Integración de modelos de agrupamiento y reglas de asociación obtenidos de múltiples fuentes de datos

Daymi Morales Vega, Diana Martín Rodríguez, Ingrid Wilford Rivera y Alejandro Rosete Suárez

Instituto Superior Politécnico "José Antonio Echeverría", La Habana, Cuba

{dmorales, dmartin, iwilford, rosete}@ceis.cujae.edu.cu

Resumen. Una alternativa posible para descubrir conocimiento sobre bases de datos distribuidas, usando técnicas de Minería de Datos, es rehusar los modelos de minería de datos locales obtenidos en cada base de datos e integrarlos para obtener patrones globales. Este proceso debe realizarse sin acceder a los datos directamente. Este trabajo se centra en la propuesta de dos métodos para la integración de modelos de Minería de Datos: Modelos de Reglas de Asociación y Agrupamiento, específicamente para reglas de asociación obtenidas usando soporte y confianza como medidas de calidad y agrupamientos basados en centroides. Estos modelos fueron obtenidos al analizar múltiples conjuntos de datos homogéneos. El estudio experimental muestra que se obtuvieron modelos globales de calidad en un tiempo razonable cuando se aumentan la cantidad de patrones locales a integrar.

Palabras clave. Integración, modelos de minería de datos, reglas de asociación, agrupamiento, Patrones.

Integration of Association Rules and Clustering Models Obtained from Multiple Data Sources

Abstract. One possible way to discover knowledge over distributed data sources, using Data Mining techniques, is to reuse the models of local mining found in each data source and look for patterns globally valid. This process can be done without accessing the data directly. This paper focuses on the proposal of two methods for integrating data mining models: Association Rules and Clustering Models, specifically rules were obtained using support and confidence as measures of quality and clustering based on centroids. It was necessary to use metaheuristics algorithms to find a global model that is as close as possible to the local models. These models were obtained using homogeneous data sources. The experimental study showed that the proposed methods obtain global models of quality in a reasonable time when increasing the amount of local patterns to integrate.

Keywords. Integration, data mining models, association rules, clustering, patterns.

1 Introducción

El uso de Internet y de las nuevas tecnologías para la comunicación ha provocado que actualmente existan muchos sistemas de información con datos distribuidos entre varios nodos ubicados en sitios distantes. En muchos de estos sistemas, no es posible o factible centralizar todos los datos distribuidos en un único repositorio con el propósito de realizar tareas de Minería de Datos (MD), debido, por ejemplo, a restricciones económicas, técnicas o legales. Adicionalmente, existen bases de datos cuyo crecimiento es vertiginoso y el tamaño de las mismas hace que las técnicas clásicas de MD no funcionen correctamente. En estos casos se puede pensar en realizar divisiones o particiones de las mismas en múltiples conjuntos. En estos entornos se hace necesario aplicar otras técnicas de MD que sean capaces de trabajar sobre múltiples fuentes o aplicar técnicas de manera independiente en cada una de las particiones para luego integrar los resultados obtenidos individualmente.

Por otro lado, entre las tareas clásicas de MD más utilizadas se encuentra el agrupamiento o c*lustering* y la obtención de Reglas de Asociación. Es por eso que extender su aplicación a múltiples fuentes de datos, ha resultado de particular interés por muchos autores. Igualmente existen métodos que

proponen la integración de modelos de agrupamiento y de reglas de asociación obtenidos de manera independiente de forma tal que no sea necesario acceder a los datos directamente [2,4-5,7-9,15,17], sin embargo estos métodos presentan ciertas limitaciones. En el caso de los métodos asociados a la tarea de Agrupamiento, algunos de estos no funcionan bien cuando las fuentes de datos son muy grandes [4,7-8,11,17], y otros, como en [5], no tienen en cuenta la calidad de los grupos a integrar ni la cantidad de instancias analizadas en los conjuntos locales que le dieron origen al modelo.

Con respecto a las reglas de asociación, se han encontrado tres propuestas específicas de métodos para la integración de este tipo de modelo generados a partir de los diferentes conjuntos de datos (*dataset*). La primera propuesta presentada por [2], tiene limitaciones importantes pues basa el diseño del método en el esquema específico de base de datos estrella. La segunda, presentada por [15], tiene como principales limitaciones: asumir que cada conjunto de datos analizado contiene similar cantidad de registros y que las reglas de asociación se obtienen considerando el mismo valor de soporte y confianza mínimo, ya que en aplicaciones reales no tienen por qué cumplirse estas restricciones. La tercera propuesta presentada por [9], presenta problemas al utilizar formulaciones matemáticas apropiadas para descubrir patrones excepcionales (tienen altos valores de soporte y confianza pero son descubiertos en pocos *datasets*) y no patrones globales soportados por la mayoría de los *datasets*.

En estos contextos se identifica como un problema a resolver la necesidad de definir métodos factibles y escalables de integración de modelos de MD, resultantes del análisis de múltiples fuentes de datos, mediante técnicas de agrupamiento y de reglas de asociación, sin acceder a los datos originales.

La factibilidad de los métodos se refiere a la obtención de patrones globales que describan todo el conjunto de datos, los cuales pudieran haber sido descubiertos de la centralización de todos los conjuntos de datos locales. Por otro lado, la escalabilidad se refiere al análisis del comportamiento de los métodos propuestos al incrementar la cantidad de patrones (grupos o reglas de asociación) locales a integrar.

En el presente trabajo se describen dos métodos de integración de modelos de minería de datos, uno de estos métodos integra modelos de agrupamiento, para agrupamientos basados en centroides y el otro integra modelos de reglas de asociación, obtenidas usando soporte y confianza como medidas de calidad. Se presentarán también los principales resultados obtenidos al aplicar experimentalmente ambos métodos.

2 Modelo de integración de conocimiento

En este apartado se describe el marco conceptual de las propuestas de métodos de integración de modelos de agrupamiento basados en centroides y de modelos de reglas de asociación, obtenidas usando soporte y confianza como medidas de calidad (Clustering Models Integration Method - *CMIM* y Association Rules Integration Model - *ARIM*). Estos métodos tienen como objetivo la integración de este tipo de modelos obtenidos desde múltiples fuentes de datos. Tanto *CMIM* como *ARIM,* forman parte del Modelo de Integración de Conocimiento [12] (Knowledge Integration Model - *KIM*) para la integración de modelos de MD descubiertos en múltiples fuentes de datos. Este modelo es muy útil para las organizaciones que no pueden compartir sus datos originales pero sí sus modelos de MD.

El Modelo de Integración de Conocimiento (*KIM*) tiene como objetivo fundamental obtener un modelo global de MD, mediante la integración de un conjunto de modelos locales obtenidos de la aplicación de técnicas de MD en varios *datasets*. El *KIM* tiene como restricción la imposibilidad de acceder a los conjuntos de datos fuente, por lo que, el proceso de “Integración” deberá realizarse conociendo únicamente los modelos locales y, asociado a cada uno, una ficha que recoge información útil para la integración. De manera general, las fichas registrarán información resumida referente a los datos fuente (por ejemplo: cantidad de registros analizados), así

como, información concerniente al proceso de minería de datos desarrollado localmente (por ejemplo, umbrales de medidas de patrones como: valor mínimo de soporte y de confianza, entre otros).

Formalmente, el problema de investigación puede ser definido como sigue:

Sea $M= \{m_1, m_2,\ldots, m_n\}$ un conjunto de modelos locales descubiertos de diferentes conjuntos de datos o *datasets*. Cada modelo $m_i \in M$ puede ser definido como un conjunto de patrones $m_i= \{p_1, p_2,\ldots, p_t\}$. Cada patrón $p_j \in m_i$ tiene la forma $p_j = \langle id_j, me_j \rangle$, donde id_j representa las características que identifican el patrón p_j, y me_j representa las medidas de dicho patrón. Si p_j representa un grupo, id_j sería su *centroide*, o *medioide* (en dependencia de cómo se haya obtenido) y me_j sería su cobertura y su precisión. Además, $F= \{f_1, f_2,\ldots, f_n\}$ representa un conjunto de fichas, donde cada ficha $f_i \in F$ contiene información general sobre el modelo local correspondiente ($m_i \in M$) y el *dataset* analizado. Entonces, el problema de la integración de *n* modelos de minería de datos locales $m_i \in M$, usando sus fichas $f_i \in F$, para obtener un modelo de minería de datos global (MG) puede ser formalizado de la siguiente forma:

$$MG = \prod_{i=1}^{n} (m_i, f_i) \mid (m_i \in M) \wedge (f_i \in F) \qquad (1)$$

donde Π representa un operador de integración de *n* modelos de MD locales.

El problema descrito anteriormente constituye un problema de optimización, cuyo objetivo es encontrar un modelo global o integrado que sea lo más semejante posible a los modelos locales. El modelo global de minería de datos estará compuesto por el subconjunto de patrones locales que más se parezcan al conjunto unión de todos los patrones locales y/o por patrones nuevos que se generen a partir de unir dos o más patrones locales. El proceso de seleccionar los patrones que pertenecerán a MG, implica determinar si cada patrón local o nuevo pertenece o no al modelo global, por tanto la cantidad de posibles combinaciones es 2^{t+s} donde *t* representa la cantidad total de patrones locales y *s* es la cantidad de patrones nuevos que se generen. En este caso, el espacio de soluciones

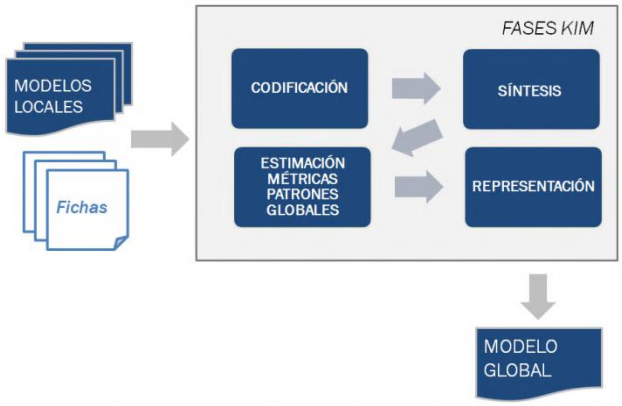

Fig. 1. Fases del Modelo de Integración de Conocimiento (KIM)

crece de manera exponencial cuando aumentan los patrones de los modelos locales (*t*) que se deben integrar, por tanto se hace necesario utilizar algoritmos no determinísticos para resolver el problema en un tiempo polinomial.

El modelo *KIM* plantea la integración de modelos en cuatro fases (Figura 1). Los métodos *CMIM* y *ARIM* implementan cada una de estas fases para lograr la obtención de un modelo global.

La fase de codificación consiste en cargar los modelos locales generados de manera independiente, representados en algún formato estándar, a una estructura de datos acorde al método de integración propuesto. Básicamente esta es una fase de transformación para facilitar la interacción con los usuarios del método y para crear las bases para las posteriores fases. La fase de Síntesis es la de mayor procesamiento ya que obtiene un modelo global a partir de seleccionar un subconjunto de patrones de los modelos locales recibidos. Para el proceso de selección del subconjunto de patrones se aplican los algoritmos metaheurísticos. Al culminar la fase de Síntesis, se obtienen patrones en el modelo global carentes de medidas que ilustren su calidad, para solucionar esto, se propone la fase de Estimación de Métricas donde se asignan valores estimados de métricas a los patrones globales. Estas métricas son cobertura y precisión para el caso de los modelos de agrupamiento y soporte y confianza para los

modelos de reglas de asociación. La fase final de Obtención del MG es muy similar a la primera ya que representa el modelo global en un formato de estándar que es recibido por los usuarios del método.

A continuación se describe con mayor nivel de detalle las fases más significativas del método: Síntesis y Estimación de las métricas de los patrones globales.

2.1 Fase de síntesis

El objetivo de esta fase es obtener un Modelo Global Integrado (MG), a partir de sintetizar o integrar los modelos locales recibidos. En esta fase se debe encontrar cual es la mejor combinación entre los patrones que componen los modelos locales de manera tal que se obtenga un conjunto de patrones (Modelo Global) que represente, de la mejor manera posible, a todos los patrones de los modelos locales.

Es decir, se tienen un conjunto de modelos locales y cada uno de estos contiene a su vez, un conjunto de patrones que lo describen. Luego, un MG estará compuesto por un subconjunto de patrones (pertenecientes a algún modelo local) y/o un conjunto de patrones nuevos que son generados a partir de combinar dos o más patrones de los modelos locales.

Como se explicó anteriormente, el problema planteado se puede clasificar como un problema de optimización combinatoria. A continuación se describirá el mismo según las características de un problema de optimización: representación de un estado, operadores, el dominio de las variables, el espacio de solución y la función objetivo.

Representación de un estado y Dominio: La representación de un estado es un vector, donde cada posición (0, 1, …, *t*+*s*) representa un patrón. Los valores en las posiciones del vector serán 1 si se encuentra el patrón en el MG o 0 en caso contrario, por tanto estos serán los valores del dominio (0 ó 1). El tamaño del vector es *t* + *s*, donde *t* se define como la suma total de los patrones de cada uno de los modelos locales, y *s* es la cantidad de patrones nuevos que se generan a partir de combinar dos o más patrones.

Operadores: Para la generación de estados candidatos, se proponen dos operadores: operador de mutación y operador de agrupamiento. En el primer caso se seleccionará un subconjunto de todos los patrones contenidos en los modelos locales para constituir el MG. En el segundo caso se obtendrán patrones nuevos resultantes de haber combinado dos o más patrones de los modelos locales y serán agregados al Modelo Global.

Espacio de solución: Teniendo en cuenta que las posibles soluciones están en dependencia de la inclusión o no de un patrón determinado y de la cantidad de patrones total (*t*+*s*) que incluye a los patrones locales y a los nuevos que se generen; el espacio de solución es: 2^{t+s}.

Función objetivo: Permite evaluar cada una de las soluciones candidatas (S_l). Donde los valores obtenidos están en el rango de 0 a $\frac{|\mathrm{MG}_\mathrm{l}'|-1}{|\mathrm{MG}_\mathrm{l}'|}$ siendo $|\mathrm{MG}_\mathrm{l}'|$ la cantidad de patrones presentes en el Modelo Global. El valor de la función objetivo representa la distancia entre el Modelo Global y un conjunto de modelos locales. En este problema el objetivo es minimizar los valores de la función objetivo, para encontrar la mejor solución.

La Función Objetivo (FO) en el problema enunciado se define como:

$$\mathrm{f}(\mathrm{S}_\mathrm{l}) = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{w}_{\mathrm{m}_\mathrm{i}} * \mathrm{d}_\mathrm{M}\,(\mathrm{MG}_\mathrm{l}', \mathrm{m}_\mathrm{i}') \tag{2}$$

$$w_{m_i} = \frac{|D_i|}{\sum_{i=1}^{n} |D_i|} \tag{3}$$

Donde w_{m_i} (3) es un peso asociado a cada modelo local m_i que se calcula a partir de la cantidad de instancias del conjunto de datos que le dio origen ($|D_i|$) y la cantidad de instancias total asociadas a todos los modelos locales ($\sum_{\mathrm{i}=1}^{\mathrm{n}} |\mathrm{D}_\mathrm{i}|$). La función $\mathrm{d}_\mathrm{M}(\mathrm{MG}_\mathrm{l}', \mathrm{m}_\mathrm{i}')$ devuelve la distancia entre el MG_l candidato cuyos patrones han sido ordenados (MG_l') y un modelo local m_i también ordenado (m_i'). Para ordenar los patrones de los modelos MG_l y m_i se utiliza el operador$\sigma(\sigma(\mathrm{MG}_\mathrm{l}, \mathrm{m}_\mathrm{i}) = (\mathrm{MG}_\mathrm{l}', \mathrm{m}_\mathrm{i}'))$. Este operador ordena (ascendentemente) los patrones de ambos modelos. Para esto se construye una matriz que contiene los valores de similitud entre cada par de patrones. Luego se busca en esta

matriz los pares de patrones más similares, de manera que, el patrón j del modelo $MG_l^{'}$ (MG_lordenado) quedará "alineado" con el patrón j del modelo $m_i^{'}$ ($m_i$ordenado) que sea más similar a él. Si la cantidad de patrones en ambos modelos no es la misma existirán entonces patrones "no alineados" en el modelo de mayor dimensión.

Entonces, la función $d_M(MG_l^{'}, m_i^{'})$ se formaliza como sigue:

$$d_M(MG_l', m_i') = \sum_{j=1}^{MIN(|MG_l'|,|m_i'|)} w_{p_{m_{i_j}}} * d_p\left(p_{MG_{l_j}'}, p_{m_{i_j}'}\right) + c_{m_i} * \sum_{j=|MG_l|+1}^{|m_i|} w_{p_{m_{i_j}}} + c_{MG_l} * \mu_{MG_l} \quad (4)$$

En (4), el primer sumando considera los pares de patrones "alineados"; mientras que, los dos sumandos restantes consideran los patrones "no alineados". Es decir, si el Modelo Global MG_l' tiene menos patrones que el modelo local m_i', entonces c_{m_i} toma valor 1 y c_{MG_l} valor 0, si se cumple lo contrario ($MG_l^{'}$tiene más o igual cantidad de patrones que $m_i^{'}$), entonces c_{m_i} toma valor 0 y c_{MG_l} valor 1. De esta forma, el valor de distancia se ve afectado por el hecho de que en alguno de los modelos existan patrones "no alineados". Los valores que puede tomar esta ecuación están entre 0 y $\frac{|MG_l'|-1}{|MG_l'|}$. Toma valor 0 cuando los modelos están conformados por patrones iguales y tienen la misma cantidad de patrones. Toma valor $\frac{|MG_l'|-1}{|MG_l'|}$ cuando los patrones alineados son totalmente diferentes, el MG_l' tiene más patrones que m_i', y este último está conformado por un solo patrón.

La función $d_p\left(p_{MG_{l_j}^{'}}, p_{m_{i_j}^{'}}\right)$ devuelve la distancia entre el patrón j del modelo $MG_l^{'}$ y el patrón j del modelo $m_i^{'}$ (patrones "alineados"). La distancia entre dos patrones se calcula a partir de dos funciones; en el caso de integración de modelos de agrupamiento se define en función de la distancia entre sus centroides de la siguiente manera:

$$d_p = d_{CC}\left(\rho_{MI_{l_j}'}, \rho_{\mu_{i_j}'}\right) \quad (5)$$

En el caso de la integración de modelos de reglas de asociación se define en función de la distancia entre el antecedente y el consecuente como sigue:

$$d_p = d_{Ant}\left(p_{MG_{l_j}'}, p_{m_{i_j}'}\right) = w_{Ant} * d_{Ant}\left(\rho_{MI_{l_j}'}, \rho_{\mu_{i_j}'}\right) + w_{Con} * d_{Con}\left(\rho_{MI_{l_j}'}, \rho_{\mu_{i_j}'}\right) \quad (6)$$

donde

$$w_{Ant} + w_{Con} = 1 \quad (7)$$

En ambos casos la distancia entre *centroides* o entre antecedentes y consecuentes se define a partir de la distancia entre atributos $d_A()$ y de la cantidad de atributos (x) como sigue:

$$d_{CC}\left(p_{MG_{l_j}'}, p_{m_{i_j}'}\right) = d_{Ant}\left(p_{MG_{l_j}'}, p_{m_{i_j}'}\right) = d_{Con}\left(p_{MG_{l_j}'}, p_{m_{i_j}'}\right) = \frac{\sum_{k=1}^{x} d_A\left(a_{MG_{l_{j_k}}'}, a_{m_{l_{j_k}}'}\right)}{x} \quad (8)$$

$$d_A\left(a_{MG_{l_{j_k}}'}, a_{m_{i_{j_k}}'}\right) = \begin{cases} 0 \text{ si } a_{MG_{l_{j_k}}'} = a_{m_{i_{j_k}}'} \\ 0.5 \text{ si } \left(a_{MG_{l_{j_k}}'} = 0 \text{ y } a_{m_{i_{j_k}}'} \neq 0\right) \text{ o si } \left(a_{MG_{l_{j_k}}'} \neq 0 \text{ y } a_{m_{i_{j_k}}'} = 0\right) \\ 1 \text{ en caso contrario} \end{cases} \quad (9)$$

Por lo que, la distancia entre el valor del atributo k en el patrón j del modelo $MG_l^{'}$ y el valor del atributo k en el patrón j del modelo m_i', tal y como se define en (9), puede tomar los valores 0, 0.5 y 1. Esta función de distancia se expresa en estos términos teniendo en cuenta que los atributos son nominales y que a menor distancia mayor similitud, por tanto se decidió establecer el valor 0 si los valores de los atributos son iguales y 1 en caso contrario. El valor 0.5 se concibe para

representar la ausencia de algún atributo en los patrones correspondientes a las reglas de asociación.

Como se aprecia en (4), cada patrón del modelo local $\mathrm{m}_{\mathrm{i}}^{'}$ tiene asociado un peso $\mathrm{w}_{\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}}}$:

$$\mathrm{w}_{\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}}} = \frac{\mathrm{Q}_{\mathrm{m}_{\mathrm{i}}}(\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}})}{\sum_{\mathrm{j}=1}^{|\mathrm{m}_{\mathrm{i}}|} \mathrm{Q}_{\mathrm{m}_{\mathrm{i}}}(\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}})} \quad (10)$$

La función $\mathrm{Q}_{\mathrm{m}_{\mathrm{i}}}(\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}})$ (11) evalúa la "calidad" de cada patrón en el modelo local y dependerá de la precisión y de la cobertura de dicho patrón en el modelo.

$$\mathrm{Q}_{\mathrm{m}_{\mathrm{i}}}\left(\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}}\right) = w_{pr} * P_r\left(\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}}\right) + w_{co} * C_o\left(\mathrm{p}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}}\right) \quad (11)$$

Los valores de w_{pr} y w_{co} son pesos correspondientes a las medidas precisión y cobertura respectivamente.

Para calcular el coeficiente $\mu_{\mathrm{MG}_{\mathrm{l}}^{'}}$ asociado al Modelo Global, se tendrá en cuenta la cantidad de patrones del $\mathrm{MG}_{\mathrm{l}}'$ que no estén "alineados" que se calculará a partir de la cantidad de patrones del $\mathrm{MG}_{\mathrm{l}}^{'}$ ($|\mathrm{MG}_{\mathrm{l}}^{'}|$) menos la cantidad de patrones del m_{i}' ($|\mathrm{m}_{\mathrm{i}}^{'}|$)

$$\mu_{\mathrm{MG}_{\mathrm{l}}'} = \frac{|\mathrm{MG}_{\mathrm{l}}'| - |\mathrm{m}_{\mathrm{i}}'|}{|\mathrm{MG}_{\mathrm{l}}'|} \quad (12)$$

Para obtener un MG final, se deben realizar varias iteraciones donde se calculará el valor de la FO $f()$ para todos los estados "candidatos" generados. Finalmente será escogido el MG_{l} para el cual la FO obtenga el menor valor. Luego, se deben estimar las medidas de los patrones que hayan sido incluidos en el MG_{l} para lo cual se aplica la fase de Estimación de Métricas.

2.2 Fase de estimación de métricas

Esta fase consiste en estimar los valores de soporte y confianza (para reglas de asociación) o de cobertura y precisión (para centroides) de los patrones que pertenecen al MG. De manera que estas métricas expresen de manera adecuada los niveles de calidad de los patrones obtenidas respecto a todas las fuentes de datos.

Para asignar a cada patrón global ρ_{G_j} las métricas adecuadas, se asocia a cada uno un conjunto de patrones locales $P_j = \left\{\rho_{m'_{i_j}}\right\}; 0 \leq i \leq q$ de manera que cada patrón $\rho_{m'_{i_j}} = < id_{j_{\mathrm{m}_{\mathrm{i}}'}}, me_{j_{\mathrm{m}_{\mathrm{i}}'}} >$ se selecciona de un modelo local m_{i} diferente [12]. Esto se hace con el objetivo de elegir de cada modelo local el patrón $\rho_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}}}^{'}}$ que sea más similar al patrón global $\rho_{\mathrm{G}_{\mathrm{j}}}$; es decir, aquel patrón $\rho_{m'_{i_j}} \in \mathrm{m}_{\mathrm{i}}'$ que devuelva el menor valor en la función de distancia: $d_p\left(\rho_{G_j}, \rho_{m'_{i_j}}\right)$.

Puede ocurrir que exista más de un patrón que devuelva el menor valor de función de distancia, por lo que se debe "seleccionar" entre los patrones que devuelven el menor valor de función de distancia, cuál es el "más similar" al patrón global. En el caso de las reglas de asociación, para realizar este proceso de selección se utiliza una función de distancia entre patrones basada en la descrita en la ecuación (6). Pero modificando la función de distancia entre los atributos, con el objetivo de priorizar aquellas reglas locales que representan un subconjunto de la regla global. La función de distancia entre atributos para seleccionar las reglas "más similares" se define como:

$$\mathrm{d}_{\mathrm{AS}}\left(\mathrm{a}_{\mathrm{MG}_{\mathrm{l}_{\mathrm{j}_{\mathrm{k}}}}'}, \mathrm{a}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}_{\mathrm{k}}}}'}\right) = \begin{cases} 0 \text{ si } \mathrm{a}_{\mathrm{MG}_{\mathrm{l}_{\mathrm{j}_{\mathrm{k}}}}'} = \mathrm{a}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}_{\mathrm{k}}}}'} \\ 0.9 \text{ si } \left(\mathrm{a}_{\mathrm{MG}_{\mathrm{l}_{\mathrm{j}_{\mathrm{k}}}}'} = 0 \text{ y } \mathrm{a}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}_{\mathrm{k}}}}'} \neq 0\right) \\ 0.2 \text{ si } \left(\mathrm{a}_{\mathrm{MG}_{\mathrm{l}_{\mathrm{j}_{\mathrm{k}}}}'} \neq 0 \text{ y } \mathrm{a}_{\mathrm{m}_{\mathrm{i}_{\mathrm{j}_{\mathrm{k}}}}'} = 0\right) \\ 1 \text{ en caso contrario} \end{cases} \quad (13)$$

Por lo que, la distancia entre el valor del atributo k en el patrón j del modelo MG' y el valor del atributo k en el patrón j del modelo m_i', toma los valores 0, 0.9, 0.2 ó 1. Si los valores del atributo k en ambos patrones son iguales, toma

valor 0. Toma valor 0.9, si el atributo k no está presente en el modelo global y sí en el modelo local, en cuyo caso el atributo toma valor 0. Toma valor 0.2, si el atributo k no está presente en el modelo local y sí está presente en el modelo global, en cuyo caso el atributo toma valor 0. Toma valor 1, si el atributo k está presente en ambos patrones con valores diferentes.

Si en el proceso de selección se encuentran varios patrones similares que son un subconjunto del patrón global se escoge el patrón de menor calidad. La calidad de un patrón local se definió en la ecuación (10).

En el caso del agrupamiento, si se encuentran más de un *centroide* local con la misma semejanza respecto al *centroide* global se selecciona aleatoriamente uno de ellos.

Una vez que se tienen los conjuntos de patrones locales P_j asociados a cada patrón global, se prosigue a la aplicación de los operadores correspondientes para la estimación de las medidas que serán asignadas a cada patrón global. A continuación se definen los operadores que soporta el modelo propuesto para la estimación de las medidas me_{Gj} de los patrones globales.

$$O_{sum}\left(W_{Me_j}, Me_j\right) = \sum_{i=1}^{m} w_{me_{j_{\mu_i}}} * me_{j_{\mu_i}} \quad (14)$$

$$O_{prom}\left(W_{Me_j}, Me_j\right) = \frac{\sum_{i=1}^{m} w_{me_{j_{\mu_i}}} * \ me_{j_{\mu_i}}}{m} \quad (15)$$

$$O_{min}\left(W_{Me_j}, Me_j\right) = MIN_{i=1}^{m}(w_{me_{j_{\mu_i}}} * me_{j_{\mu_i}} \quad (16)$$

$$O_{max}\left(W_{Me_j}, Me_j\right) = MAX_{i=1}^{m}(w_{me_{j_{\mu_i}}} * me_{j_{\mu_i}}) \quad (17)$$

Donde Me_j y W_{Me_j} son los conjuntos de medidas locales y sus pesos respectivamente, que corresponden al patrón global ρ_{Gj}. El peso de un patrón W_{Me_j} representa el factor de semejanza entre el patrón global ρ_{Gj} y el patrón local seleccionado, el cual se define:

$$W_{Me_j} = 1 - \ d_p\left(\rho_{G_j}, \rho_{m'_{i_j}}\right) \quad (18)$$

En la integración de reglas de asociación, una vez ajustadas las métricas de los patrones del MG es necesario actualizarlo, eliminando aquellos patrones globales que no cumplen con el mínimo de soporte y confianza establecidos. En el caso de la integración de grupos el MG obtenido después de aplicar la estimación de métricas se mantiene con el mismo número de patrones. Finalmente se prosigue a la fase de Representación del modelo global resultante en el formato requerido, para ser entregado al usuario.

3 Estudio experimental

En este apartado, se documenta un estudio experimental realizado a los métodos propuesto con el objetivo general de: demostrar la aplicabilidad del Método de Integración de Modelos de Agrupamiento, *CMIM* y del Método de Integración de Reglas de Asociación*, ARIM*. Para demostrar la aplicabilidad se trazaron dos objetivos específicos: comprobar la factibilidad (obtiene modelos globales semejantes a los modelos locales) y la escalabilidad (observar el comportamiento aumentando el número de patrones locales)

Para la realización de este estudio, se utilizó una base de datos que colecta 8624 registros de pacientes con padecimientos de diabetes, obtenida a partir de una pesquisa realizada en la localidad de Jaruco, Provincia Mayabeque, Cuba. Esta base de datos consta de 5 atributos nominales que describen a los pacientes. Para obtener los modelos de MD de agrupamiento y de reglas de asociación, se utilizó la versión 3.6.1 de la herramienta Weka [14]. Para desarrollar los objetivos específicos se definieron las siguientes tareas:

1. Creación de los modelos de agrupamiento y de reglas de asociación a integrar.
2. Analizar el comportamiento de los algoritmos metaheurísticos en *CMIM* y *ARIM* determinando el que mejor se ajusta a las

características de nuestro problema, es decir, comprobar la Fase de Síntesis de *CMIM y ARIM*.
3. Variar los operadores de estimación de métricas, con el objetivo de determinar para cada métrica cuál es el operador de mejor comportamiento.
4. Determinar modelos de integración de agrupamiento y reglas de asociación, aumentando el número de modelos locales, con el objetivo de comprobar la escalabilidad de *CMIM y ARIM*.

Para la fase de Síntesis se aplicaron varios algoritmos metaheurísticos implementados en la biblioteca de clases BiCIAM [3]: Búsqueda Aleatoria (BA) [10], Escalador de Colinas Estocástico con Primer Ascenso (ECE-PA) [16] y Algoritmos Genéticos (AG) [16]. Para el ECE-PA o EC, se realizaron pruebas variando la manera de construir la solución inicial y la manera de generar un punto a partir de un estado candidato. Las variantes de construcción de solución inicial que se utilizaron fueron:

– Solución inicial aleatoria (SA): Selecciona aleatoriamente un conjunto de posiciones estableciéndoles valor “1”, de esta manera se incluye en el MG inicial los patrones correspondientes a las posiciones seleccionadas.
– Solución inicial que incluye un modelo local (SM): Selecciona aleatoriamente uno de los modelos locales existentes, y luego establece valor “1” para las posiciones correspondientes a los patrones del modelo seleccionado.

Para generar un punto a partir de un estado candidato, se utilizaron los operadores de mutación y de agrupamiento, variando las probabilidades para cada uno de ellos de manera tal que se utilicen ambos aleatoriamente o solamente el operador de mutación. Teniendo en cuenta esto se conciben dos variantes para generar un punto:

– Solamente aplicando el operador de mutación: Se establece valor 1 a la probabilidad asociado a ese operador.
– Aplicando ambos operadores aleatoriamente: Se establece valor 0.25 a la probabilidad de mutación y 0.75 a la probabilidad de agrupamiento, de esta manera se permite utilizar ambos operadores, aunque se da más peso al operador de agrupamiento con el objetivo de distinguir los resultados que obtiene el método al generar patrones nuevos a partir de otros.

Adicionalmente el AG requiere de parámetros especiales:

– Cantidad de elementos que formarán parte de la población: 20 individuos
– Operador de selección: Selección por truncamiento con el 30% de la población
– Operador de cruzamiento: Cruzamiento uniforme.
– Operador de mutación: Mutación en un punto.

En el caso de AG, se realizaron experimentos partiendo de dos tipos de poblaciones iniciales de puntos. En el primer tipo todos los individuos son generados aleatoriamente y en el segundo se incorpora un individuo igual a un modelo local aleatorio y el resto aleatorios. Luego de generada la población inicial, el AG itera realizando selección, cruzamiento y mutación según los parámetros descritos anteriormente.

En todos los entornos de pruebas se realizaron 30 ejecuciones con 5000 iteraciones para cada una de las variantes de algoritmos metaheurísticos. Todas las ejecuciones de los algoritmos se realizaron utilizando una computadora con una capacidad de procesamiento correspondiente a las siguientes características: 2 Gb de memoria RAM y microprocesador Intel Core 2 Duo con velocidad de 2.66 GHz. El *Java Heap Space* de la máquina virtual de Java se configuró con 1024 Mb.

3.1 Aplicación del método de integración de modelos de agrupamiento

Sobre el conjunto de los 8624 datos se aplicaron los algoritmos de agrupamiento particional Expectation-Maximization (EM) y K-Means, ambos implementados en Weka, para generar varios modelos locales y el modelo correspondiente a los 8624 datos, que se denominará “modelo centralizado” (MC). Para generar los modelos locales se realizaron 3 y 7

particiones del conjunto de datos y se generaron varios conjuntos de modelos locales:

– Con las 3 particiones y aplicando EM y luego K-Means, se obtuvieron 3 modelos locales con un total de 11 patrones. En esta variante, se utilizó la opción que brinda EM para determinar la cantidad de grupos que existe en un conjunto de datos y luego se aplicó el algoritmo K-Means estableciendo como parámetro k (cantidad de grupos a generar) la cantidad de grupos obtenido por EM previamente. EM solo se utilizó para estimar la cantidad de grupos de manera no-supervisada ya que K-Means no tiene esa opción.
– Con las 3 particiones y aplicando solamente K-Means fijando el parámetro k, se generaron 3 modelos locales con un total de 30 patrones y 3 modelos locales con un total de 60 patrones.
– Con las 7 particiones se generaron 7 modelos con un total de 70 patrones, aplicando K-Means.

Para generar los MC se aplicaron los algoritmos de agrupamiento de la misma manera que se aplicaron para obtener los modelos locales. De esta forma se obtuvieron 4 MC (1 aplicando EM y luego K-Means, y 3 aplicando solo K-Means con el parámetro k fijo) para compararlos con los MG obtenidos con el método. Finalmente teniendo los modelos locales y los modelos centralizados se aplicaron las fases de Síntesis y Estimación de Métricas.

Al analizar los resultados obtenidos en la aplicación de los algoritmos metaheurísticos en estos entornos se observó que:

– Cuando el espacio de solución no es muy grande (2^{11} 11 patrones en total), el mejor algoritmo que se comporta es BA, sin embargo cuando aumentan el número de patrones y el número de modelos (2^{30}, 2^{60}, 2^{70}), el algoritmo que mejor se comporta es AG en todos los entornos de prueba experimentados.
– El algoritmo que se ejecutó en menor tiempo en todos los entornos fue AG.
– Los mejores resultados se obtienen cuando se inicia la búsqueda con una solución que incluye un modelo local aleatorio.
– Al construir puntos del espacio aplicando ambos operadores, los algoritmos no varían mucho su funcionamiento, por lo que se puede decir que son más sensibles a la forma de generar la solución inicial que al operador para generar la vecindad que se aplique.

Para comprobar la factibilidad y escalabilidad del método en los entornos de prueba se observaron las siguientes medidas:

Al analizar los datos de la Tabla 1, y los distintos escenarios de prueba, se observaron varias tendencias. Primeramente se determinó que los valores de óptimos (menores valores de la FO) obtenidos estaban directamente relacionados con la diferencia entre la cantidad de patrones que como promedio tienen los modelos locales (PP_L) y la cantidad de patrones mayor entre los modelos locales (PM_L). La relación entre estas variables, significa que los resultados del método propuesto dependen de las características de los modelos locales, es decir, mientras más equilibrados estén los mismos en cuanto a cantidad de patrones, menor será la diferencia entre PM_L y PP_L y por tanto menor será el valor de la FO, que refleja la semejanza del MG al resto de los modelos locales.

Los valores de distancia entre MC y MG mostrados en la Tabla 1, fueron calculados para

Tabla 1. Resultados obtenidos en la aplicación de CMIM

Medidas	**3 modelos**			**7 modelos**
Cantidad de patrones a integrar	**11**	**30**	**60**	**70**
Cantidad de patrones iguales (MC-MG)	3/5	7/16	16/33	7/14
Menor valor de la FO	0.19	0.27	0.29	0.45
Distancia MC-MG	0.25	0.19	0.15	0.16
Tiempo promedio (seg)	0.2	2	6	3

observar la semejanza entre estos modelos en cuanto a los patrones que lo componen. Para esto se halló la distancia entre estos modelos usando la ecuación (4) pero sin tener en cuenta los patrones "no alineados". Esto se debe a que se generaron varios modelos de agrupamiento con todos los datos unidos (varios MC) de manera tal que la cantidad grupos a obtener fuera la misma que la obtenida por el MG con el cual se iba a comparar, por tanto, solamente se refleja la semejanza entre estos modelos en cuanto a los valores de los atributos de cada patrón (según ecuaciones (9)). Teniendo en cuenta esto, se puede decir que los valores de distancia obtenidos muestran que existe semejanza entre los patrones de ambos modelos, siendo en algunos casos los patrones exactamente iguales y en otros casos diferentes en 1 o 2 atributos.

En cuanto al tiempo promedio de ejecución del método, se pudo observar que aumenta en dependencia de la complejidad de la FO ($O(s + t * q * x)$), la cual depende de la cantidad de patrones nuevos (*s*), la cantidad de patrones totales en los modelos locales (*t*), la cantidad de patrones incluidos en el MG (*q*) y de la cantidad de atributos (*x*). En el estudio experimental realizado se determinó la relación que existe entre la complejidad y el tiempo cuando se ejecuta el método utilizando AG, dejando fijo el valor de *x* (5 atributos) y sin tener en cuenta el valor de *s* (AG no aplica el operador que obtiene patrones nuevos). Para esto se realizó un gráfico de dispersión entre varios puntos que reflejaban distintas medidas de complejidad y el tiempo de ejecución correspondiente. Con esta gráfica se utilizó la opción de agregar la línea de tendencia entre los puntos y se obtuvo que la dependencia entre estas variables (tiempo y complejidad) es lineal y está dada por la función: $T = 0.578x + 479.2$ con un índice de correlación de 0.999. Esto indica, que a medida que aumente la complejidad de la FO, aumentará linealmente el tiempo de ejecución del algoritmo metaheurístico AG para dar una solución.

Utilizando esta función se determinó que en condiciones extremas donde la cantidad de patrones totales (*t*) fuera 400 y todos estuvieran incluidos en el MG (*q*=400), por tanto la complejidad fuera cuadrática, el método se demoraría aproximadamente 8 minutos en ejecutarse utilizando AG, el cual puede ser considerado un tiempo pequeño.

En cuanto a la aplicación de la fase de Estimación de Métricas, se observaron los valores obtenidos por los operadores al estimar los valores de las métricas de los patrones existentes en los MG que coincidían con los patrones de los MC, los cuales fueron 55 en total. Para la métrica cobertura se aplicaron los operadores: O_{min}, O_{max}, O_{prom}, y O_{sum}, y para la precisión se aplicaron los operadores: O_{min}, O_{max} y O_{prom}. En esta última, no se aplicó el *Osum* porque esta medida está entre 0 y 1 y aplicando dicho operador se podían obtener valores absurdos. Para representar el resultado obtenido por los operadores se utilizó el porcentaje de "error relativo" (E_r) que indica cuan cercano fue el valor estimado por los operadores al valor real del patrón en el MC. En la siguiente tabla se muestra un resumen de los resultados obtenidos por los operadores en cuanto a los valores promedios de E_r y la cantidad de veces que cada uno se comportó mejor que el resto.

Según los resultados obtenidos en la Tabla 2, para estimar el valor de cobertura de los patrones, el operador que menor promedio de porcientos de E_r obtiene es el O_{prom}. Sin embargo, haciendo una comparación entre los cuatro operadores aplicados para esta métrica, en cuanto a los valores de E_r estimados por patrón, se observa que el O_{max} obtiene el menor E_r tres veces más que el O_{prom}. Teniendo en cuenta que la diferencia en cuanto a la segunda medida no es tan significativa, se considera que el operador de mejores resultados para la métrica de cobertura es el O_{prom}, aunque es necesario reajustarlo para que los porcientos de E_r sean menores y se estimen valores más cercanos (menos del 50% de E_r) a los de los patrones del MC.

En cuanto a la métrica de precisión el operador de mejores resultados en ambas medidas observadas, fue el O_{max}, con % de E_r relativamente pequeños y resultando el mejor 44 veces de las 55 aplicadas. El O_{min} en este caso obtuvo resultados mucho peores que el resto ya que no estimó valores cercanos ninguna de las veces aplicadas.

Tabla 2. Aplicación de Operadores de Estimación en CMIM

Medidas	Ajuste de Cobertura				Ajuste de Precisión		
	O_{min}	O_{max}	O_{prom}	O_{sum}	O_{min}	O_{max}	O_{prom}
Promedio de E_r al estimar 55 patrones	70%	119%	**66%**	350%	30%	**6%**	15%
Cant. de veces que obtuvo el menor E_r	8	**19**	16	12	0	**44**	11

3.2 Aplicación del método de integración de reglas de asociación

Para comprobar la factibilidad de *ARIM* se decidieron crear 4 conjuntos de entrenamientos: D, D1, D2 y D3. El conjunto de entrenamiento D lo conformaban un total de 8624 registros o instancias de pacientes. Posteriormente, a partir de este conjunto de entrenamiento se generaron aleatoriamente los conjuntos disjuntos, homogéneos entre sí en cuánto a sus atributos y de tamaños desiguales, respecto a D: D1, D2 y D3, de 1294 (15%), 2587 (30%), 4743 (55%) registros respectivamente.

Para la obtención de los modelos de MD locales (ML) se utilizaron los conjuntos de entrenamiento D1, D2 y D3. Se ejecutó el algoritmo *A priori* [1, 13] indicándose que devolviera todas las reglas de asociación que tuviesen un factor de confianza mayor o igual que 0.8, los restantes parámetros no se variaron, teniendo en cuenta las recomendaciones propuestas en la bibliografía consultada .La cantidad de reglas obtenidas para cada uno de los modelos locales fueron 55, 32 y 30 respectivamente. El espacio de búsqueda de soluciones fue 2^{117}, pues la cantidad de patrones locales a integrar fueron 117.

También se determinó el modelo de reglas de asociación (modelo centralizado, MC) tomando el conjunto de entrenamiento D, es decir, todos los datos en un único *dataset*. Pues es necesario comparar el modelo global resultante de la fase de síntesis con el MC. Dado que el objetivo de *ARIM* es integrar los modelos locales para obtener un modelo global que contenga patrones globales válidos, los que pudieran haber sido descubiertos de la unión de todos los *datasets*.

Al analizar los resultados de los algoritmos metaheurísticos en *ARIM* se observó el siguiente comportamiento:

– El AG fue el de mejor comportamiento en cuanto a tiempo de ejecución y valor promedio de la FO.
– De las variantes de inicialización de la población en AG, se observó que partiendo de la solución inicial que incluye un individuo igual a un modelo local aleatorio, se obtuvieron los mejores resultados, por tanto es el que mejor se ajusta a este método. Con esta configuración, este algoritmo convergió al mínimo de la FO en las 30 ejecuciones, obteniendo el mismo MG en cada una de ellas, obteniendo 30 patrones.
– El EC obtuvo los segundos mejores resultados en ambas medidas (tiempo de ejecución y valor promedio de FO) seguido por BA.

A continuación se presenta la aplicación experimental de la fase de Estimación de métricas de *ARIM* para lo cual se utilizará el MG comentado anteriormente. Para el desarrollo de esta tarea se ajustarán las métricas: soporte y confianza de cada una de las reglas del MG,

Tabla 3. Análisis comparativo de diferentes MG con el MC

Lista de Modelos	Cantidad Total Patrones	Tiempo Promedio (s)	MG	Comunes MG con MC	No alineados	Distancia MG-MC
3	117	35	21	21	6	0.21
5	216	83	25	24	2	0.07
7	304	131	20	13	7	0.26
10	446	255	24	12	3	0.11

utilizando los operadores comentados en el apartado Estimación de métricas de los patrones globales.

Para analizar la calidad de la estimación de las métricas de cada operador, se analizan aquellos patrones que forman parte del MG y del MC. Con el objetivo de calcular el error relativo de las métricas ajustadas respecto al valor de las métricas en el MC.

Los resultados obtenidos a partir de realizar este análisis fueron los siguientes: para ajustar el soporte de las reglas el operador que mejor resultado obtuvo fue el de Suma (*Osum*) y en el caso de la confianza el Promedio (*Oprom*), puesto que los errores relativos promedios fueron 0,0027 y 0,011 respectivamente.

Una vez estimadas las métricas de los patrones del MG es necesario actualizar el MG eliminando aquellos patrones que no cumplen con el mínimo de soporte y confianza establecidos. El mínimo de confianza establecido fue 0.8, ya que este fue el valor mínimo de confianza para generar los patrones en el MC y en los modelos locales. No obstante se aceptan patrones con un mínimo de confianza de 0.79, pues el error relativo promedio cometido fue de 0.01. El MG obtenido una vez culminada la fase de Estimación de métricas consta de 21 patrones.

Resulta necesario analizar la calidad del modelo global (MG) obtenido por *ARIM*. El análisis de la calidad del MG está dado por la comparación de MG con el Modelo Centralizado. El modelo global de reglas de asociación (MG) obtenido por *ARIM* es semejante al MC. Puesto que al calcular la función de distancia (4) de un modelo respecto al otro se obtiene un valor de 0.2180. El MC tiene 27 patrones y el MG 21 patrones en total. Todos los patrones del MG son comunes con el MC.

Por otra parte, para comprobar la escalabilidad de *ARIM* es necesario analizar su comportamiento en función del tiempo y la calidad de los modelos globales al variar las variables que influyen en la complejidad del método propuesto. En la aplicación de *ARIM* solamente se utilizó el operador de mutación, por lo cual no se generaron nuevos patrones y la complejidad del método se reduce a $O(t * q * x)$.

Para analizar la escalabilidad de *ARIM* se decidieron crear 4 conjuntos de modelos locales (ML) de tamaños 3, 5, 7 y 10 respectivamente. Para la obtención de los modelos de MD locales de los cuatro conjuntos de modelos, se crearon conjuntos de entrenamientos de datos de diferentes tamaños a partir del conjunto D. Se ejecutó el algoritmo *Apriori*, indicándose que devolviera todas las reglas de asociación que tuviesen un factor de confianza mayor o igual que 0.8. El total de patrones locales a integrar para los conjuntos de modelos de tamaño 3, 5, 7 y 10 fueron: 117, 216, 304 y 446 patrones respectivamente.

Para comprobar la escalabilidad se empleó *ARIM* sobre cada uno de los conjuntos de modelos locales, utilizando Algoritmo Genético con la configuración que resultó mejor. Se escoge este algoritmo para comprobar la escalabilidad, a partir de los resultados obtenidos sobre el

comportamiento de los algoritmos metaheurísticos en *ARIM.* Los resultados de la escalabilidad se basan en la calidad del MG que se obtuvo para cada uno de los conjuntos de modelos locales y en el tiempo de ejecución del método. Para analizar la calidad de estos MG es necesario compararlos con el MC. La Tabla 3 muestra la cantidad de patrones que se integraron para cada uno de los conjuntos de modelos (3, 5, 7, 10), resume los resultados relacionados al tiempo promedio de ejecución de *ARIM* en las 30 ejecuciones realizadas, así como la cantidad de patrones que se obtuvieron en los MG al culminar la Fase de Estimación de Métricas y de estos patrones cuántos son comunes con el MC, el cual contiene 27 patrones. También sintetiza la distancia de los MG al MC, la cual se basa en la distancia de un modelo a otro, definida en la ecuación (4).

Como se aprecia en la Tabla 3 los MG obtenidos son semejantes al MC, puesto que la mayoría de sus patrones son comunes con el MC y los valores de distancia del MG respecto al MC son bajos, considerando que estos pueden estar en un rango entre 0 y 2. También se observa que en la función de distancia entre dos modelos, el peso por la penalización de los patrones no alineados en el modelo local (ecuación (4), 2do sumando) es muy alto, en este caso el modelo local está referido a MC.

Como se aprecia en la Tabla 3, el tiempo promedio máximo de ejecución de *ARIM*, fue de 4 minutos al integrar los 10 ML con un total de patrones de 446. Este tiempo se considera aceptable teniendo en cuenta el volumen de patrones y modelos que fueron necesarios integrar. En este sentido es importante analizar la tendencia de tiempo de ejecución de *ARIM* al aumentar la cantidad de patrones a integrar y a su vez la cantidad de patrones que formarán parte del MG. Para analizar esta tendencia, se realizó un gráfico de dispersión entre varios puntos utilizando los resultados de las medidas descritas en la Tabla 3 relacionados con la cantidad de patrones a integrar y el tiempo promedio de ejecución. Luego se agregó una línea de tendencia entre puntos y se obtuvo que la dependencia entre estas variables está dada por la función $T = 0.614x - 34.98$ con un índice de correlación de 0.945. Este análisis de tendencia permite realizar una predicción en función de diferentes valores de cantidad de patrones a integrar. Por ejemplo para 1200 patrones el tiempo estimado de ejecución de ARIM sería 12 minutos aproximadamente, lo cual se considera aceptable.

4 Conclusiones

En este trabajo se han propuesto dos métodos para la integración de modelos de minería de datos de reglas de asociación y agrupamiento. A partir de los estudios realizados podemos concluir que el problema de la integración de modelos de minería de datos locales, constituye un problema de optimización y la utilización de algoritmos metaheurísticos resulta una variante efectiva para tratarlo. Además en la fase de Síntesis, el algoritmo metaheurístico de mejor resultados (en función del mínimo de la FO y el tiempo promedio de ejecución) fue Algoritmo Genético partiendo de un modelo local como solución inicial.

Los métodos propuestos son escalables, pues obtuvieron buenos resultados (respecto al tiempo de ejecución y de la calidad del MG) al aumentar la cantidad de patrones de los modelos locales. Los modelos globales obtenidos por *ARIM* y por *CMIM* tienen calidad, al ser semejantes al Modelo Centralizado. Los valores de la FO, que indican la semejanza entre el MG y los modelos locales, y está directamente relacionada con la composición de los modelos locales, de forma tal que si la cantidad de patrones en ellos es similar, se obtendrán mejores valores de la FO, y viceversa.

5 Líneas de trabajo futuro

1. Ajustar la función de distancia entre modelos de minería de datos para disminuir el peso de los patrones locales no alineados.
2. Probar los métodos propuestos con bases de datos públicas de mayor dimensión (cantidad de atributos) y comparar los resultados trabajos encontrados en la bibliografía.
3. Extender el método CMIM para integrar los grupos resultantes al aplicar algoritmos de agrupamiento jerárquico y grupos caracterizados por atributos numéricos o por

ambos tipos de datos (numéricos y nominales).

Referencias

1. **Agrawal, R. & Srikant, R. (1994).**Fast algorithms for mining association rules in Large Databases. *20th International Conference on Very Large Data Bases (VLDB'94),* Santiago de Chile, Chile, 487–499.
2. **Crestana-Jensen, V. & Soparkar, N. (2000).** Frequent Itemset Counting Across Multiple Tables. *4th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'00),* Kyoto, Japan, 49–61.
3. **Fajardo, J., Rosete, A. (2011).**Algoritmo Multigenerador de soluciones, para la competencia y colaboración de generadores metaheurísticos. *Revista Internacional de Investigación de Operaciones (RIIO),* 1(1), 57-63, ISSN: 2145-9517.
4. **Gionis, A., Mannila, H., &Tsaparas, P. (2005).**Clusteringaggregation. *21st International Conferenceon Data Engineering (ICDE 2005),* Tokyo, Japan, 341–352.
5. **Hore, P., Hall, L.O., &. Goldgof, D.B. (2009).**A scalable framework for cluster ensembles. *Pattern Recognition,* 42(5), 676–688.
6. **Horn, J. & Goldberg, D.E. (1994).**Genetic Algorithm Difficulty and the Modality of Fitness Landscapes. *Third Workshop on Foundations of Genetic Algorithms*, Colorado, USA, 243–269.
7. **Lange, T. & Buhmann, J.M. (2005).**Combining partitions by probabilistics label aggregation. *11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD'05),* Chicago, IL, USA, 147–156.
8. **Long, B., Zhang, Z., & Yu, P.S. (2005).**Combining multiple clusterings by soft correspondence. *Fifth IEEE International Conference on Data Mining (ICDM'05),* Houston, Texas, 282–289.
9. **Paul, S. & Saravanan, V. (2008).** Knowledge integration in a Parallel and distributed environment with association rule mining using XML data. *IJCSNS International Journal of Computer Science and Network Security*, 8(5), 334–339.
10. **Rosete, A. (2000)**. Una solución flexible y eficiente para el trazado de grafos basada en el Escalador de Colinas *Estocástico.* Tesis de Doctorado, Facultad de Ingeniería Industrial, CEIS, La Habana, Cuba.
11. **Strehl, A. & Ghosh, J. (2002).** Cluster Ensembles A Knowledge Reuse Framework for Combining Multiple Partitions. *The Journal of Machine Learning Research*, 3(Dec), 583–617.
12. **Wilford-Rivera, I., Ruiz-Fernández, D., Rosete-Suárez, A., & Marín-Alonso, O. (2010)**. Integrating Data Mining Models from Distributed Data Sources. *7th International Symposium on Distributed Computing and Artificial Intelligence* (DCAI´2010), Advances in Intelligence and Soft Computing, 79, 389–396.
13. **Witten, I.H. & Frank, E. (2000).** Nuts and Bolts: Machine Learnig Algorithms in Java. *Data Mining Practical Machine Learning Tools and Techniques with Java Implementations* (265–276), San Francisco, Calif.: Morgan Kaufmann.
14. **Witten, I.H. & Frank, E. (2005).**The Weka machine learning workbench. *Data Mining Practical Machine Learning Tools and Techniques,* Second Edition (363–427), San Francisco, Calif.: Morgan Kaufmann.
15. **Wu, X. & Zhang, S. (2003).** Synthesizing High-Frequency Rules from Different *Data Sources. IEEE Transactions on Knowledge and Data Engineering*, 15(2), 353–367.
16. **Yuret, D. & de la Maza, M. (1993)**. Dynamic Hill Climbing: Overcoming the limitations of optimization techniques. *Second Turkish Symposium on Artificial Intelligence and Neural Networks*, 208–212.
17. **Zhang, X. & Brodley, C.E. (2004)**. Solving cluster ensemble problem by bipartite graph partitioning. *Twenty-first international Conference on Machine learning (ICML'04), A*lberta, Canada, 36.

Daymi Morales Vega graduada de Ingeniería Informática en el año 2008 en el Instituto Superior Politécnico “José Antonio Echeverría” (Cujae), en La Habana, Cuba. En este mismo instituto realizó la maestría en Informática Aplicada, obteniendo el título de Máster en Ciencias en elaño 2010. Actualmente se desempeña como investigadora y docente en el Cujae. Se encuentra desarrollando una investigación doctoral en el tema de métodos de decisión multicriterios (MCDM) aplicados a la evaluación de la calidad de los servicios.

Diana Martín Rodriguez graduada de Ingeniería Informática en el año 2008 en el Instituto Superior Politécnico "José Antonio Echeverría" (Cujae) en La Habana, Cuba. En este mismo instituto realizó una maestría en Informática Aplicada, obteniendo el títulode Máster en Ciencias en el año 2010. Actualmente se desempeña como investigadora y docente en el Cujae. Se encuentra desarrollando una investigación doctoral en el tema de algoritmo evolutivos multiobjetivos para la obtención de reglas de asociación cuantitativas.

Ingrid Wilford Rivera graduada de Ingeniería Informática en el año 2004 en el Instituto Superior Politécnico "José Antonio Echeverría" (Cujae), en La Habana, Cuba.. En este mismo instituto realizó una maestría en Informática Aplicada, obteniendo el título de Máster en Ciencias en el año 2006. Realizó sus estudios de doctorado en la universidad de Alicante, España, en el tema de integración de modelos de Minería de Datos. Obtuvo el título de Doctora en Ciencias en el año 2010. Actualmente se desempeña como investigadora y docente en el Cujae.

Alejandro Rosete Suárez graduado de Ingeniería Informática en el año 1993 en el Instituto Superior Politécnico "José Antonio Echeverría" (Cujae) en La Habana, Cuba. En este mismo instituto realizó una maestría en Informática Aplicada, obteniendo el título de Máster en Ciencias en el año 1995. Realizó sus estudios de doctorado en el tema de optimización del trazado de grafos. Obtuvo el título de Doctor en Ciencias en el año 2000. Actualmente se desempeña como investigador y docente en el Cujae. Los temas de investigación de su interés están relacionados con la optimización combinatoria, metaheurística y minería de datos.

Artículo recibido el 04/02/2011; aceptado el 10/10/2011.