrupos que reciben el nombre de SDIGA [7] y SDMGA [6], y que son respectivamente mono-objetivo y multi-objetivo.

El objetivo de la propuesta será encontrar reglas que describan relaciones entre el uso que hacen los alumnos de las distintas actividades presentes en un curso y la calificación final que presentan. Como comprobaremos más adelante, estas reglas pueden ayudar al profesor en la detección de relaciones beneficiosas o perjudiciales entre el uso de recursos docentes basados en web y el aprendizaje de los alumnos.

El trabajo está organizado de la siguiente forma. En la sección 2 se introduce el descubrimiento de subgrupos y una descripción de los algoritmos evolutivos de inducción de reglas utilizados, SDIGA y SDMGA, y otros dos algoritmos clásicos de descubrimiento de subgrupos utilizados en este trabajo, Apriori-SD y CN2-SD. En la sección 3 se describe el problema objeto de estudio en este trabajo, el sistema de enseñanza a distancia Moodle implantado en la Universidad de Córdoba, se muestra el desarrollo experimental y se analizan algunas de las reglas obtenidas. Finalmente se presentan las conclusiones del trabajo.

II. DESCUBRIMIENTO DE SUBGRUPOS: ENFOQUES CLÁSICOS Y ALGORITMOS SDIGA Y SDMGA

En la literatura especializada podemos encontrar diferentes propuestas de algoritmos de extracción de reglas en el ámbito del descubrimiento de subgrupos.

Dos algoritmos clásicos son adaptaciones de los conocidos algoritmos Apriori (extracción de reglas de asociación) y CN2 (extracción de bases de reglas para clasificación), y denominados Apriori-SD y CN2-SD respectivamente. SDIGA [7] es un algoritmo evolutivo de inducción de reglas difusas que utiliza reglas lingüísticas como lenguaje de descripción para la especificación de los subgrupos y adaptaciones de las medidas utilizadas en los algoritmos de inducción de reglas de asociación como medidas de calidad para la tarea de descubrimiento de subgrupos (confianza y soporte). SDMGA es una versión multiobjetivo del anterior.

En las siguientes subsecciones describimos brevemente los algoritmos clásicos de descubrimiento de subgrupos, la extracción de reglas con algoritmos evolutivos, y los dos modelos evolutivos SDIGA y SDMGA.

A. Algoritmos clásicos de descubrimiento de subgrupos

A continuación describimos brevemente los algoritmos Apriori-SD y CN2-SD:

- Apriori-SD [10] utiliza una variante de la precisión relativa con pesos para las reglas inducidas, y para la evaluación utiliza el soporte y la relevancia de cada regla individual. Es una modificación del algoritmo Apriori-C [3], que propone la integración de clasificación y minería de reglas de asociación adaptando para ello el conocido algoritmo Apriori [2].
- El algoritmo CN2-SD [11] induce subgrupos en forma de reglas utilizando como medida de calidad la relación entre verdaderos y falsos positivos. Para la selección de reglas utiliza como Apriori-SD una variante de la precisión relativa con pesos.

B. Inducción evolutiva de reglas

Se han desarrollado múltiples propuestas de Algoritmos Genéticos (AGs) para la extracción de reglas de distintos tipos: clasificación, asociación o dependencias funcionales. El aspecto más determinante de cualquier AG de inducción de reglas es el esquema de codificación utilizado. En este aspecto, las distintas propuestas en la bibliografía especializada se agrupan en torno a dos enfoques [5]:

- El enfoque "Cromosoma = Regla", en el que cada individuo codifica una única regla.
- El enfoque "Cromosoma = Base de Reglas", o enfoque *Pittsburgh*, en el que cada individuo codifica un conjunto de reglas.

Dentro del enfoque "Cromosoma = Regla" existen tres propuestas genéricas, el enfoque *Michigan*, el enfoque *IRL* (Iterative Rule Learning) y el enfoque *cooperativo-competitivo*. En [5] se puede encontrar una descripción completa de distintas propuestas bajo estos enfoques para la inducción evolutiva de reglas difusas.

En procesos de descubrimiento de reglas de descripción de subgrupos es más adecuado el enfoque "Cromosoma = Regla" ya que el objetivo es encontrar un conjunto reducido de reglas en las que la calidad de cada regla se evalúa de forma independiente a la del resto. A continuación se describen dos propuestas de algoritmos evolutivos para descubrimiento de subgrupos, centradas en dicho enfoque.

C. Algoritmo evolutivo de inducción de reglas SDIGA

El algoritmo SDIGA (Subgroup Discovery Iterative Genetic Algorithm) es un modelo evolutivo para la extracción de reglas difusas para la tarea de descubrimiento de subgrupos. Este algoritmo está descrito en detalle en [7], presentándose aquí de forma breve sus principales características.

En la tarea de descubrimiento de subgrupos hay un conjunto de variables descriptivas y una sola variable objetivo que describe los subgrupos. Como el objetivo es obtener un conjunto de reglas que describan subgrupos para todos los valores de la variable objetivo, el AG de esta propuesta descubre reglas difusas donde el consecuente está prefijado a uno de los posibles valores de esta variable objetivo. Así, cada ejecución de SDIGA obtiene un conjunto de reglas para el valor especificado de la variable objetivo, y habrá que ejecutar el algoritmo para cada uno de los valores posibles de la variable objetivo.

Cada solución candidata se codifica de acuerdo con el enfoque "*Cromosoma = Regla"*, representando sólo el antecedente de la regla (puesto que todos los individuos de la población están asociados con el mismo valor de la variable objetivo). El antecedente está formado por una conjunción de parejas variable-valor. La información relativa a cada regla está almacenada en un cromosoma de longitud fija para el que se utiliza un modelo de representación entera (la i-ésima posición indica el valor adoptado por la i-ésima variable).

El núcleo de SDIGA es un AG que utiliza una etapa de post-procesamiento basada en una búsqueda local (un procedimiento de ascensión de colinas). El AG híbrido extrae una regla difusa sencilla e interpretable optimizando el soporte y la confianza. La etapa de post-procesamiento se aplica para incrementar la generalidad de las reglas extraídas.

Este AG híbrido se incluye en un proceso iterativo para la extracción de un conjunto de reglas que describen diferentes zonas (no necesariamente disjuntas) del espacio de búsqueda. Se obtiene un conjunto de soluciones generadas en ejecuciones sucesivas del AG que corresponden con un mismo valor de la variable objetivo. Esto se consigue marcando las instancias cubiertas por la regla obtenida, de forma que se evita que una regla obtenida después cubra exactamente los mismos ejemplos que otra previamente extraída. De esta forma se obtienen reglas difusas diferentes, aunque pueden estar solapadas.

El modelo utiliza reglas difusas, que ofrecen una mejor interpretabilidad de las reglas extraídas debido al uso de una representación del conocimiento cercana al experto, permitiendo además la utilización de variables numéricas sin una discretización previa. Los conjuntos difusos correspondientes a las etiquetas lingüísticas se definen mediante las correspondientes funciones de pertenencia que pueden ser especificadas por el usuario o definidas mediante una partición uniforme si no se dispone de conocimiento de un experto (utilizando particiones uniformes con funciones de pertenencia triangulares).

La función de evaluación del AG combina, según la siguiente expresión, tres factores: la confianza, el soporte y el grado de interés de la regla:

$$fitness(c) = \frac{\omega_1 \cdot Confianza(c) + \omega_2 \cdot Soporte(c) + \omega_3 \cdot Interés(c)}{\omega_1 + \omega_2 + \omega_3}$$

Estas medidas se calculan de la siguiente forma:

- *Confianza*. Determina la precisión de la regla ya que refleja el grado con el que los ejemplos pertenecientes a la zona del espacio delimitado por el antecedente verifican la información indicada en el consecuente de la regla. Se calcula mediante una adaptación de la expresión de precisión de Quinlan [15] utilizada en la generación de reglas de clasificación difusas [4]: el cociente entre la suma del grado de pertenencia de los ejemplos de la clase a la zona determinada por el antecedente y la suma del grado de pertenencia de todos los ejemplos (independientemente de su clase) a la misma zona.
- *Soporte*. Es una medida del grado de cobertura que la regla ofrece a los ejemplos de la clase. Esta medida pretende fomentar la obtención de reglas diferentes en sucesivas ejecuciones del AG híbrido. Así, para el cálculo del soporte sólo se consideran los ejemplos que no están cubiertos por reglas previamente obtenidas en ejecuciones anteriores del AG. De esta forma, el soporte se define como el cociente entre el número de nuevos ejemplos cubiertos por la

regla y el número de ejemplos que quedaban por cubrir en el conjunto de ejemplos.

- *Interés*. El grado de interés se determina en esta propuesta objetivamente mediante el criterio de interés aportado por Noda y otros [14] en un proceso de modelado de dependencias. En la propuesta se utiliza sólo la parte referente al antecedente para el cálculo del interés, puesto que el consecuente está prefijado.

El objetivo global de la función de evaluación es orientar la búsqueda hacia reglas que maximicen la precisión y la medida de interés, minimizando el número de ejemplos negativos y no cubiertos.

El AG utiliza un modelo de reproducción de estado estacionario modificado que intenta obtener un equilibrio entre convergencia y diversidad como se explicará posteriormente, en el que la población original sólo se modifica mediante la sustitución de los peores individuos por los individuos resultantes del cruce y la mutación. La recombinación se lleva a cabo mediante un operador de cruce en dos puntos y un operador de mutación aleatorio sesgado. El cruce se aplica sobre los dos mejores individuos de la población, obteniendo dos nuevos individuos, que sustituirán a los dos peores individuos de la población. Se utiliza un operador de mutación aleatoria sesgado que se aplica a un gen del cromosoma elegido de acuerdo con la probabilidad de mutación. Este operador puede ser aplicado de dos formas distintas, con una probabilidad de 0.5 cada una: en la primera forma, la mutación provoca la eliminación de la variable a la que corresponde el gen, y en la segunda se asigna de forma aleatoria un valor de entre los que puede tomar la variable. La aplicación de este operador sirve para potenciar la diversidad de la población.

Por último, se le aplica a la regla obtenida mediante el AG una etapa de post-procesamiento, que mejora la regla obtenida mediante un proceso de ascensión de colinas, modificando la regla para incrementar el grado de soporte. Para ello, se selecciona en cada iteración una variable que, al ser eliminada, se incrementa el soporte de la regla resultante, obteniendo de esta forma reglas más generales. Finalmente, la regla optimizada sustituirá a la regla original sólo si supera la confianza mínima establecida.

D. Algoritmo evolutivo multiobjetivo SDMGA

Esta propuesta de algoritmo evolutivo multiobjetivo [6] extrae reglas utilizando el mismo esquema de representación que el algoritmo anterior, y cuyo objetivo es extraer para cada valor de la variable objetivo un número variable de reglas diferentes que expresen información sobre los ejemplos del conjunto de partida. El algoritmo permite generar reglas difusas y/o nítidas, para problemas con variables continuas y/o nominales.

El AG multiobjetivo sigue el enfoque SPEA2 [22], y por tanto aplica los conceptos de elitismo en la selección de reglas (utilizando una población secundaria) y búsqueda de soluciones óptimas en el frente de Pareto (se ordena a los individuos de la población de acuerdo a si cada individuo es o no dominado usando el concepto de óptimo de Pareto). Cualquier AG multiobjetivo debe diseñarse para lograr dos propósitos: lograr buenas aproximaciones al frente de Pareto y mantener la diversidad de las soluciones, con el objetivo de muestrear adecuadamente el espacio de soluciones y no converger a una solución única o a una sección acotada del frente. Para preservar la diversidad a nivel fenotípico el algoritmo utiliza una técnica de nichos que considera la cercanía en valores de los objetivos. La Tabla I muestra el esquema de funcionamiento del modelo propuesto.

TABLA I
ESQUEMA DEL ALGORITMO EVOLUTIVO PROPUESTO

Inicialización: Generar la población inicial P_0 y crear una población elite vacía $P'_0 = Ø$.
Repetir
  Asignación de fitness: Calcular el fitness de los individuos de la población P_t y de la población elite P'_t de forma conjunta.
  Selección de entorno: Copiar todos los individuos no dominados de la población P_t y la población elite P'_t en la población elite P'_{t+1}. Si el tamaño de P'_{t+1} sobrepasa el número de elementos a guardar, reducir P'_{t+1} mediante el operador de truncado; en otro caso, si el tamaño de P'_{t+1} es inferior al número de elementos, rellenarlo con individuos dominados de P_t y de P'_t.
  Esquema de reproducción: Realizar selección por torneo binario con reemplazo sobre la población elite P'_{t+1} aplicando después operadores de cruce y mutación. El resultado es la población P_{t+1}.
Mientras no se verifique la condición de parada.
Devolver los individuos no dominados de la población elite P'_{t+1}.

A continuación se describen brevemente los objetivos, la selección de entorno y el esquema de reproducción.

En el proceso de descubrimiento de subgrupos se intentan conseguir reglas con capacidad descriptiva alta, comprensibles e interesantes. En esta propuesta multiobjetivo se han definido tres objetivos: confianza, soporte e interés. La confianza y el interés se definen de la misma forma que en el anterior algoritmo SDIGA, pero se utiliza una definición distinta del soporte porque se tienen conjuntos de reglas distintas sin necesidad de aplicar un esquema iterativo. En este caso, el soporte se calcula como el cociente entre el número de ejemplos de la clase descritos por la regla y el número total de ejemplos de la clase.

Con respecto a la selección de entorno, se establece un tamaño fijo para la población elite, de forma que es necesario definir una función de truncado y otra de rellenado. La función de truncado permite eliminar soluciones no dominadas de la población elite si excede el tamaño definido. Para ello se utiliza un esquema de nichos definido en torno a la densidad medida según el k-ésimo vecino más cercano, en el que, en un proceso iterativo, en cada iteración se elimina de la población elite aquel individuo que está más cerca de otros respecto a los valores de los objetivos. La función de rellenado permite añadir elementos dominados tanto de la población como de la población elite hasta completar el tamaño de la misma (ordenando los individuos según su valor de fitness).

El algoritmo utiliza el siguiente esquema de reproducción:

- Se une la población original con la población elite y se obtienen los elementos no dominados de la unión de ambas poblaciones.
- Se aplica un esquema de selección por torneo binario sobre los individuos no dominados.
- A la población resultante, se le aplica recombinación a través del operador de cruce en dos puntos y un operador de mutación uniforme sesgado con el que la mitad de las mutaciones realizadas eliminan la variable correspondiente, para incrementar la generalidad de las reglas.

III. CASO DE ESTUDIO DE E-LEARNING: SISTEMA DE ENSEÑANZA A DISTANCIA MOODLE IMPLANTADO EN LA UNIVERSIDAD DE CÓRDOBA

En esta sección presentamos el caso de estudio en el ámbito de e-learning. El estudio lo presentamos en 3 subsecciones, en la primera describimos el problema, en la segunda mostramos los resultados experimentales de la aplicación de los diferentes algoritmos de descubrimiento de subgrupos sobre los datos disponibles, y en la tercera sección presentamos un análisis de algunas de las reglas obtenidas desde la perspectiva de los cursos de enseñanza a distancia utilizados.

A. Descripción del problema

Como ya se ha comentado anteriormente, se van a utilizar los datos de utilización de los alumnos del sistema Moodle, que es uno de los sistemas de enseñanza basada en web más utilizados [8].

El objetivo que se pretende conseguir con la aplicación de descubrimiento de subgrupos es analizar qué relación puede tener la realización de actividades complementarias de una asignatura realizadas en un sistema de enseñanza a distancia con la nota final obtenida por los estudiantes en dicha asignatura. La calificación final se utiliza como variable a caracterizar, utilizando las diferentes calificaciones para dividir los datos en clases y codificados como valores del consecuente de las reglas.

Se van a utilizar diferentes algoritmos de descubrimiento de subgrupos para poder comparar los resultados obtenidos y poder demostrar qué tipo de algoritmo descubre la información de mayor calidad/utilidad para el profesor del curso. La información descubierta en forma de reglas se mostrará al profesor para que la pueda utilizar para tomar decisiones sobre si, o bien, fomentar aun más el uso de determinado tipo de actividades ya que ha comprobado que están relacionadas con la obtención de una alta puntuación, o por el contrario, eliminar determinadas actividades al estar más relacionadas con bajas puntuaciones.

TABLA II
ATRIBUTOS UTILIZADOS PARA CADA ALUMNO

Nombre	**Descripción**
Course	Número identificador del curso
n_assigment	Nº de trabajos realizados
n_assigment_a	Nº de trabajos aprobados
n_assigment_s	Nº de trabajos suspendidos
n_quiz	Nº de cuestionarios realizados
n_quiz_a	Nº de cuestionarios aprobados
n_quiz_s	Nº de cuestionarios suspendidos
n_messages	Nº de mensajes enviados al chat
n_messages_ap	Nº de mensajes enviados por el alumno al profesor del curso
n_posts	Nº de mensajes enviados al foro
n_read	Nº de mensajes leídos del foro

En nuestro caso, se dispone de la información correspondiente a 192 cursos, correspondientes a distintas asignaturas de las titulaciones que se imparten en la Universidad de Córdoba. De entre ellos, se han seleccionado los 5 cursos que han hecho un mayor uso de las principales actividades y recursos, con un número total de 300 alumnos.

La base de datos de Moodle dispone de una gran cantidad de información muy detallada almacenada en multitud de tablas dentro de una base de datos relacional. Por esta razón, ha sido necesario realizar una primera etapa de preprocesado de la información. En primer lugar, se ha creado una nueva tabla resumen, Tabla II, con la información que se ha considerado más importante para nuestro objetivo. Posteriormente se procedido a la transformación al formato requerido por las implementaciones de los algoritmos que se van a utilizar. Esta tabla almacena un resumen por fila de todas las actividades realizadas por cada alumno en el curso, así como la nota final obtenida por cada alumno en dicha asignatura, discretizada con los

valores categóricos tradicionales de sobresaliente, notable, aprobado y suspenso. Finalmente, se ha exportado toda la información de la tabla resumen a un fichero tipo texto con formato de datos KEEL [1] debido a que los algoritmos de descubrimiento de subgrupos se encuentran implementados dentro de esta plataforma.

B. Resultados experimentales de la aplicación de los algoritmos de descubrimiento de subgrupos

Para la realización de las pruebas se han utilizado cuatro algoritmos distintos: los algoritmos clásicos Apriori-SD y CN2-SD y los algoritmos evolutivos SDIGA y SDMGA. Se han realizado varias ejecuciones de los diferentes algoritmos para obtener los valores medios de las medidas de evaluación de la calidad de las reglas.

En los dos algoritmos clásicos se han realizado 5 ejecuciones distintas variando uno de sus parámetros. En el caso del Apriori-SD se ha variado el soporte mínimo (0.03, 0.1, 0.2, 0.3, y 0.4) para diferentes valores de confianza mínima (0.6, 0.7, 0.8, 0.9). En el caso del CN2-SD se ha variado el tamaño de la estrella (1, 2, 3, 4, 5) para diferentes valores del parámetro γ (0.9, 0.7, 0.5 y aditivo).

En los algoritmos evolutivos se han realizado 5 ejecuciones para cada una de las 4 clases del atributo objetivo nota (sobresaliente, notable, aprobado y suspenso) para diferentes valores de confianza mínima (0.6, 0.7, 0.8, 0.9). Además, para el algoritmo evolutivo multiobjetivo hemos utilizado una población elite de tamaño 5. Para ambos algoritmos los siguientes parámetros son comunes:

- Tamaño de población: 100.
- Número de evaluaciones: 10,000.
- Probabilidad de cruce: 0.6.
- Probabilidad de mutación: 0.01.
- Etiquetas lingüísticas para las variables continuas: 5 (muy alto, alto, medio, bajo y muy bajo).

En la Tabla III se muestran los resultados obtenidos en términos de valores promedio de: el número de reglas total descubierto, el número de atributos en el antecedente de las reglas y los valores para las medidas de soporte, cobertura, precisión y relevancia de las reglas.

Analizando el número de reglas y número de variables podemos realizar las siguientes observaciones:

- Los algoritmos evolutivos descubren un menor número de reglas y en cambio CN2-SD es el que mayor número de reglas descubre.
- Con respecto al número de atributos Apriori-SD y SDMGA son los que obtienen un menor número de atributos, nuevamente CN2-SD obtiene un mayor número de atributos.
- Desde el punto de vista de nuestro problema interesa tener una cantidad no muy elevada de reglas y con pocos atributos para facilitar la comprensibilidad de dichas reglas al profesor. Por lo tanto el algoritmo CN2-SD no es el más apropiado.

TABLA III
RESULTADOS DE LOS ALGORITMOS

Algoritmo	Número de Reglas	Número de Atributos	Soporte	Cobertura	Confianza	Relevancia
SDIGA CfMin 0,6	8,4	2,4306	0,7260	0,2921	0,8284	20,2476
SDIGA CfMin 0,7	6,8	2,6143	0,6253	0,1816	0,8732	21,8973
SDIGA CfMin 0,8	4,2	2,9300	0,4137	0,1167	0,9573	19,6746
SDIGA CfMin 0,9	3,0	2,8000	0,0226	0,0075	1,0000	5,6190
SDMGA CfMin 0,6	7,8	1,9484	0,9288	0,3829	0,6755	8,7596
SDMGA CfMin 0,7	5,8	1,5486	0,9219	0,5267	0,5769	11,1724
SDMGA CfMin 0,8	5,4	1,4943	0,9493	0,5904	0,5162	12,9659
SDMGA CfMin 0,9	6,0	1,9229	0,9164	0,4874	0,6111	10,8591
Apriori-SD CfMin 0,6	9,8	1,0400	0,5924	0,6001	0,6157	27,3901
Apriori-SD CfMin 0,7	10,4	1,3238	0,5513	0,6232	0,6301	31,4304
Apriori-SD CfMin 0,8	5,0	0,8294	0,3734	0,1451	0,3842	21,2968
Apriori-SD CfMin 0,9	4,6	1,1692	0,2089	0,1164	0,3787	21,0734
CN2-SD (γ=0.5)	15,6	5,6406	0,9342	0,4461	0,7143	45,8554
CN2-SD (γ=0.7)	18,4	5,6857	0,9876	0,4600	0,7177	47,0058
CN2-SD (γ=0.9)	25,2	5,7177	0,9890	0,4703	0,7184	47,2862
CN2-SD (add)	31,5	5,7741	1,0000	0,5038	0,7129	54,7134

Con respecto a las medidas utilizadas se puede observar que:

- Para el soporte, es precisamente el algoritmo CN2-SD el que presenta mayores valores muy cercanos a 1, indicando que las reglas cumplen casi el 100 % de los estudiantes. Esto es debido a la naturaleza del propio algoritmo que añade muchos atributos en las reglas hasta poder cubrir a todas las instancias de datos. Valores de soporte altos muy cercanos al anterior muestra también el algoritmo multiobjetivo, con la ventaja de utilizar sólo la mitad de atributos en las reglas. Apriori-SD y SDIGA son los que presentan valores más bajos de soporte.
- La medida de cobertura es al igual que el soporte una medida de la generalidad de la regla, midiendo en este caso el número de alumnos que cumplen el antecedente de la regla. En este caso, es el algoritmo Apriori-SD el que presenta los valores más altos, muy seguido del algoritmo SDMGA y CN2-SD, y por último SDIGA.
- La medida confianza o exactitud de la regla, indica el número de estudiantes cubiertos por el antecedente de la regla y que corresponde a la clase asociada a la misma (en términos de clasificación hablaríamos de correctamente clasificados por la regla). El algoritmo que presenta unos valores más altos muy cercanos al 100% es SDIGA, seguido de CN2-SD, SDMGA y Apriori-SD.
- La medida de relevancia es una medida cuantitativa de la relevancia y del interés de la regla. El algoritmo CN2-SD es el que presenta valores de relevancia más altos, seguidos del algoritmo Apriori-SD, después SDIGA y por último el multiobjetivo SDMGA. Esto se debe a que los algoritmos evolutivos aquí presentados no utilizan la relevancia como medida de calidad durante el proceso de minería de datos.

Con respecto a los valores de estas cuatro medidas, el mejor algoritmo sería aquel que presentara los valores más altos en todas ellas. Como se ha podido comprobar no hay un único algoritmo que presente los valores más altos simultáneamente en las cuatro medidas utilizadas, por lo que no se puede elegir un mejor algoritmo.

C. Análisis de la comprensibilidad de las reglas obtenidas por los algoritmos evolutivos

Con respecto a la comprensibilidad de las reglas para su utilización directa en la toma de decisiones del profesor del curso, los algoritmos evolutivos presentan las reglas con una mayor interpretabilidad debido a que utilizan el formato para los atributos de ETIQUETA = VALOR, donde el valor en lugar de números, son etiquetas lingüísticas proporcionadas por el experto de más fácil interpretación para el profesor. El algoritmo Apriori-SD, utiliza también el mismo formato de regla pero los valores en lugar de etiquetas son valores numéricos, por lo que son algo menos interpretables. Por otro lado, el algoritmo CN2-SD utiliza valores numéricos y los operadores igual, mayor que, menor que y distinto, de forma que las reglas obtenidas son las más difíciles de interpretar.

A continuación se van a mostrar algunos ejemplos de reglas descubiertas con los algoritmos evolutivos y se va a analizar su significado aplicado para una posible mejora del curso.

SI course = 110 Y n_assignment = Alto Y n_posts = Alto
ENTONCES nota = Notable
Soporte: 0.70454544 Confianza: 0.7230769

Para el curso 110, los alumnos que han realizado muchos trabajos y han enviado muchos mensajes al foro han obtenido una nota alta. El profesor de este curso debe seguir fomentando este tipo de actividades ya que ha podido comprobar su efectividad en la nota final obtenidas por los alumnos que las realizan.

SI course = 88 Y n_messages = Muy Alto
ENTONCES nota = Suspenso
Soporte: 0.19298245 Confianza: 0.9444444

Para el curso 88, los alumnos que han enviado muchos mensajes al chat, luego han suspendido. El profesor de este curso puede eliminar el chat debido a que no ha aportado beneficio a los alumnos, al contrario ha podido ser una fuente de distracción.

SI n_read = Muy Bajo
ENTONCES nota = Suspenso
Soporte: 0.73240235 Confianza: 0.6103379

Para cualquier curso si el número de mensajes leídos del foro es muy bajo entonces la nota final obtenida es de suspenso. El profesor a partir de esta información puede prestar más atención a estos alumnos, ya que van a tender a suspender y podría intentar motivarlos todavía a tiempo para poder aprobar la asignatura.

SI n_read = Alto Y n_messages_ap = Muy Bajo
ENTONCES nota = Suspenso
Soporte: 0.11759777 Confianza: 0.7500000

Para cualquier curso si el número de mensajes leídos del foro es alto pero el número de mensajes enviados al profesor es bajo, entonces la nota obtenida es de suspenso. Esta regla puede parecer que contradice a la regla anterior, pero aporta información sobre otro grupo de alumnos distinto menos numeroso y que también tiende a suspender. Al igual que antes el profesor puede prestar más

atención a estos alumnos e intentar motivarlos a tiempo para poder aprobar.

Finalmente, hay que indicar que también se han encontrado reglas que o bien no aportan ninguna información nueva o bien daban información obvia para el profesor. Por ejemplo, la regla

SI n_quiz_a = Muy baja
ENTONCES nota = Suspenso
Soporte: 0.93296087 Confianza: 0.72661173

indica que si el número de cuestionarios aprobados es muy bajo entonces la nota final obtenida es suspenso. Para el profesor esto es totalmente lógico (el que aprueba los cuestionarios en línea luego aprueba el examen en papel) y no le aporta nada nuevo de cómo mejorar el curso.

IV. CONCLUSIONES

En este trabajo se ha presentado la aplicación de algoritmos evolutivos para la inducción descriptiva de reglas que describen subgrupos en un problema real de extracción de conocimiento en sistemas de e-learning.

Después de realizar pruebas comparando los algoritmos propuestos con otros algoritmos clásicos de descubrimiento de subgrupos, se ha comprobado que los algoritmos evolutivos son muy apropiados para resolver el problema propuesto. Obtienen un número reducido de reglas comprensibles (por su reducido tamaño y el uso de etiquetas lingüísticas) que las hace más interpretables para el profesor, además de obtener unos niveles parecidos en las medidas de evaluación de la calidad de las reglas y el óptimo en dos de los seis objetivos. A partir de las reglas obtenidas el profesor del curso puede tomar decisiones sobre las actividades del curso y sus alumnos para mejorar su rendimiento.

AGRADECIMIENTOS

Este trabajo ha sido financiado por el MCYT a través de los proyectos TIN2005-08386-C05-01, TIN2005-08386-C05-02 y TIN2005-08386-C05-03.

REFERENCIAS

[1] Alcalá, J. del Jesús, M.J., Garrell, J.M., Herrera, F., Hervás, C., Sánchez, L., Proyecto KEEL: Desarrollo de una Herramienta para el Análisis e Implementación de Algoritmos de Extracción de Conocimiento Evolutivos. Tendencias de la Minería de Datos en España. Eds. Giraldés, J., Riquelme, J.C., Aguilar, J.S., pp. 413-423. 2004.

[2] Agrawal, R., Imielinski, T., Swami, A., Mining association rules between sets of items in large databases. ACM SIGMOD Conference on Management of Data, pp. 207-216. 1993.

[3] Bing L., Wynne H., Yiming M., Integrating Classification and Association Rule Mining. Fourth International Conference on Knowledge Discovery and Data Mining, pp. 80-86, 1998.

[4] Cordón, O., del Jesus, M.J. and Herrera, F., Genetic Learning of Fuzzy Rule-based Classification Systems Co-operating with Fuzzy Reasoning Methods. International Journal of Intelligent Systems, Vol. 13 No. 10/11. pp. 1025-1053. 1998.

[5] Cordón, O., Herrera, F., Hoffmann,, F. and Magdalena, L., Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases. World Scientific. 2001.

[6] del Jesus, M.J., González, P., and Herrera, F., Inducción evolutiva multiobjetivo de reglas de descripción de subgrupos en un problema de marketing. IV Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados, pp. 661-669, Granada. 2005.

[7] del Jesus, M.J., González, P., Herrera, F. and Mesonero, M., Evolutionary fuzzy rule induction process for subgroup discovery: a case study in marketing. IEEE Trans. Fuzzy Systems, In press.

[8] Flate, M., Online education and learning management systems. Global e-learning in a Scandinavian perspective. NKI Gorlaget. Oslo. 2003.

[9] Freyberger, J., Heffernan, N.T., Ruiz, C., Using Association Rules to Guide a Search for Best Fitting Transfer Models of Student Learning. Int. Conf. on Intelligent Tutoring Systems. pp 1-10. 2004.

[10] Kavsek, B., Lavrac, N., Jovanoski, V., APRIORI-SD: Adapting association rule learning to subgroup discovery. Advances in Intelligent Data Analysis V. p. 230-241. 2003.

[11] Lavrac, N., Kavsec, B., Flach, P. and Todorovski, L., Subgroup discovery with CN2-SD. Journal of Machine Learning Research, Vol. 5 pp. 153-188. 2004.

[12] Minaei-Bidgoli, B., Punch, W.F., Predicting student performance: an application of data mining methods with the educational web-based system LON-CAPA. IEEE Frontiers in Education. pp 1-6. 2003.

[13] Minaei-Bidgoli, B., Punch, W.F., Using Genetic Algorithms for Data Mining Optimization in an Educational Web-Based System. Genetic and Evolutionary Computation Conference GECCO. pp. 2252-2263. 2003.

[14] Noda, E., Freitas, A.A., Lopes, H.S., Discovering Interesting Prediction Rules with a Genetic Algorithm. Congress on Evolutionary Computation. pp. 1322-1329. 1999.

[15] Quinlan, J.R., Generating Production Rules from Decision Trees. International Joint Conference on Artificial Intelligence. pp. 304-307. Milan, Italy. 1987.

[16] Romero, C., Ventura, S., de Bra, P., Knowledge Discovery with Genetic Programming for Providing Feedback to Courseware Author. User Modeling and User-Adapted Interaction. Vol. 14. No. 5. pp. 425-464. 2004.

[17] Romero, C., Ventura, S., Educational data mining: a survey from 1995 to 2005. Expert Systems with Applications. Vol. 33. num 1. September 2006. Elsevier. pp. 1-12.

[18] Wang, F., On Analysis and Modeling of Student Browsing Behavior in Web-Based Asynchronous Learning Environments. Int. Conf. on Web-based Learning. pp. 69-80. 2002.

[19] Yu, P., Own, C., Lin, L., On the Learning Behavior Analysis of Web Based Interactive Environmnent. pp. 1-8. ICCE. 2001.

[20] Zaïane, O.R., Web Usage Mining for a Better Web-Based Learning Environment. Conference on Advanced Technology for Education. Alberta, pp 60-64. 2001.

[21] Zaïane, O.R., Building a Recommender Agent for e-Learning Systems. International Conference on Computers in Education. New Zealand. pp 55-59. 2002.

[22] Zitzler, E., Laumanns, M., Thiele, L., SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimisation. Evolutionary methods for design, optimisation and control, CIMNE. pp. 95-100. 2002.