Universidad de las Ciencias Informáticas

Facultad 3

Título:

Implementación de un algoritmo Apriori-like para el minado de reglas de asociación en las declaraciones aduanales de mercancías en Cuba.

Trabajo de Diploma para optar por el título de

Ingeniero Informático.

Autor: Andy Fernandez Garabote

Tutor: Msc. Julio Cesar Diaz Vera

Junio, 2012

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 3 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Andy Fernandez Garabote

_____

Autor

Msc. Julio Cesar Diaz Vera

_____

Tutor

DATOS DE CONTACTO

Julio Cesar Diaz Vera, graduado de Ingeniería en Telecomunicaciones en la Universidad Martha Abreu ubicada en Villa Clara, Cuba. Máster en Gestión de Proyectos, grado científico alcanzado en la Universidad de la Ciencias Informáticas, La Habana, Cuba. Posee varias publicaciones nacionales e internacionales referentes al tema abordado en esta investigación, alcanzando una experiencia de más de 10 años.

Correo electrónico: jcdiaz@uci.cu

AGRADECIMIENTOS

A mi mamá, que siempre ha estado a mi lado regalándome todo el amor y apoyo necesarios para hacer realidad este momento, que ha sacrificado su vida para que yo pudiese obtener logros como este. A ti, mami, va mi más grande y mejor agradecimiento porque sin ti, no fuese la persona que soy.

A mi papá, a Carlos, a Marcos; que han estado presentes en los momentos en que los he necesitado. Sin su ayuda incondicional y desinteresada, mi carrera hubiese sido mucho más difícil de recorrer.

A mi hermana, Ailín, que siempre se preocupa por mí, manteniéndose al tanto de todas mis cosas y ayudándome en todo lo que puede, te quiero mi hermanita.

A mis hermanos de la UCI, los vandálicos, la nueva familia que he descubierto aquí. Han sido mi mayor apoyo durante toda la carrera. Han estado ahí para cualquier cosa que he necesitado y me han aceptado con todos mis defectos. Bernardo, Carlos, Boris, Daniel, Roberto gracias por estar siempre ahí, ustedes constituyen una parte importante de mi vida.

A mi tutor, Julio, a quien tuve la suerte de conocer en segundo año y que ha sido la persona a quien más he molestado durante todo este tiempo. Nunca me ha fallado ni como profesor ni como amigo. Julio, no tengo palabras para agradecer todo lo que desinteresadamente me has ayudado.

A las “perris” Maylevis y Dianela gracias por existir, por su ayuda incondicional y por los muchos momentos emotivos que compartimos juntos, de ustedes he recibido, entre muchas otras cosas, muy buenos consejos que no tengo como agradecer. Espero contar siempre con su amistad.

A Janet, Yelenis y Analay; la parte femenina de la pequeña comunidad de amigos que comenzó desde primer año y que ha durado durante todo el tiempo que hemos estado aquí, muchas gracias por existir y darme la oportunidad de conocerlas.

A Arturo, mi primer amigo en esta Universidad; con quien tuve que compartir la taquilla durante dos años seguidos. A Silvio, Yordan, Jorge, Miguel, Leodán, y todos lo que convivieron conmigo en los diferentes apartamentos durante estos años. De todos ustedes aprendí cosas nuevas e importantes, sobre todo a compartir, a escuchar, a entender a los demás y darme cuenta de muchos de mis defectos.

A todos aquellos que de una u otra forma han contribuido a mi formación como persona y profesional, espero contar con todos siempre y pueden estar seguros que siempre podrán contar conmigo.

Muchas gracias.

DEDICATORIA

A mi mamá, que siempre soñó con este día…

Te Quiero.

RESUMEN

El presente documento propone la implementación de un algoritmo para la extracción de reglas de asociación en las declaraciones de mercancías de las aduanas. Dicho algoritmo está basado en la propuesta presentada en (Agrawal R., 1994) bajo el principio de "Clausura descendente del soporte de ítemsets o conjuntos de elementos", el cual plantea que todo subconjunto de un ítemset frecuente es frecuente, mientras que cualquier supraconjunto de un ítemset no frecuente tampoco es frecuente (Medina Pagola, et al., 2007). Un ítemset es frecuente cuando cumple con un soporte mínimo definido con anterioridad. Los principales ajustes del algoritmo clásico están enfocados a la primera y más costosa etapa del mismo, la generación de ítemsets frecuentes. Se define una heurística que mejora la complejidad computacional de la misma, permitiendo la obtención de resultados relevantes en un menor tiempo de ejecución. Además se aplican ajustes en la segunda etapa o etapa de generación de reglas de asociación que también contribuyen a la reducción del costo computacional del minado de este tipo de reglas. El algoritmo Apriori-like obtiene reglas de asociación relevantes para el dominio de las aduanas contribuyendo a la detección de riesgos de fraudes en los procesos de importación – exportación de mercancías.

PALABRAS CLAVES

Algoritmo, Apriori, ítemsets, minería de datos, reglas de asociación.

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

INTRODUCCIÓN

La aduana de cualquier país cumple la misión de controlar la importación y exportación de mercancías, al tiempo que cobra los impuestos asociados a estas actividades. El cobro de los impuestos de aduana constituyen una fuente importante de ingresos al país, su monto está estrechamente relacionado al tipo y la cantidad de mercancías involucradas en la operación de importación-exportación. Existen tres comportamientos delictivos fundamentales que son utilizados para evitar el cumplimiento de estas obligaciones cuando las mercancías pasan por la aduana: el ocultamiento, la declaración de menores cantidades y la incorrecta clasificación de las mercancías. Todos estos comportamientos clasifican dentro de la denominación de fraude aduanal (Laleh, y otros, 2009).

La mayoría de los países consumen importantes recursos materiales y humanos con el fin de garantizar, mediante el chequeo físico, la veracidad de lo declarado en las operaciones aduanales. Este proceso tiene dos elementos negativos, el primero de ellos asociado a que el volumen de tráfico mercantil actual hace muy difícil examinar más del 10% de las mercancías y comúnmente solo el 1% de las examinadas es detectada como fraudulentas (Comisión, 2006), el segundo elemento está asociado al compromiso de las aduanas en la facilitación del comercio que se ve lastrado en la medida que aumenta el chequeo físico.

La Organización Mundial de Aduanas (OMA) promueve la utilización de las Tecnologías de la Información y las Comunicaciones para la gestión de los procesos aduanales. Dicha política está fuertemente fundamentada en la imposibilidad de aumentar las facilidades para el comercio, sin poner en peligro el cumplimiento de las normativas de cada país por medios no informáticos, como se ha reflejado en el marco normativo de la OMA y el protocolo de Kyoto (OMA, 1997).

Una buena parte de los países, sobre todo los desarrollados, utilizan sistemas de información que reducen la cantidad de papel y agilizan el despacho de las mercancías. Al mismo tiempo han introducido técnicas no intrusivas, como los rayos Gamma y los rayos X para agilizar la revisión física de las cargas. A pesar de ello, los volúmenes actuales de tráfico mercantil hacen imposible que las aduanas puedan mantener los estándares de tiempo para el despacho de mercancía a la par que realizan revisión física del 100 % de la carga.

En este escenario es deseable para las aduanas poder contar con mecanismos que a partir de datos previos permitan predecir aquellas cargas de mayor riesgo de fraude aduanal para que sean estas las que pasen a examen físico. La utilización de técnicas de minería de datos que analicen la información histórica y permitan crear modelos predictivos pudiese ser relevante para alcanzar resultados positivos. En particular la minería de reglas de asociación que posibilita la obtención de reglas relevantes para la clasificación del riesgo de fraude en las declaraciones de mercancías.

Es por esto que se define como **problema** a tratar en el presente trabajo ¿Cómo contribuir a la clasificación de riesgo de fraude en las declaraciones de mercancías a partir del uso de minería de reglas de asociación?

El **objeto de estudio** estará enmarcado en el minado de reglas de asociación, definiéndose como **campo de acción** el algoritmo Apriori.

El presente trabajo persigue como **objetivo general** implementar un algoritmo Apriori-like para la extracción de reglas de asociación en declaraciones de mercancías ante la aduana cubana. Para cumplimentar este objetivo se definen los siguientes objetivos específicos:

- Establecer el marco conceptual de referencia.
- Definir los requisitos especiales de los datos.
- Ajustar el algoritmo Apriori para transacciones que contengan datos categóricos.
- Definir el modelo de componentes.
- Probar la validez del resultado.

De los anteriores se derivan las tareas de investigación que se presentan a continuación:

1. Recopilación de la bibliografía referente.
2. Selección de la bibliografía.
3. Análisis de la bibliografía.
4. Determinación de los tipos de reglas de asociación que se ajustan al problema.
5. Establecimiento del mecanismo de extracción de ítemsets frecuentes.
6. Establecimiento del mecanismo de generación de reglas de asociación.
7. Definición de un marco arquitectónico.
8. Implementación de las funciones asociadas al algoritmo.

9. Validación del resultado obtenido.
10. Presentación de los resultados.

Se espera obtener como **resultado de la investigación** la implementación de un algoritmo basado en el Apriori capaz de extraer reglas de asociación válidas para la clasificación de riesgo de fraude aduanal, a partir de los datos históricos asociados a las declaraciones de mercancías que se encuentran en los archivos digitales de la aduana cubana.

En el **Capítulo 1** se desarrolla el marco conceptual de la investigación, se presentan los principales conceptos de minería de datos, extracción de conocimiento, se aclara el uso indistinto de estos conceptos por varios autores. Se abordan los principales objetivos de la extracción de conocimiento, definiendo el objetivo en que se enmarcan las reglas de asociación. Se definen estas últimas así como algunas variantes importantes que se han presentado para resolver ciertos problemas específicos. Por último se presenta el algoritmo Apriori básico, detallando sus dos procesos principales a partir de los cuales se obtienen las reglas de asociación relevantes.

El **Capítulo 2**, a través de un caso de estudio sobre las declaraciones de mercancías de las aduanas, se describe todo el proceso que conllevará a la propuesta de solución esclareciendo los principales inconvenientes que se pretenden resolver con la investigación. Se proponen los ajustes al algoritmo de extracción de reglas de asociación con los que se pretende mejorar la complejidad computacional de este proceso. Se presenta la solución implementada definiendo las entradas correspondientes al algoritmo, los procesos básicos del mismo así como sus salidas. Por último se realizan algunas pruebas al código para validar el correcto funcionamiento de la implementación.

El **Capítulo 3**, presenta los diferentes mecanismos utilizados para validar los resultados de una investigación. Entre ellos se encuentran los experimentos, modelos matemáticos, el razonamiento lógico y la demostración. Este último es el seleccionado para validar la solución propuesta en esta investigación por adaptarse a las características de la misma. Se describen los recursos computacionales utilizados para desarrollar la demostración. Se incluye además la descripción de todos los elementos que conforman el dominio de la demostración. Por último se presentan los resultados obtenidos y se realiza una detallada discusión de los mismos, demostrando la validez de la presente investigación.

1. CAPÍTULO 1: MARCO CONCEPTUAL

1.1. Introducción

El aumento del volumen y la variedad de la información que se encuentra informatizada en bases de datos digitales ha crecido espectacularmente en las últimas décadas. Esta información, procedente de los sucesos históricos de las diversas organizaciones, puede ser utilizada para explicar el pasado, entender el presente y predecir la información futura. Para lograrlo es necesario analizar exhaustivamente los datos históricos con el incentivo de obtener beneficios en función de las diferentes entidades.

Tradicionalmente el análisis de los datos contenidos en bases de datos se efectúa utilizando lenguajes generalistas de consultas. Algunos lenguajes como el estructurado de consultas (SQL por sus siglas en inglés) permiten generar información resumida en informes. Esta solución es muy poco flexible y sobre todo poco escalable ante grandes volúmenes de datos.

Otras herramientas utilizadas para analizar los datos son las estadísticas. Algunos paquetes estadísticos son capaces de inferir patrones a partir de los datos. El problema radica en que generalmente estos no funcionan bien para las bases de datos actuales con millones de registros, además no se integran bien con los sistemas de información existentes (Orallo Hernández, y otros, 2004).

Trabajando en pos de crear mecanismos que posibiliten a las empresas aprovechar los datos históricos, se ha hecho patente la necesidad de una nueva generación de herramientas y técnicas computacionales para soportar la extracción de conocimiento útil desde los datos disponibles, muchos de ellos enmarcados en el dominio de la “Minería de Datos”.

1.2. Minería de datos

Comúnmente es utilizado de manera indistinta el concepto de minería de datos y el de extracción de conocimiento en bases de datos (KDD por sus siglas en inglés). Ambos están estrechamente relacionados pero no significan exactamente lo mismo. La primera definición de extracción de

conocimiento y una de las más referenciadas es la propuesta en (Fayyad, et al., 1996) donde se define como un proceso no trivial de identificación de patrones válidos, novedosos, potencialmente útiles, entendibles y comprensibles que se encuentran ocultos en los datos. Es importante destacar algunos vocablos utilizados por Fayyad: el término no trivial implica que el proceso no es superficial, los patrones que se obtengan no deben ser evidentes, deben tener la capacidad de producir algún efecto, novedosos y potencialmente útiles refiere que sean poco conocidos o totalmente nuevos y produzcan beneficios considerables, además destaca que deben ser factibles al entendimiento humano.

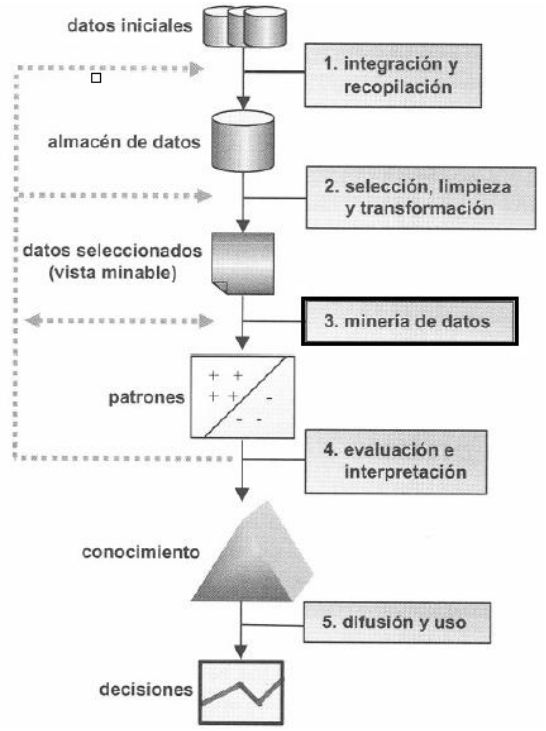


Figura 1. Fases del proceso de extracción de conocimiento en bases de datos (Fayyad, et al., 1996).

En la Figura 1 se muestran las fases del proceso de extracción de conocimientos, una de ellas es la minería de datos, esta se encarga de extraer los patrones ocultos en los datos a partir de la vista minable, datos previamente procesados con el fin de eliminar los elementos no deseables que puedan existir en ellos. Esta es la fase de mayor importancia y de mayores dimensiones dentro de todo el proceso, razón por la que muchos autores usan los términos de manera indistinta. En esta investigación se enmarca la minería de datos como fase dentro del proceso de extracción de

conocimiento, definiéndola como una familia de métodos computacionales que tienen como objetivo recolectar y analizar datos relacionados a un sistema de interés con el propósito de ganar un mejor entendimiento del mismo (Triantaphyllou, 2010). Este concepto se ajusta correctamente al objetivo del presente trabajo, refiriéndose a aplicar métodos computacionales en busca de una mayor comprensión del domino que se analiza, dejando la inferencia de conocimiento novedoso al proceso KDD y centrándose solamente en el análisis de los datos.

El proceso de extracción de conocimiento persigue dos objetivos fundamentales, la predicción y la descripción. El aprendizaje automático es la disciplina que se encarga de estudiar todo lo referente al primero ellos. El segundo se ajusta a la definición de minería de datos propuesta en (Triantaphyllou, 2010) donde el propósito no es otro que describir un sistema de interés.

Asociadas a estas dos vertientes existen una serie de tareas que se encargan del uso de varias variables conocidas para intentar determinar los valores desconocidos de otras y de buscar patrones interpretables que expliquen la naturaleza de los datos, refieriéndose a la predicción y la descripción respectivamente (Jimenez Ruiz, 2010). Algunas de ellas son:

- **Clasificación**: tiene como objetivo pronosticar el valor o la clase que puede tomar un atributo en función de los valores que toman otros atributos.
- **Regresión**: pretende pronosticar el valor numérico que tiene un determinado atributo.
- **Agrupamiento**: intenta crear grupos de individuos en función de sus similitudes de manera que los objetos de un grupo son muy similares entre sí y muy diferentes a los de otro grupo.
- **Reglas de asociación**: identifican relaciones no explícitas entre atributos categóricos.

El hecho de que existan dos objetivos bien marcados dentro del proceso de KDD no quiere decir que estos no se relacionen. Para hablar de predicción es necesario contar primeramente con una descripción detallada del dominio sobre el que se actúa.

Las reglas de asociación como tarea descriptiva, ofrecen como resultado una serie de patrones cuya interpretación puede ser utilizada para predecir comportamientos futuros en los datos analizados. El siguiente epígrafe abordará detalladamente los aspectos asociados a estas.

1.3. Reglas de asociación

El minado de reglas de asociación recibe especial atención por aquellos que enmarcan sus esfuerzos en el campo de la minería de datos, lo cual sucede gracias a sus probados resultados en este ámbito. Se comienza a hablar de este tipo de reglas en (Agrawal, et al., 1993) donde plantea que siendo $I = \{i_1, \dots, i_n\}$ un conjunto finito de ítems, *D* una base de datos que contiene las transacciones o subconjuntos propios de ítems no vacíos que pertenecen a *I*, asociándosele además un identificador único. Se define una regla de asociación como una implicación de la forma $X \rightarrow Y$, donde $X, Y \subset I$ son conjuntos de elementos o ítems, llamados ítemsets, nombrando a *X* como antecedente, *Y* como consecuente y cumpliendo que $X \cap Y = \emptyset$.

Es importante aclarar que un ítemset o conjunto de ítems es aquel cuyos elementos son atributos, variables o campos de cualquier base de datos. Su tamaño se determina de acuerdo a la cantidad de ítems que contiene, definiéndose como *k*-ítemset, donde *k* es igual al número de ítems (Medina Pagola, et al., 2007).

En la expresión $\{pan, mantequilla\} \rightarrow \{refresco\}$, el antecedente de la regla sería el conjunto de ítems formado por $\{pan, mantequilla\}$, el consecuente sería el formado por $\{refresco\}$ y la intersección entre ambos sería nula.

El descubrimiento de reglas de asociación, en grandes volúmenes de datos, suele generar un gran número de ellas y se corre el riesgo de que no todas sean realmente relevantes. Para asegurar la relevancia de los resultados existen métricas que permiten determinar su nivel de veracidad, estas se conocen como medidas de interés, clasificándose en objetivas y subjetivas de acuerdo a la intervención o no del usuario en la determinación de los resultados. Algunas de ellas se muestran a continuación.

Importancia:

La medida de importancia de una regla de asociación, también conocida como RIM por sus siglas en inglés, se apoya en la teoría de elementos rugosos para obtener reducciones o subconjuntos de atributos que sean suficientes para definir todos los conceptos inmersos en los datos, plantea que si

una regla *r* es generada con mayor frecuencia desde los diferentes conjuntos de reglas *CR* generadas a partir de las reducciones propuestas, esta es más importante que las generadas en menor frecuencia desde la misma fuente (Li, et al., 2009).

$$RIM(r_i) = \frac{|\{cr_j \in CR | r_i \in cr_j\}|}{n}$$

Según la fórmula anterior, la importancia de la regla *r* se calcula a partir de la división entre el número de veces que esta aparece contenida en los conjuntos de reglas creadas, partiendo del conjunto de reducciones y el total de elementos que este contiene.

Una vez calculado el valor de la importancia de todas las reglas se establece un ranking entre ellas, ordenándolas de mayor a menor de acuerdo al resultado, de manera que aquellas reglas cuyos resultados sean mayores serán las primeras y por tanto se consideran como las de mayor importancia.

Soporte y Confianza:

Entre las medidas de interés utilizadas para medir la significancia de las expresiones que se obtienen al aplicar técnicas de minado de reglas de asociación, las más populares son el soporte y la confianza (Jimenez Ruiz, 2010).

Se define como soporte de un ítemset la probabilidad de ocurrencia del mismo en una transacción *t* de la base de datos *D*. Calculándose de la forma siguiente:

$$sop(X) = \frac{|\{t \in D | X \subseteq t\}|}{|D|}$$

En este caso se calcula el soporte del ítemset *X* contando las transacciones en que este aparece y dividiéndolo entre el total transacciones de la base de datos *D.*

El soporte de una regla de asociación se calcula de manera similar al de un ítemset como se muestra a continuación:

$$Sop(X) = \frac{|\{t \in D | X \cup Y \subseteq t\}|}{|D|}$$

Con la particularidad de que se cuenta la cantidad de apariciones conjuntas entre el antecedente *X* y el consecuente *Y* en las transacciones *t* de la base de datos *D*, luego se divide entre el total de estas últimas.

La confianza está dada de acuerdo al porcentaje de transacciones *t* en *D* que contienen $X \cup Y$ dividido entre todas aquellas que contienen al antecedente, como se muestra a continuación:

$$Conf(X) = \frac{|\{t \in D | X \cup Y \subseteq t\}|}{|\{t \in D | X \subseteq t\}|}$$

Basados en estas métricas, las reglas de asociación cuyo soporte y confianza sea mayor o igual que los definidos por los interesados o especialistas en los datos analizados, se les denomina reglas fuertes (Jimenez Ruiz, 2010).

La interpretación de una regla de asociación teniendo en cuenta el soporte y confianza podría ser:

- El 75% de los clientes de una cafetería que compran pan, compran también refresco; un 10% de todas las ventas contienen estos elementos.

En este caso particular el 75% se refiere a la confianza de la regla y el 10% a su soporte.

Certeza:

Varios autores como (Brin, et al., 1997), (Silverstein, et al., 1998), (Sánchez, 1999) han señalado ciertos inconvenientes en cuanto al uso del soporte y la confianza, la mayoría de ellos asociados a la intervención de los usuarios en la determinación de los umbrales de evaluación, de manera que los resultados obtenidos están condicionados a la percepción de los mismos, existiendo la posibilidad de pasar por alto reglas relevantes.

El factor de certeza de una regla, métrica generalmente utilizada en el minado de reglas de asociación difusas (ver sección 1.3.2) se presenta con el propósito de erradicar algunos de estos inconvenientes, asegurando el descubrimiento de reglas relevantes sin la intervención de los usuarios.

El factor de certeza FC (Delgado, y otros, 2003) de la regla de asociación $\{A \rightarrow B\}$ se calcula de la siguiente forma:

$$FC(A \rightarrow B) = \frac{Conf(A \rightarrow B) - sop(B)}{1 - sop(B)} \quad si\ la\ Conf(A \rightarrow B) > sop(B)$$

$$FC(A \rightarrow B) = \frac{Conf(A \rightarrow B) - sop(B)}{sop(B)} \quad si\ la\ Conf(A \rightarrow B) \leq sop(B)$$

Asumiendo que si el $sop(B) = 1$ entonces la $FC(A \rightarrow B) = 1$ y si el $sop(B) = 0$ entonces la $CF(A \rightarrow B) = -1(\text{Delgado, et al., 2003})$.

Al evaluar las reglas de asociación mediante el factor de certeza los resultados obtenidos están contenidos en el rango $[-1,1]$. Si el valor resultante es positivo entonces la dependencia entre *A* y *B* es positiva, si resulta ser 0 significa que no existe tal dependencia, en cambio si es negativo indica que la dependencia también es negativa.

Según este tipo de medidas la regla de asociación $\{A \rightarrow B\}$ es muy fuerte si las reglas $\{A \rightarrow B\}$ y $\{\neg B \rightarrow \neg A\}$ son reglas fuertes atendiendo a la clasificación establecida por el soporte y la confianza anteriormente mencionados (Delgado, et al., 2003).

1.3.1. Reglas de asociación multiniveles

Las reglas de asociación multiniveles o generalizadas (estos términos serán usados indistintamente en lo adelante) propuestas por primera vez en (Ramakrishnan, et al., 1995) extienden los modelos de reglas de asociación, proporcionando la capacidad de descubrir reglas que se cumplen a lo largo de una jerarquía representadas de manera taxonómica. Las taxonomías pueden ser definidas sobre algunos atributos almacenados en bases de datos, la Figura 2 muestra una jerarquía de productos donde se define, entre otros, que “Arroz” es un “Cereal”, “Deshuesada” es un “Congelado” que este a la vez es una “Carne”.

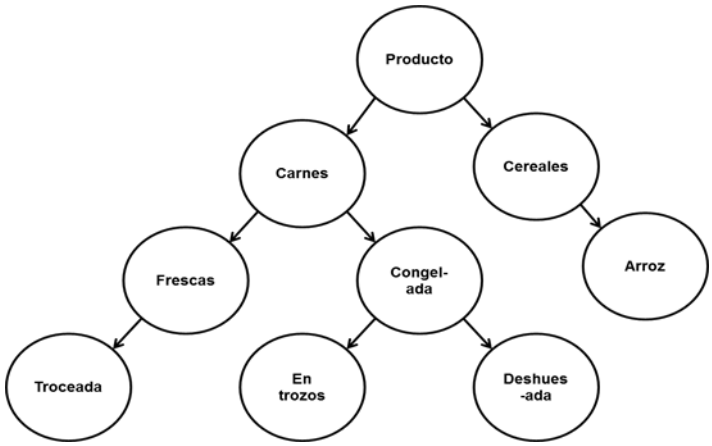


Figura 2. Taxonomía de productos.

El minado de reglas de asociación tradicional sobre los datos referentes a los productos anteriores pueden descubrir reglas como:

- El 46% de los clientes que compran arroz compran carne congelada deshuesada.
- El 30% de los clientes que compran arroz compran carne fresca troceada.
- El 41% de los clientes que compran arroz compran carne congelada en trozos.

El nivel de especificidad de estas reglas es tan alto que en la mayoría de los casos sus valores de confianza no son relevantes, provocando que se pierdan reglas que pudiesen ser de interés para los usuarios finales. Utilizando la taxonomía mencionada anteriormente, el nivel de especificidad puede ser ajustado, asegurando la obtención de reglas de mayor calidad como se presenta a continuación:

- El 82% de los clientes que compran arroz compran carne.

Para definir formalmente una regla de asociación multinivel (Sriphaew, et al., 2002) plantea que siendo $I = \{i_1, i_2, \dots, i_m\}$ un conjunto de ítems diferentes, $T = \{1,2, \dots, n\}$ un conjunto de identificadores de transacciones, $D = \{t_j | j \in T\}$ una base de datos de entrada donde t_j representa la transacción j, $\mathcal{T}$ una taxonomía en forma de árbol sobre los ítems. Una arista en $\mathcal{T}$ representa una relación “es - un” o padre

- hijo. Cuando existe una arista desde i_1 hasta i_2, el primero es llamado padre o ancestro del segundo, denotándose como $\hat{\imath}$.

Una regla de asociación multinivel o generalizada es una implicación de la forma $I_1 \longrightarrow I_2$ donde $I_1, I_2 \subseteq I, I_1 \cap I_2 = \emptyset$ y además ningún ítem perteneciente a I_2 es ancestro de ningún ítem perteneciente a I_1.

Los valores de soporte y confianza de estas reglas se calculan de forma similar a las tradicionales, con la particularidad que en este caso varía el concepto de ítemsets por el de ítemsets generalizado que es la base para calcular estos indicadores.

Se le conoce como ítemsets generalizado al conjunto $I_G \subseteq I$ cuando I_G que no contiene los ítems y sus ancestros a la vez. Además $t(I_G)$ se define como un conjunto de transacciones que contienen un subconjunto de I_G.

El soporte de I_G, denotado como $\sigma(I_G)$, se define como el porcentaje de las transacciones en las que I_G está presente entre el total de transacciones:

$$\sigma(I_G) = \frac{|\mathrm{t}(I_G)|}{|\mathrm{T}|}$$

Los ítemsets generalizados que cumplen este indicador definido por el usuario se les conoce como ítemsets generalizados frecuentes.

El soporte de la regla se calcula como el porcentaje que contiene las ocurrencias de la unión entre el antecedente y el consecuente, de la siguiente forma:

$$\sigma(I_1 \longrightarrow I_2) = \frac{|t(I_1) \cap t(I_2)|}{|T|}$$

La confianza de la regla se define como la razón entre el soporte de la regla y el soporte del antecedente, como se muestra a continuación:

$$\varsigma(I_1 \longrightarrow I_2) = \frac{\sigma(I_1 \cup I_2)}{\sigma(I_1)}$$

Se le llama regla de asociación generalizada a aquella que cumpla con el mínimo de confianza definido por el usuario.

Esta clase de reglas de asociación favorece un desempeño eficiente de los algoritmos empleados en su descubrimiento, puesto que reduce el número de ítemsets a explorar y por lo tanto el tamaño de la instancia.

1.3.2. Reglas de asociación difusas

Los primeros estudios relacionados con las reglas de asociación difusas propuestos en (Agrawal, et al., 1993) se centraban en el uso de etiquetas lingüísticas para generar reglas de asociación cuantitativas. Las reglas cuantitativas se basan en dividir el dominio del atributo en intervalos para luego descubrir reglas cuyos ítems son los pares <atributo; intervalo> en lugar de <atributo; valor>. Al realizar este procedimiento en un número fijo de intervalos aparece el llamado "problema de la frontera", que se traduce en la posibilidad de excluir intervalos de interés por estar muy cerca de los extremos. Ante esta situación se introduce el uso de conjuntos difusos en lugar de intervalos, de manera que cierto elemento está presente en un conjunto de este tipo con cierto grado de pertenencia que oscila entre $[0,1]$. Los conjuntos difusos permiten el uso de variables lingüísticas que facilitan la interpretación de las reglas descubiertas (Delgado, et al., 2008).

Los conceptos asociados a las reglas de asociación propiamente dichas no fueron diseñados para el dominio de la lógica difusa, algunos de ellos fueron adaptados según las exigencias asociadas a este tipo de reglas.

Según (Kuok, et al., 1998) un ítemset difuso es un conjunto de ítems de la forma $\{A_i;\ a_i\} \cup \ldots \cup \{A_i;\ a_i\}$ que se denota como $\{X, A\}$ donde *X* contiene todos los elementos o ítems y *A* los conjuntos difusos asociados a cada atributo en *X*.

Una regla de asociación difusa de acuerdo a (Gyenesei, 2001) es una expresión de la forma:

$$si\ X = \{x_1, \ldots, x_p\}\ es\ A = \{a_1, \ldots, a_p\}\ entonces\ Y = \{y_1, \ldots, y_p\}\ es\ B = \{b_1, \ldots, b_p\}$$

$$a_i \in \{conjuntos\ difusos\ asociados\ a\ x_i\}$$
$$b_i \in \{conjuntos\ difusos\ asociados\ a\ y_i\}$$

Siendo $D = \{t_i, \dots, t_i\}$ la base de datos que contiene todas las transacciones con atributos I y los conjuntos difusos asociados a esos atributos. Además $\{X, Y \in I\}$ son conjuntos disjuntos de ítems y $\{A, B\}$ contienen los conjuntos difusos correspondientes a los elementos de X e Y respectivamente. A la primera parte de la regla $\{X\ es\ A\}$ se le llama antecedente y la segunda parte $\{Y\ es\ B\}$ es el consecuente. Para abreviar la forma de escribir la regla desde ahora se denotará como:

$$\{X, A\} \rightarrow \{Y, B\}$$

Para calcular el soporte de los ítemsets difusos, según el enfoque que se propone en (Gyenesei, 2001) y (Kuok, et al., 1998), se utilizan las fórmulas siguientes:

$$fsop(X, A) = \frac{\sum_{t_i \in D} \prod_{x_i \in X} \mu^i_{a_i}(x_i)}{|D|}$$

Donde $|D|$ es el número de transacciones de la base de datos, $\mu^i_{a_i}(x_i)$ es el grado de pertenencia del atributo $x_j \in X$ en la transacción *i*-ésima al conjunto difuso $a_j \in A$.

Partiendo del resultado del soporte de un ítemset se puede calcular el soporte y la confianza de las reglas de asociación difusas a partir de las siguientes formulas:

$$fSop(\{X, A\} \rightarrow \{Y, B\}) = fsop(Z, C)$$

$$fConf(\{X, A\} \rightarrow \{Y, B\}) = \frac{fsop(< Z, C >)}{fsop(< X, A >)}$$

$$Z = [X \cup Y]\ y\ C = [A \cup B].$$

Otra definición de reglas de asociación difusas es la propuesta en (Delgado, et al., 2003), para el término transacción difusa plantea lo siguiente: una transacción difusa es un subconjunto difuso no vacío $\tilde{T} \subseteq I$ donde, $I = \{i_1, \dots, i_m\}$ representa un conjunto finito de ítems.

Para cada $i \in I$, se denota $\tilde{T}(i)$ al grado de pertenencia de i a la transacción difusa $\tilde{T}$. Se indica como $\tilde{T}(I_0)$ al grado de inclusión de un ítemset $I_0 \subseteq I$ en la transacción difusa $\tilde{T}$ y se define como:

$$\tilde{T}(I_0) = \min_{i \in I_0} \tilde{T}(i)$$

Se define T-set a un conjunto de transacciones ordinarias y FT-set a un conjunto de transacciones difusas.

Sean I un conjunto de ítems, D un FT-set, $X, Y \subseteq I$ dos subconjuntos difusos donde $X, Y \neq \emptyset$ y $X \cap Y = \emptyset$. Una regla de asociación difusa $X \rightarrow Y$ se cumple en D si y solo si:

$$\tilde{T}(X) \leq \tilde{T}(Y)\ \forall \tilde{T} \in D$$

El grado de inclusión de Y es mayor que el de X en todas las transacciones difusas $\tilde{T}$.

La evaluación de las reglas de asociación difusas puede realizarse a partir de la generalización del soporte y la confianza utilizando sentencias cuantificadas (Delgado, et al., 2003), (Delgado, et al., 2005). Una sentencia cuantificada es una expresión de la forma "*Q de los F son G*" donde F y G son dos subconjuntos difusos de un conjunto finito X y Q es un cuantificador difuso relativo. Los cuantificadores relativos son etiquetas lingüísticas que representan porcentajes difusos y que pueden expresarse como conjuntos difusos en [0,1]. De manera particular la propuesta de (Delgado, et al., 2003) plantea el uso de una familia de cuantificadores relativos denominada cuantificadores coherentes.

La evaluación de una sentencia cuantificada devuelve un valor en el intervalo [0,1] que representa el grado de cumplimiento de la misma.

El soporte de un ítemset $I_0 \subseteq I$ en D es la evaluación de la sentencia cuantificada:

$$Q\ de\ los\ D\ son\ \tilde{\Gamma}_{I_0}$$

Donde $\tilde{\Gamma}_{I_0}$ es un conjunto difuso en D definido como:

$$\tilde{\Gamma}_{I_0}(\tilde{T}) = \tilde{T}(I_0)$$

El soporte de una regla de asociación difusa $X \rightarrow Y$ en el conjunto de transacciones difusas D es $sop\ (X \cup Y)$, lo que equivale a la evaluación de la sentencia:

$$Q\ de\ los\ D\ son\ \tilde{\Gamma}_{X \cup Y} = \ Q\ de\ los\ D\ son\ (\tilde{\Gamma}_X \cap \tilde{\Gamma}_Y)$$

La confianza de la regla de asociación difusa $X \rightarrow Y$ en el conjunto de transacciones difusas D es la evaluación de la sentencia cuantificada:

$$Q\ de\ los\ \tilde{\Gamma}_X\ son\ \tilde{\Gamma}_Y$$

El soporte y la confianza dependerán del método de evaluación que sea escogido. La propuesta de (Delgado, et al., 2003) es utilizar el método GD (Delgado, et al., 2000) que ha demostrado tener buenas propiedades y mejor desempeño que otros. La evaluación de la sentencia "*Q de los F son G*" con el método GD se define como:

$$GD_Q\left(\frac{G}{F}\right) = \sum_{\alpha_i \in \Delta(G/F)} (\alpha_i - \alpha_{i+1}) Q\left(\frac{|(G \cap F)_{\alpha_i}|}{|F_{\alpha_i}|}\right)$$

Donde $\Delta(G/F) = \Lambda(G \cap F) \cup \Lambda(F)$, con $\alpha_i > \alpha_{i+1}$ para toda $i \in \{1, \ldots, p\}$. El conjunto F se asume está normalizado de lo contrario debe normalizarse y el factor de normalización aplicarse a $G \cap F$.
Este enfoque establece medidas de soporte y confianza que dependen del método de evaluación y el cuantificador elegido. En (Delgado, et al., 2003) y (Delgado, et al., 2005) se justifica el uso del método GD y el cuantificador Q_M que cumple: $Q_M(x) = x$.

Para el minado de reglas de asociación se han propuesto una serie de algoritmos que pretenden descubrir la mayor cantidad de reglas relevantes utilizando la menor cantidad de recursos computacionales, entre ellos se encuentra el SEMT, AIS, Partition, Eclat, entre otros. La mayoría de ellos surgen a partir del propuesto en (Agrawal, et al., 1994) bajo el nombre: Apriori. A continuación se detallan sus principales características.

1.4. Algoritmo Apriori

Uno de los algoritmos más propagados en el campo de la minería de reglas de asociación es el Apriori propuesto en (Agrawal, et al., 1994), este se basa en el principio de "Clausura descendente del soporte de ítemsets", el cual plantea que el soporte de cualquier subconjunto de un ítemset es mayor o igual que el soporte de ese ítemset. Además, todo subconjunto de un ítemset frecuente es frecuente, mientras que cualquier supraconjunto de un ítemset no frecuente tampoco es frecuente (Medina Pagola, et al., 2007). Un ítemset es frecuente cuando cumple con un soporte mínimo definido con anterioridad. Es importante destacar que Apriori asume que los ítems de cualquier transacción están ordenados lexicográficamente, para una correcta generación de los *k*-ítemsets candidatos (Motoda, et al., 2009).

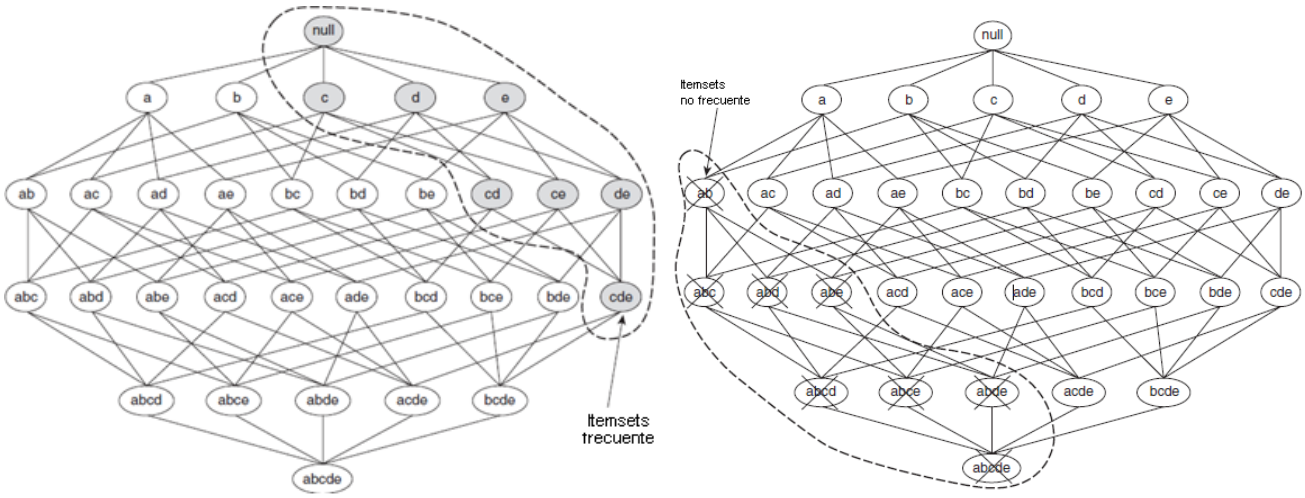


Figura 3. Principio "Clausura descendente del soporte de ítemsets".

En la figura anterior se muestran todos los posibles *k*-ítemsets generados a partir de los 1-ítemsets $\{a\},\{b\},\{c\},\{d\},\{e\}$, a la izquierda se puede observar que el ítemset $\{c,d,e\}$ es frecuente, por lo que todos los subconjuntos de ítemsets que él contiene son también frecuentes. A la derecha se puede observar que el ítemset $\{a,b\}$ no es frecuente, por lo tanto tampoco lo es ningún supraconjunto de ítemsets que lo contenga, como refiere el principio anterior.

Este algoritmo está compuesto por 3 secciones para la obtención de los *k*-ítemsets frecuentes:

- **Generación de candidatos:** se generan los candidatos de nivel *k* (*k*-1 ítemsets) haciendo uso de la información referida a los conjuntos frecuentes de la iteración anterior (*k*-1 ítemsets frecuentes).
- **Poda:** se garantiza que no pase al conteo de soporte los ítemsets que se conoce a priori que no van a ser frecuentes, eliminando a todos aquellos candidatos que no cumplan con que todos sus subconjuntos sean también frecuentes.
- **Conteo de soporte:** se cuenta la cantidad de apariciones de los *k*-ítemsets candidatos en la base de datos y se determina, según el soporte mínimo establecido por el usuario, el conjunto de los *k*-ítemsets frecuentes.

Estas secciones se repiten hasta que ya no haya más conjuntos frecuentes. Los 1-ítemsets frecuentes son obtenidos por el sistema al inicio del proceso (Mesa Rodriguez, et al., 2009). Como se muestra a continuación:

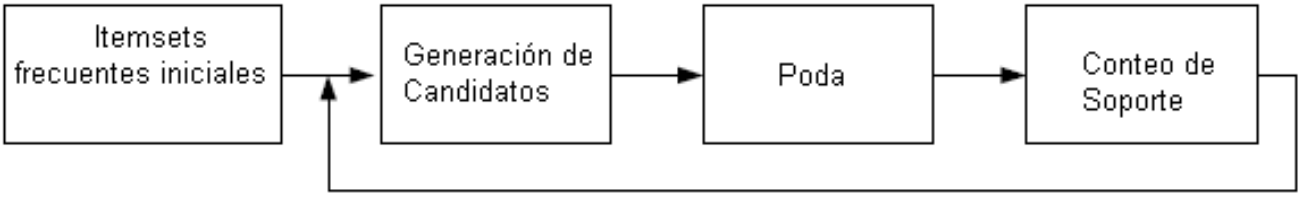


Figura 4. Estructura funcional de Apriori para obtener k-ítemsets frecuentes.

Una vez generados todos los ítemsets frecuentes se procede a generar todas las posibles reglas de asociación por cada uno de ellos. Para cada *k*-ítemset frecuente se puede generar $2^k - 2$ reglas de asociación, ignorando aquellas reglas que carezcan de antecedente o consecuente ($\{\emptyset \rightarrow A\}, \{A \rightarrow \emptyset\}$). Las reglas se pueden extraer dividiendo el ítemset en dos subconjuntos no vacíos, por ejemplo, el ítemset $\{A, B\}$ puede dividirse en los subconjuntos $\{A\}$ y $\{B\}$, de los cuales pueden obtenerse las reglas $\{A \rightarrow B\}$ y $\{B \rightarrow A\}$, estas deben cumplir con los umbrales de soporte y confianza definidos por el usuario (Agrawal, et al., 1994).

Apriori Básico:

Entrada: I, D, minsup, minconf

Salida: Conjunto de reglas de asociación con soporte y confianza ≥ que minsup y minconf.

Algoritmo:

1. Generar todos los ítemsets frecuentes con soporte ≥ minsup.
2. Dado un ítemset frecuente $X = \{X_i \dots X_k\}$ con $K \geq 2$ generar todas las reglas de la forma $X \mid \{x_j\} \rightarrow \{x_j\}$, siendo el soporte de dicha regla el soporte de X y la confianza, el cociente entre el soporte de este y el soporte de $X \mid \{x_j\}$.

1.5. Conclusiones parciales

El presente capítulo recoge los principales conceptos asociados a la minería de datos como etapa de mayor volumen e importancia dentro del proceso de extracción de conocimiento, específicamente centra la atención en el estudio de las reglas de asociación como técnica descriptiva del proceso anteriormente mencionado. La interpretación de los patrones obtenidos a partir del minado de reglas de asociación puede ser utilizada para predecir comportamientos futuros en los datos. Se definen algunas métricas utilizadas para evaluar la relevancia de las reglas en cuestión, así como algunos tipos específicos de reglas como las multinivel y las difusas, concluyendo que las medidas más utilizadas son el soporte y la confianza, aunque los resultados obtenidos con el uso de estas son subjetivos pues están condicionados por el intelecto humano.

Por último se presenta el algoritmo Apriori como uno de los más difundidos en este ámbito, debido a los buenos resultados obtenidos con su aplicación. Apriori es el algoritmo básico por excelencia para la minería de reglas asociación y uno de los más referenciados, siendo objeto de varias modificaciones de acuerdo a la naturaleza del dominio sobre el cual se desea aplicar, con el propósito de mejorar su desempeño.

2. CAPÍTULO 2: ANÁLISIS Y DISEÑO DEL ALGORITMO

2.1. Introducción

La declaración de mercancías es el documento oficial que contiene la información del contenido de las cargas, sirviendo de base para determinar el monto del impuesto a pagar en las operaciones de importación-exportación. En ocasiones se detectan problemas de mala clasificación que afectan el importe total a pagar por la transacción.

La muestra de datos pertenecientes a varias declaraciones de mercancías, registradas en las bases de datos de las aduanas que se presenta a continuación, puede derivar en información útil para la determinación del riesgo de mala clasificación de la mercancía, una vez que se apliquen técnicas de extracción de reglas de asociación.

En dicha muestra se recogen los datos referentes al origen o país donde se efectúa más del 50% de la manufactura de los productos. El campo “Mercancía” muestra, para mejor comprensión, la descripción correspondiente a la codificación establecida en el Sistema Armonizado de Clasificación de Productos (SACLAP), aunque en la práctica se almacena el código determinado; el tipo de moneda con que se procede a efectuar el pago y el importe total a pagar del impuesto o tasa que ampara esa declaración. Además, se muestra la cantidad de bultos que contiene la carga, el monto de los gastos por flete en USD, el peso bruto total del producto, el tipo de declaración de mercancías que puede ser: completa, provisional, anticipada o incompleta; almacenándose solo el identificador correspondiente a cada clasificación. También se registra la nacionalidad del medio de transporte en el que vienen los productos, los datos del Agente de Aduana o Apoderado debidamente registrado como declarante y por último si fue o no mal clasificada.

Origen	Mercancía	t_mon	Importe	c_bultos	Flete	Peso	Tipo_dm	Nac	Decl	m_clas
Colombia	carne porcina congelada deshuesada	USD	20000	1	612	2000	2	Brasil	1	true
Canadá	carne pollo congelado en trozos	USD	1000	2	520	500	1	Canadá	5	false
Colombia	carne porcina congelada deshuesada	USD	55000	5	460	8000	2	EEUU	2	true
Brasil	carne porcina fresca en trozos	USD	4500	3	702	1000	3	Brasil	2	false
Argentina	carne pollo fresca sin trocear	USD	8000	6	821	6000	1	Uruguay	4	true
Colombia	carne porcina congelada deshuesada	USD	2000	8	963	25000	2	Venezuela	1	true
Brasil	carne equina fresca en trozos	USD	4500	3	426	1000	3	Chile	2	false
México	carne pollo congelado sin trocear	USD	150000	10	148	55500	1	EEUU	2	true
Argentina	carne pollo fresca en trozos	USD	8000	6	821	6000	1	Uruguay	4	true

Tabla 1. Muestra de datos asociados a las declaraciones de mercancías aduanales.

Los algoritmos utilizados para el minado de este tipo de reglas de asociación, específicamente el algoritmo Apriori básico al que se hace referencia en el capítulo anterior, fue diseñado para el caso conocido como “cesta de compras”, es decir, toma como instancia de entrada, entre otras, una serie de transacciones donde cada una de ellas constituye una compra realizada por un cliente donde aparecen todos los productos que fueron adquiridos por él. Guardando uno en aquellos productos adquiridos y cero en los que no, de la forma siguiente.

Id	Pan	Tomates	Huevos	Mantequilla	Cerveza
1	0	1	0	0	1
2	1	0	1	1	0
3	0	0	0	0	1
4	1	0	0	1	0

Tabla 2. Transacciones correspondientes a la compra.

Utilizando este mecanismo, en la muestra anterior (Tabla 1) donde cada uno de los atributos presenta valores diferentes a cero y uno, habría que modificar la forma en que están escritos, transformando los atributos en la sucesión de cada uno de sus valores ajustándose a la entrada del algoritmo en cuestión de la siguiente forma:

Origen	Colombia	Canadá	Brasil	Argentina	México

Tabla 3. Transformación del atributo ¨Origen".

De manera que en vez de utilizar un solo campo para el origen de la mercancía se utilizarían cinco, procediendo de la misma forma con los demás atributos. En el caso del elemento “Mercancía”, cuya clasificación registrada en el SACLAP puede tomar más de 150000 valores diferentes, ocuparía dicha cantidad de campos en cada tupla. Una vez ajustados todos los atributos que contiene la declaración de mercancías a la entrada utilizada por Apriori, se procede a obtener los conjuntos de ítemsets frecuentes. Para conocer si un ítemset es frecuente es necesario calcular el soporte asociado al mismo. Para esto, Apriori propone recorrer cada una de las transacciones contando aquellas en las que aparezca el ítemset, dividiendo este resultado entre el total de transacciones. A partir del resultado de este soporte se define si el ítemset en cuestión es frecuente o no. Esta tarea se repite mientras se puedan crear nuevos ítemsets candidatos. Es lógico pensar que el costo computacional de esta operación es bastante alto cuando se procesan miles de ítems.

Una vez obtenidos los ítemsets frecuentes se procede a determinar las reglas de asociación para cada uno de ellos. Una vez que se tienen las reglas, se procede a descartar aquellas que no cumplan con los umbrales de soporte y confianza definidos por el usuario. En la muestra anterior (Tabla 1), al aplicar el algoritmo, se obtienen reglas como:

- El 100% de la carne porcina congelada deshuesada de origen colombiano se clasifica de manera incorrecta, con soporte = 30%.
- El 100% de los productos que vienen en un medio de transporte de nacionalidad estadounidense se clasifican de manera incorrecta, con soporte = 20%.
- El 100% de los productos de origen argentino que vienen en un medio de transporte uruguayo se clasifican de manera incorrecta, con soporte = 20%.

Origen	Mercancia	t_mon	Monto	c_bultos	Flete	Peso	Tipo_dm	Nac	Decl	m_clas
Colombia	carne porcina congelada deshuesada	USD	20000	1	612	2000	2	Brasil	1	true
Canada	carne pollo congelado en trozos	USD	1000	2	520	500	1	Canada	5	false
Colombia	carne porcina congelada deshuesada	USD	55000	5	460	8000	2	EEUU	2	true
Brasil	carne porcina fresca en trozos	USD	4500	3	702	1000	3	Brasil	2	false
Argentina	carne pollo fresca sin trocear	USD	8000	6	821	6000	1	Uruguay	4	true
Colombia	carne porcina congelada deshuesada	USD	2000	8	963	25000	2	Venezuela	1	true
Brasil	carne equina fresca en trozos	USD	4500	3	426	1000	3	Chile	2	false
Mexico	carne pollo congelado sin trocear	USD	150000	10	148	55500	1	EEUU	2	true
Argentina	carne pollo fresca en trozos	USD	8000	6	821	6000	1	Uruguay	4	true

Tabla 4. Muestra de datos donde se aprecian el origen de reglas de asociación.

Las reglas de asociación anteriores tienen su base en la (Tabla 4) donde se señala en verde la demostración de la primera de ellas, en rojo la segunda y con color azul la tercera. En la práctica el cúmulo de transacciones almacenadas en las bases de datos de la aduana es mucho mayor, alcanzando los millones, y crece a razón de miles por mes, de manera que el análisis anterior se hace muy engorroso.

La complejidad de Apriori está dada por $O_{(n)} = m * n^2$, donde m es el total de transacciones y n la cantidad total de ítems, por lo que a medida que estas aumentan, el costo computacional asociado a su procesamiento también aumenta. Teniendo en cuenta que los datos referentes a las declaraciones de mercancías constan de millones de tuplas con cientos de atributos que pueden tomar miles de valores diferentes, es imposible obtener resultados de forma manual. Los algoritmos de extracción de reglas de asociación, específicamente el Apriori, obtienen resultados relevantes con un alto costo computacional en conjuntos de datos genéricos. Es necesario ajustar dicho algoritmo para que permita determinar reglas de asociación válidas para el dominio de las declaraciones de mercancías en la aduana.

2.2. Propuesta de solución

Partiendo de las situaciones planteadas anteriormente se propone *Apriori-like*, algoritmo que realiza varios ajustes al Apriori presentado en (Agrawal, et al., 1994), pretendiendo adicionar la posibilidad de

extraer reglas de asociación a partir de transacciones que contengan datos categóricos y mejorar su complejidad computacional en el dominio de las declaraciones de mercancías de las aduanas.

Uno de los principales problemas del algoritmo Apriori básico consiste en el número de pases que este realiza sobre el conjunto de datos para determinar el soporte de los ítemsets. Este inconveniente provoca un aumento considerable en el tiempo de ejecución del mismo. Algunos autores como (A. Savasere, 1995), (Chung., 2001) proponen el particionamiento de las transacciones o los ítems respectivamente. Aunque estos algoritmos suelen ser eficientes en datos dispersos, presentan el problema de realizar tantos recorridos como cantidad de particiones se tengan, y al procesar cada partición, necesitan generar los conjuntos candidatos de todos los tamaños. Contar al mismo tiempo los conjuntos candidatos de todos los tamaños resulta costoso, principalmente en las primeras particiones, las cuales generan el mayor número de candidatos (Hernández León, et al., 2010).

En la etapa de generación de ítemsets frecuentes, *Apriori-like*, propone contar el soporte de todos los ítemsets de un mismo tamaño, de manera que el número de pases sobre las transacciones se reduce considerablemente sin hacer ninguna de las iteraciones demasiado costosas. En la propuesta hecha en (Agrawal, et al., 1994) el número de iteraciones sobre los datos es igual a la suma de todos los *k*-ítemsets frecuentes, multiplicado por k, en el caso particular de las declaraciones de mercancías, el número de ítems supera los miles, por lo que reducir esta cantidad de iteraciones a solo una por cada generación de nuevos candidatos, incide favorablemente sobre el costo computacional.

El segundo subproceso que define el Apriori se encarga de generar todas las reglas posibles a partir de un ítemset dado, descartando aquellas que no sean relevantes atendiendo al soporte y la confianza. Para esto utiliza el método clásico de extracción de reglas de asociación (Agrawal, et al., 1994). La lógica de este método consiste en extraer todas las posibles reglas teniendo en cuenta todas las combinaciones posibles sobre los subconjuntos del ítemset. *Apriori-like* propone solo extraer la mitad de estas reglas, puesto que la otra mitad, refiere las mismas reglas con la particularidad de que se invierten el antecedente y el consecuente como se muestra a continuación:

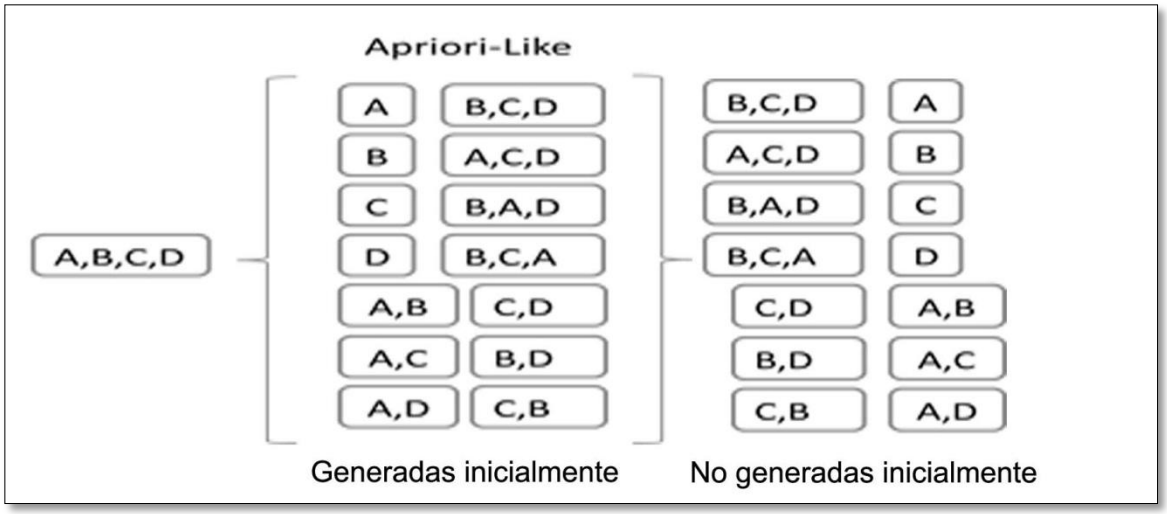


Figura 5. Reglas de asociación generadas y no generadas inicialmente por Apriori-like.

Una vez generadas las reglas iniciales, *Apriori-like,* propone invertir el antecedente por el consecuente de la misma, calculando nuevamente su soporte y confianza, en caso de cumplir con los parámetros establecidos se guarda dicha regla, de lo contrario se descarta. Lo anterior asegura que no existan reglas sin evaluar y agiliza el proceso de extracción de las mismas.

2.3. Solución

La implementación del algoritmo *Apriori-like* recibe cuatro parámetros. El primero de ellos es el conjunto de transacciones, las mismas se encontrarán almacenadas en una base de datos (D) que una vez procesada la información contenida en ella, se escribirá en un fichero en texto plano. Los datos recopilados en dicho fichero deben cumplir con el formato:

$$tabla - atributo: valor, tabla - atributo: valor, \ldots,$$

donde $tabla$ responde al nombre que llevan las tablas en la base de datos; $atributo$ representa el nombre de los campos contenidos en las tablas y $valor$ almacena los valores correspondientes a cada campo. Es importante que no queden espacios entre las nomenclaturas ni los signos empleados, los separadores a utilizar deben ser solo los definidos anteriormente en el mismo orden propuesto. A

continuación se muestra un ejemplo de los datos de la muestra (Tabla 1) transformados al formato anteriormente descrito:

```
declaracion_mercancia-codarticulo:016,declaracion_mercancia-nodm:00039
declaracion_mercancia-codarticulo:001,declaracion_mercancia-nodm:00050
declaracion_mercancia-codarticulo:002,declaracion_mercancia-nodm:00050
declaracion_mercancia-codarticulo:003,declaracion_mercancia-nodm:00050
declaracion_mercancia-codarticulo:001,declaracion_mercancia-nodm:00049
declaracion_mercancia-codarticulo:002,declaracion_mercancia-nodm:00049
declaracion_mercancia-codarticulo:001,declaracion_mercancia-nodm:00047
declaracion_mercancia-codarticulo:001,declaracion_mercancia-nodm:00054
```

Figura 6. Base de transacciones de las declaraciones de mercancías.

El segundo parámetro que recibe es el conjunto de todos los ítems que pueden estar contenidos en las transacciones. Estos ítems deben estar escritos en el formato descrito anteriormente y ubicados en la primera línea del fichero donde se guardan las transacciones.

Las últimas dos entradas definidas para este algoritmo son los valores mínimos de soporte y confianza respectivamente. Estos valores oscilarán ente 0 y 100 de acuerdo al porciento que determine el usuario para que una regla sea relevante.

El algoritmo *Apriori-like* al ser una variante del Apriori básico, además de las entradas clásicas, mantiene los mismos subprocesos básicos de generación de ítemsets frecuentes y generación de las reglas de asociación relevantes. Para la implementación del mismo se diseñaron dos clases principales:

- La clase “Rule” contiene las propiedades básicas de las reglas de asociación definidas como el antecedente, el consecuente y la confianza.
- La clase “Apriori” contiene una lista de todas las transacciones así como de los ítems, contiene además un listado de los ítemsets que resulten frecuentes y una serie de funciones que responden a la solución propuesta.

Se definieron además, dos algoritmos fundamentales que responden a los subprocesos mencionados anteriormente. A continuación se presentan de manera detallada.

El algoritmo encargado de encontrar todos los ítemsets frecuentes se denomina "*GetFrequentKItemsets"* este algoritmo recibe como parámetros los ítemsets candidatos de tamaño k y el soporte mínimo. Se recorren todos los ítemsets candidatos retornando aquellos que cumplan con el soporte definido por el usuario, como se muestra a continuación:

```
private Hashtable GetFrequentKItemsets(Hashtable candidates, double minSupport) {
      Hashtable<String, Double> frequentItemsetsReturn = new Hashtable<>();
      for (int i = candidates.size() - 1; i >= 0; i--) {
         String item = (String) candidates.keySet().toArray()[i];
         double support = Math.rint((candidates.get(item) / idLastTransaction) * 100) / 100;
         if ((support >= minSupport)) {
            frequentItemsetsReturn.put(item, support);
            allfrecuentItemsets.put(item, support);
         }
      }
      return frequentItemsetsReturn;
   }
```

Figura 7. Algoritmo de generación de ítemsets frecuentes.

La mejora visible del subproceso de generación de ítemsets frecuentes se observa fundamentalmente en la generación de los ítemsets candidatos, específicamente en el cálculo del soporte de estos.

La función "*GenerateCandidates*" recibe como parámetros los ítemsets frecuentes obtenidos en la iteración anterior, es decir, los $k-1$ ítemsets, a partir de los cuales se obtendrán los nuevos candidatos de tamaño k como se muestra a continuación:

```
private Hashtable<String, Double> GenerateCandidates(Hashtable<String, Double> frequentKItemsets) {
Hashtable<String, Double> candidatesReturn = new Hashtable<>();
    for (int i = 0; i < frequentKItemsets.size() - 1; i++) {
        String firstItem = Alphabetize((String) frequentKItemsets.keySet().toArray()[i]);
       for (int j = i + 1; j < frequentKItemsets.size(); j++) {
          String secondItem = Alphabetize((String) frequentKItemsets.keySet().toArray()[j]);
          String generatedCandidate = GetCandidate(firstItem, secondItem);
          if (!generatedCandidate.isEmpty()) {
             generatedCandidate = Alphabetize(generatedCandidate);
             candidatesReturn.put(generatedCandidate, (double) 0);
          }
       }
    }
    Object[] candidates = candidatesReturn.keySet().toArray();
    candidatesReturn = GetSupports(candidates);
   return candidatesReturn;
 }
```

Figura 8. Algoritmo de generación de ítemsets candidatos.

Esta función crea los candidatos con soporte igual 0 y luego los envía al método "*GetSupports*", que se encarga de calcular el soporte de cada uno de los candidatos al mismo tiempo como se muestra a continuación:

```
private Hashtable<String, Double> GetSupports(Object[] candidates) {
double[] supp = new double[candidates.length];
Hashtable<String, Double> candReturn = new Hashtable<>();
    for (int i = 0; i < transactions.size(); i++) {
        for (int j = 0; j < candidates.length; j++) {
            String transAlp = (String) candidates[j];
            if (IsSubstring(transAlp, transactions.get(i + 1))) {
              supp[j]++;
            }
          }
       }
       for (int i = 0; i < candidates.length; i++) {
          candReturn.put((String) candidates[i], supp[i]);
      }
       return candReturn;   }
```

Figura 9. Algoritmo encargado de calcular los soportes de todos los ítemsets candidatos.

Al obtener todos los ítemsets frecuentes con sus soportes correspondientes, se aplica el segundo algoritmo, que responde al segundo subproceso del Apriori básico. La función "*GetStrongRules*" recibe como entrada la confianza mínima definida por el usuario y un conjunto de reglas generadas anteriormente, este conjunto solo contiene la mitad de las posibles reglas, aplicando la estrategia descrita en el epígrafe anterior. La función "*GetStrongRules*" se auxilia del método "*AddStrongRule*" para determinar cuáles reglas son fuertes para que sean estas las que se devuelvan, como se muestra a continuación:

```
private LinkedList<Rule> GetStrongRules(double minConfidence, LinkedList<Rule> rules) {
      LinkedList<Rule> strongRulesReturn = new LinkedList<>();
     for (int i = 0; i < rules.size(); i++) {
        String strXY = Alphabetize(rules.get(i).getCombination() + "," + rules.get(i).getRemaining());
        strongRulesReturn = AddStrongRule(rules.get(i), strXY, strongRulesReturn, minConfidence);
     }
      return strongRulesReturn;
  }
```

Figura 10. Algoritmo encargado de devolver reglas de asociación relevantes.

```
private LinkedList<Rule> AddStrongRule(Rule rule, String strXY, LinkedList<Rule> strongRulesReturn, double minConfidence) {
double confidence = GetConfidence(rule.getCombination(), strXY);
    Rule newRule;      boolean flag = true;
    if (confidence >= minConfidence) {
        newRule = new Rule(rule.getCombination(), rule.getRemaining(), confidence);
       strongRulesReturn.add(newRule);
        flag = false;
    }
     if (flag) {
       confidence = GetConfidence(rule.getRemaining(), strXY);
       if (confidence >= minConfidence) {
          newRule = new Rule(rule.getRemaining(), rule.getCombination(), confidence);
           strongRulesReturn.add(newRule);
       }
     }
   return strongRulesReturn;
 }
```

Figura 11. Algoritmo encargado de determinar reglas de asociación relevantes.

La secuencia de ejecución de estos algoritmos está dirigida por la función denominada "*Solution*" encargada de obtener finalmente las reglas de asociación fuertes y devolverlas al usuario. De manera general el pseudocódigo del algoritmo *Apriori-like* quedaría de la siguiente forma:

Apriori-like:

Entrada: D, I, minsup, minconf.

Salida: Conjunto de reglas de asociación con soporte y confianza ≥ que minsup y minconf.

Algoritmo:

1. Generar todos los ítemsets frecuentes con soporte ≥ minsup.
   - Generar los ítemsets frecuentes de tamaño $k = 1$.
   - Generar los ítemsets candidatos de tamaño $k + 1$.
   - Determinar el soporte de los ítemsets candidatos en una sola iteración sobre las transacciones.
   - Eliminar los ítemsets no frecuentes.
   - Repetir el proceso mientras se puedan generar nuevos candidatos.

2. Dado un ítemset frecuente $X = \{X_i \dots X_k\}$ con $K \geq 2$ generar todas las reglas de la forma $X \mid \{x_j\} \rightarrow \{x_j\}$, siendo el soporte de dicha regla el soporte de X y la confianza, el cociente entre el soporte de este y el soporte de $X \mid \{x_j\}$.
   - Generar todas las reglas con antecedente de tamaño uno.
   - Generar solo la mitad de las reglas posibles a partir del ítemset.
   - Eliminar las reglas no relevantes.
   - Repetir mientras existan ítemsets frecuentes sin procesar.

2.4. Pruebas funcionales

El funcionamiento de *Apriori-like* fue validado a partir de una prueba de caja negra con la cual se verifica si la solución desarrollada devuelve un grupo de reglas de asociación fuertes. Estas reglas se obtuvieron a partir de un conjunto de datos escritos en el formato establecido anteriormente. La muestra con la que se probó el algoritmo fue extraída de las bases de datos de la Aduana General de la República de Cuba. Dicha muestra fue procesada y escrita de acuerdo al formato definido con anterioridad para las entradas del algoritmo. Una vez procesada se contabilizó un total de 6221 ítems y 10000 transacciones. Con este volumen de datos, para valores mínimos de soporte y confianza de 86 y 94 porciento respectivamente, la ejecución de *Apriori-like* devolvió resultados de acuerdo a lo esperado en un tiempo de 4 minutos aproximadamente. La prueba fue desarrollada en una computadora con 4 Gb de memoria RAM y un procesador intelCore2Duo a 1.8 GHz. Los resultados obtenidos se muestran a continuación:

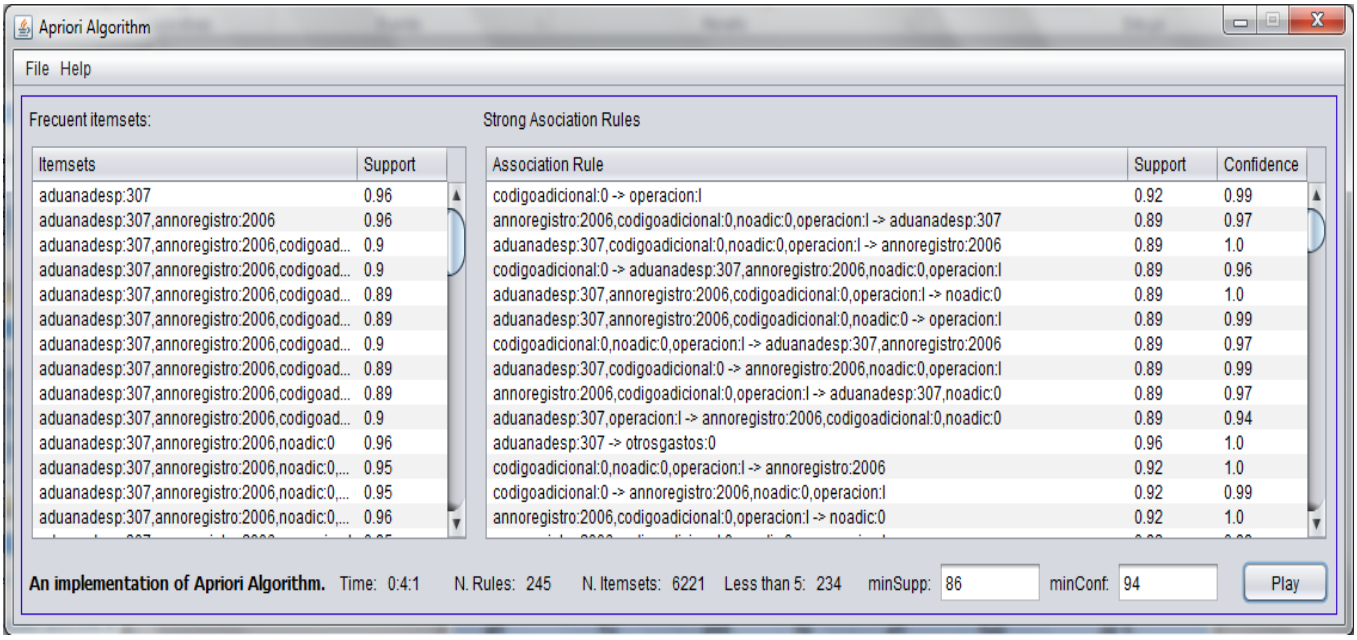


Figura 12. Resultados obtenidos al ejecutar Apriori-like.

2.5. Conclusiones Parciales

En el presente capítulo se muestra un ejemplo con el que se pretende clarificar el problema que representa la extracción de reglas de asociación en las bases de datos de la aduana. Se presenta una propuesta de solución que pretende modificar el algoritmo de extracción de reglas de asociación propuesto en (Agrawal, et al., 1994) bajo el nombre de Apriori. El Apriori básico, entre sus limitaciones, está diseñado para descubrir reglas a partir de datos numéricos, las bases de datos aduanales contienen datos tanto categóricos como numéricos. Debido a esto, los principales ajustes están enmarcados en la entrada del mismo así como su mecanismo de conteo de soporte y generación de reglas relevantes, con el objetivo de obtener resultados sobre la información referente a las declaraciones de mercancías almacenadas en las bases de datos aduanales. Se presenta la implementación de la solución propuesta, describiendo cada uno de los algoritmos desarrollados. Se valida dicho algoritmo a partir de pruebas de caja negra mostrando resultados satisfactorios en un tiempo aceptable.

3. CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN

3.1. Introducción

Existen una serie de patrones o métodos que permiten evaluar y validar los resultados alcanzados en una investigación. Entre ellos se encuentran la demostración, experimentación, simulación, uso de métricas, evaluación comparativa, razonamiento lógico y los modelos matemáticos (Vaishnavi , et al., 2008). A continuación se describen algunos de estos métodos que a criterio de los investigadores de este trabajo tienen mayor relevancia en las ciencias de la computación.

- **Demostración**

  Intenta demostrar que la solución es factible y válida para una o varias situaciones predefinidas. Es especialmente relevante cuando la demostración de una solución en sí misma se considera una contribución. Consta de dos momentos importantes, construir la solución o prototipo de solución que demuestre que esta es factible y demostrar que la solución construida es razonable para un conjunto predefinido de situaciones. Estas situaciones deben estar predefinidas y no se ha creado para adaptarse a la solución (Vaishnavi , y otros, 2008).

  La demostración puede mostrar las deficiencias de la solución. Por otra parte, puede mostrar que la solución es viable y aceptable. Si las situaciones de prueba están diseñadas apropiadamente, entonces la construcción de la solución y sus pruebas para estas situaciones pueden demostrar su validez, a pesar de que teóricamente constituyen el patrón de validación menos formal o débil. Ha sido utilizado en innumerables artículos científicos de la especialidad hasta el punto de ser el tipo de validación que más se utiliza desde el año 1999 (Shaw, 2002).

- **Experimentación**

  Intenta validar o rechazar un conjunto de hipótesis asociada a las afirmaciones acerca de la solución. Según (H. Sampieri, y otros, 1991) un experimento es un estudio de investigación en el que se manipulan deliberadamente una o más variables independientes para analizar las consecuencias de esa manipulación sobre una o más variables dependientes, dentro de una

situación de control para el investigador. La experimentación procede a establecer resultados asociados con la solución del problema de investigación en situaciones en que la recogida y el análisis de los datos son el único método factible de validación.

- **Simulación**

  Intenta validar la solución propuesta para el problema de investigación a través de un software de simulación. Consta de cinco momentos importantes, el primero de ellos es desarrollar el modelo conceptual del problema y su solución para que sea simulado en una computadora. Luego se desarrolla el conjunto inicial de datos de prueba, se selecciona la simulación diseñada específicamente para el dominio del problema. Se ejecuta dicha simulación para el conjunto de prueba elaborado con anterioridad y por último se demuestra la validez de la solución argumentando que las pruebas realizadas representan situaciones de la vida real (Vaishnavi , y otros, 2008).

- **Razonamiento Lógico**

  Utiliza la argumentación como forma de validación de la solución. Es una forma más débil de validación que el modelo matemático o el uso de experimentos. Consta de tres momentos importantes, identificación de las suposiciones (axiomas) relacionadas con el problema, identificación de las reglas (reglas de deducción) relacionadas con el problema o la solución y construcción de un camino lógico de las hipótesis (axiomas) a los planteamientos de la solución, utilizando las reglas de deducción que se identifiquen.
  Cuando los axiomas, reglas de deducción y las afirmaciones acerca de la solución se pueden afirmar con precisión; esta técnica constituye un modelo matemático para validar cualquier investigación siempre que no exista vaguedad en demostrar que las afirmaciones son consecuencia lógica de los axiomas (Vaishnavi , y otros, 2008).

- **Modelos Matemáticos**

  Intenta demostrar matemáticamente las afirmaciones acerca de la solución desarrollada. Consta de cuatro momentos importantes, expresar las afirmaciones acerca de la hipótesis de

> forma cuantitativa y precisa, convertir dichas afirmaciones para ser probadas como un teorema bien definido, demostrar los resultados auxiliares (lemas) que pueden ayudar a demostrar el teorema y por último demostrar el teorema de las afirmaciones, que pueden utilizar los lemas ya probados. Este modelo ofrece la forma más segura de validación de la solución. Esta validación es incluso más certera que la validación experimental (Vaishnavi , y otros, 2008).

La aplicación de estos patrones varía de acuerdo a su idoneidad y la seguridad con la que se puede establecer la validez de una solución. La demostración provee la forma más débil de la validación. Puede, sin embargo, ser adecuado si la solución es novedosa y resuelve un problema para el cual no existe ninguna solución previa. Por otro lado, las pruebas matemáticas constituyen la forma más certera de validación. La certeza del razonamiento lógico depende en gran medida de la precisión de sus argumentos y suposiciones. La experimentación y simulación son útiles cuando el problema es complejo y es inviable efectuar una demostración matemática. El uso de métricas y la evaluación comparativa, mecanismos para cuantificar afirmaciones respecto a una solución, son generalmente útiles cuando se utiliza en la experimentación y la simulación.

El uso de uno o varios métodos de validación depende en gran medida de las características del problema estudiado y de las convenciones aceptadas por la comunidad de investigadores como alternativas válidas de corroboración en la temática. La presente investigación utiliza el método de “Demostración” como mecanismo de validación. Este patrón se adapta correctamente a las características del trabajo siendo ideal para corroborar la validez del mismo pues no existen soluciones anteriores para este problema. La demostración en sí misma tiene valor práctico al estar compuesta por un prototipo funcional que permite construir modelos descriptivos aun cuando es necesario perfeccionar algunos detalles de usabilidad en dicho prototipo. Por último, la “Demostración” constituye el patrón de validación más utilizado en las publicaciones científicas para el área de las ciencias de la computación de acuerdo a los trabajos de (Shaw, 2002), (Cañete, 2002), (Shaw, 2003).

3.2. Recursos computacionales utilizados

Los recursos computacionales en esta investigación refieren las características de hardware y software de la computadora utilizada en el proceso de validación de la solución. Para efectuar el proceso de "Demostración" se utilizó una computadora ACPI Multiprocessor PC. Esta máquina cuenta con una motherboard Intel Rogers City DG965RY que incorpora un procesador DualCore Intel Core 2 Duo E4500 a 1.8 GHz y una memoria RAM DDR2 SDRAM Kingston 9905320-007.A00LF con 4 GB de capacidad. Los algoritmos propuestos fueron probados en una plataforma Microsoft Windows XP Professional publicada en el año 2002, a la que se le incorpora el paquete de corrección de errores Service Pack 3.

3.3. Conjuntos de datos utilizados

El conjunto de datos utilizados para efectuar la demostración fue recolectado de las bases de datos de la Aduana General de la República de Cuba. Específicamente, fueron seleccionados los datos referentes a las declaraciones de mercancías. Estos contienen elementos significativos para la obtención de modelos que permitan determinar las cargas con mayor riesgo de fraude en las operaciones de importación-exportación. Los datos fueron transformados en transacciones escritas en el formato definido para la entrada del algoritmo Apriori-like (epígrafe 2.3). A partir de estos se evaluó la cantidad de reglas obtenidas definiendo un máximo de 250 reglas como valor adecuado para la utilización por parte de los usuarios finales, este dato es empírico, no está validado y requiere atención en futuras investigaciones. La relevancia de estas reglas se determinó a partir del criterio de los especialistas en el tema y por último se tuvo en cuenta la cantidad de ítems de cada una de ellas, definiendo como reglas de fácil comprensión aquellas que contengan como máximo 5 elementos o ítems debido a que es la cantidad de elementos máximos que aparecen en las reglas que fueron clasificadas como relevantes por los especialistas. Este valor también tiene un alto grado de empirismo y la razón por la que se computa es que puede servir como mecanismo de optimización en trabajos futuros.

La muestra de datos obtenida a partir de las bases de datos de la aduana consta de 10000 tuplas y 33 columnas. Se determinó que fuera 10000 el número de las transacciones puesto que muchas de las

soluciones publicadas para extraer reglas de asociación (Li, et al., 2009), (Agrawal, et al., 1993), (Yu-Lu, et al., 2009), (Yueqin, et al., 2009) utilizan valores semejantes para probar su validez. Además, con este volumen de datos se puede determinar el buen funcionamiento del algoritmo y obtener reglas de asociación relevantes para el dominio estudiado.

Teniendo en cuenta la heurística para la cual fue diseñado el algoritmo Apriori y el formato (epígrafe 2.3) requerido por el algoritmo Apriori-like, es necesario transformar cada una de las columnas en la sucesión de sus distintos valores, obteniendo como resultado aproximadamente 6200 nuevas columnas que formarían el conjunto I que contiene todos los ítems. Por otro lado al transformar las tuplas solo cambiaría el formato de escritura sin que se afecte el número de las mismas. Como resultado final, se obtienen conjuntos de entrada para el algoritmo Apriori-like de longitud igual a 6221 en el caso del conjunto de ítems I y 10000 en el caso de la base de transacciones D. Además se utilizarán valores de soporte y confianza mínimos, que en la práctica cotidiana, serán definidos por el usuario. En el presente caso de pruebas esos valores serán variables, de manera que permitan obtener diferentes conjuntos de reglas.

Ítems	Transacciones	Soporte (%)	Confianza (%)
6221	10000	60	70
6221	10000	86	94
6221	10000	90	98
6221	10000	87	74

Tabla 5. Especificación de datos utilizados.

3.4. Discusión de los resultados

Al aplicar el algoritmo Apriori-like utilizando los datos descritos anteriormente se obtuvo un grupo variable de reglas para cada combinación de los umbrales de soporte y confianza. A partir del total de reglas generadas para cada caso, se puede determinar con cuáles combinaciones de soporte y confianza el algoritmo devuelve resultados adecuados para el trabajo de los usuarios finales,

atendiendo al indicador mencionado anteriormente. A continuación se muestra la cantidad de reglas generadas para cada caso así como la clasificación del algoritmo de acuerdo a esta cantidad.

Soporte (%)	Confianza (%)	Total	Clasificación
60	70	5002	No adecuado
86	94	245	Adecuado
90	98	159	Adecuado
87	74	245	Adecuado

Tabla 6. Clasificación del algoritmo de acuerdo a la cantidad de reglas generadas.

Estas reglas de asociación fueron además presentadas a expertos de la Aduana General de la República de Cuba para que valoraran la relevancia de las mismas aplicadas a los procesos de importación-exportación. La revisión devolvió (para cada combinación de soporte y confianza planteados anteriormente) un grupo de reglas relevantes, otras muy relevantes y otras no relevantes para este dominio. Además, se adicionó un indicador que permite conocer que tan efectivo se comportó el algoritmo para cada combinación de soporte y confianza. La efectividad (E) de la ejecución del algoritmo para cada combinación de soporte y confianza se calcula atendiendo al porciento que representa la suma de las reglas relevantes (r) y muy relevantes (R) con respecto al total de reglas generadas (T). De la siguiente forma:

$$E = \frac{(r + R) * 100}{T}$$

La tabla que se presenta a continuación muestra la efectividad del desempeño del algoritmo para cada variación de soporte y confianza. Además, se especifica la cantidad de reglas obtenidas por cada clasificación determinada por los especialistas del Centro de Dirección y Automatización de la Información, perteneciente a la Aduana General de la República de Cuba.

Entrada		Salida				
Sop. (%)	**Conf. (%)**	**No Rel.**	**Rel.**	**Muy Rel.**	**Total**	**Efectividad (%)**
60	70	4537	292	173	5002	9,3
86	94	49	113	83	245	80
90	98	76	49	34	159	52,2
87	74	127	76	42	245	48,2

Tabla 7. Clasificación de las reglas de asociación según su relevancia.

De acuerdo a la tabla anterior, los valores de soporte y confianza con mayor efectividad en la extracción de reglas relevantes y muy relevantes para el dominio de las declaraciones de mercancías son 86 y 94 porciento respectivamente. Es importante destacar que estos valores pueden variar para conjuntos de datos diferentes e incluso para diferentes expertos en este tipo de procesos puesto que la evaluación depende del criterio personal de los mismos. A continuación se presentan algunas de las reglas consideradas muy relevantes:

Reglas de asociación muy relevantes	Sop. (%)	Conf. (%)
aduanadesp:307, codigoadicional:0, noadic:0, otrosgastos:0 → clasificacion:false	88	98
aduanadesp:316, operacion:I, noadic:0 → clasificacion:false, valorfactura:500	92	100
codigoadicional:0, valorfactura:75, preciounit:4 → clasificacion:false, pesoneto:29	88	96
aduanadesp:307 → clasificacion:false	92	96

Tabla 8. Muestra de reglas de asociación muy relevantes a criterio de especialistas.

Las reglas mostradas en la tabla anterior difieren en cuanto a la cantidad de ítems o elementos por los que están compuestas. Atendiendo a esta cantidad de elementos las reglas pueden clasificarse en fáciles o difíciles de comprender, como se menciona anteriormente. Aquellas reglas cuya cantidad de elementos que la componen es menor o igual que 5, se consideran de fácil comprensión para los usuarios finales. A continuación se muestran las reglas obtenidas clasificadas de acuerdo al criterio anteriormente expuesto.

Entrada		Salida		
Sop. (%)	**Conf. (%)**	**Difícil Comprensión**	**Fácil Comprensión**	**Total**
60	70	17	4985	5002
86	94	11	234	245
90	98	0	159	159
87	74	11	234	245

Tabla 9. Clasificación de las reglas de asociación según la cantidad de ítems.

Al observar los resultados se puede apreciar que en la mayoría de los casos todas las reglas descubiertas fueron de fácil comprensión, por tanto los itemsets frecuentes generados no tuvieron un tamaño mayor a 5 elementos.

3.5. Conclusiones Parciales

El presente capítulo presenta una serie de patrones o métodos utilizados para validar los resultados de una investigación científica. Se determinó que el patrón “Demostración” es el adecuado para realizar las pruebas de validación de la solución presentada. Se definieron los elementos de la demostración desarrollada así como los recursos computacionales con que se efectuaron las pruebas.

Para demostrar el correcto funcionamiento del algoritmo, se varió el soporte y confianza sobre un conjunto de datos extraído de las bases de datos de la Aduana General de la República de Cuba. Los resultados obtenidos demuestran que Apriori-like obtiene reglas de asociación relevantes para la clasificación de riesgo de fraude aduanal. Estos resultados influyeron en la efectividad de la salida del algoritmo implementado.

Se determinó que si el soporte y la confianza son demasiado altos, se pierden reglas, que según el criterio de los especialistas son relevantes para el dominio estudiado. Por otro lado, si son muy bajos el algoritmo devuelve un número de reglas que hace bastante engorroso el proceso de determinar cuáles son relevantes, razón por la que la efectividad disminuye considerablemente. Por último, se clasificaron las reglas de acuerdo a la cantidad de elementos que la conforman determinando que en su mayoría son de fácil comprensión.

CONCLUSIONES

En el presente trabajo se implementó un algoritmo para la extracción de reglas de asociación en conjuntos de datos pertenecientes a las declaraciones de mercancías en la Aduana General de la República de Cuba. El referido algoritmo sigue los supuestos teóricos establecidos en el diseño del algoritmo Apriori y realiza un grupo de mejoras enfocadas a la tipología específica de los conjuntos de datos bajo estudio, dentro de ellas se destacan las siguientes:

- Se ajustó el algoritmo para que pueda trabajar con datos categóricos en lugar de binarios.
- Se modificó la expresión para el conteo de soporte de forma tal que se reduce significativamente la cantidad de lecturas sobre la base de datos.
- Se ajustó el mecanismo de conformación de reglas a partir de los itemsets frecuentes de manera que se generan inicialmente solo la mitad, basados en la propiedad de intercambio del antecedente por el consecuente.

Los resultados alcanzados permiten concluir que:

1. Las reglas de asociación permiten obtener modelos significativos que pueden ser empleados en la clasificación de riesgo de fraude aduanal de las mercancías.
2. Es posible obtener un conjunto de reglas de asociación relevantes para esta tarea a partir de la aplicación del algoritmo diseñado.
3. La cantidad de reglas y efectividad del algoritmo dependen en gran medida de los valores de soporte y confianza determinados por los usuarios.
4. El volumen total de reglas generadas es muy grande y debe ser tratado para reducirlo en función de su relevancia.

RECOMENDACIONES

Para el mejoramiento de este trabajo es importante optimizar un grupo de elementos que no fueron tomados en cuenta por cuestiones de tiempo, dentro de ellos señalamos los siguientes, debido a su relevancia y a que forman parte del trabajo futuro que se desarrollará en la temática:

1. Establecer un mecanismo de ponderación de las reglas.
2. Determinar un volumen de reglas manejables por los especialistas humanos.
3. Trabajar en el perfilado de los datos de la aduana para ajustar la entrada del algoritmo.

BIBLIOGRAFÍA

Delgado, M., Sánchez, D. y Vila, M. A. 2000. *Fuzzy cardinality based evaluation of quantified sentences* s.l. : International Journal of Approximate Reasoning, 2000.

Delgado, M., y otros. 2005. *Mining fuzzy association rules: an overview.* s.l. : Soft Computing fo Information Processing and Analysis, 2005.

A. Savasere, E. Omiecinski, and S.Navathe. 1995. *An eficient algorithm for mining association rules i large.* Atlanta : Technical Report GIT-CC-95-04, Institute of Technology, 1995.

Agrawal R., Srikant R. 1994. *Fast Algorithms for Mining Association Rules.* 1994.

Agrawal, R y Srikant, R. 1994. *Fast algorithms for mining association rules.* s.l. : Proceedings of the 20th 1994.

Agrawal, R, Imielinski, T y Swami, A. 1993. *Mining Associations between sets of items in massive databases.* ACM-SIGMOD International Conference on Data : s.n., 1993.

Brin, S, y otros. 1997. *Dynamic itemset counting and implication rules for market basket data.* s.l. SIGMOD, 1997.

Cañete. 2002. *¿Qué se entiende, en España, por Investigación en Ingeniería del Software?* s.l. : MIFISIS 2002.

Chen, G, y otros. 2002. *Simple association rules (SAR) and the SAR-based.* s.l. : Computers & Industria Engineering, 2002.

Chena, Y y Weng, C.H. 2008. *Mining association rules from imprecise ordinal data. Fuzzy.* 2008.

Chung., J. D. Holt and S. M. 2001. *Multipass algorithms for mining association rules in text databases* s.l. : Knowledge and Information Systems, Springer-Verlag, 2001.

Clark, P. y Boswell, R. 2000. *Data Mining. PracticalMachine Learning Toolsand Techniques.* s.l. : Morgai Kaufmann Publishers, 2000.

Comisión, económica para américa latina y el caribe. 2006. *Indices de comercio exterior de Cuba* 2006.

Delgado, M., y otros. 2003. *Fuzzy association rules: general model and applications.* s.l. : Fuzzy Systems IEEE Transactions on, vol. 11, no. 2, 2003.

Delgado, Miguel, Ruiz, M. Dolores y Sánchez, Danel. 2008. *Reglas de asociación difusas: Nuevo Retos.* Granada : ESTYLF08, Cuencas Mineras, 2008.

Delgado, Miguel, y otros. 2003. *Fuzzy Association Rules: General Model.* s.l. : IEEE TRANSACTIONS ON FUZZY SYSTEMS, 2003.

Fayyad, U M, y otros. 1996. *Advances in Knowledge Discovery and Data Mining.* Cambridge, MA : AAA Press and MIT Press, 1996.

Frawley, W., Piatetsky-Shapiro, G y Matheus. 1992. *Knowledge discovery in databases: An overview* s.l. : AAA/MIT Press, 1992.

Gyenesei, A. 2001. *A fuzzy approach for mining quantitative.* s.l. : Acta Cybern, 2001.

H. Sampieri, C. Roberto, Fernández Collado, Carlos y Baptista Lucio, Pilar. 1991. *Metodología de la investigación.* s.l. : McGRAW - HILL INTERAMERICANA DE MÉXICO, 1991.

Han, Jiawei y Fu, Yongjian. 1995. *Discovery of Multiple-Level Association Rules from Large Databases* British Columbia : School of Computing Science Simon Fraser University, 1995.

Hernández León, Raudel, y otros. 2010. *Descubrimiento de conjuntos frecuentes de items en datos estaticos y dinamicos.* La habana : Dpto. Minería de Datos, Centro de Aplicaciones de Tecnología de Avanzada (CENATAV), 2010.

Jimenez Ruiz, Maria Dolores. 2010. *Modelado formal para la representación y evaluación de reglas de asociación.* Granada : Departamento de ciancias de la computación e inteligencia artificial, 2010.

Kuok, C.M, Fu, A.W. y Wong, M.H. 1998. *Mining fuzzy association rules in databases.* s.l. : SIGMOD Record, 1998.

Laleh, Naeimeh y Abdollahi Azgomi, Mohammad. 2009. *A Taxonomy of Frauds and Fraud Detection Techniques.* Tehran : Department of Computer Engineering, 2009.

Li, Jiye y Cercone, Nick. 2009. *Introducing A Rule Importance Measure.* Canada : s.n., 2009.

Li, YingJiu, y otros. 2003. *Discovering calendar-based temporal association rules.* s.l. : Knowledge-Based Systems, 2003.

Medina Pagola, José E., y otros. 2007. *Generación de conjuntos de items y reglas de asociación.* La Habana : Dpto. Minería de Datos, Centro de Aplicaciones de Tecnología de Avanzada (CENATAV), 2007.

—. 2007. *Generación de conjuntos de items y reglas de asociación.* La Habana : Dpto. Minería de Datos Centro de Aplicaciones de Tecnología de Avanzada (CENATAV), 2007.

Mesa Rodriguez, Alejandro, y otros. 2009. *Obtencion d conjuntos frecuentes usando computo reconfigurable.* La Habana : CENATAV, 2009.

Motoda, Hiroshi y Ohara, Kouzou. 2009. *Apriori.* s.l. : Taylor & Francis Group, 2009.

OMA, Organización Mundial de Aduanas. 1997. *Convenio de Kyoto - Directivas de control aduanero* Kyoto : http://www.wcoomd.org/Kyoto/20Sp/cap6, 1997.

Orallo Hernández, José, Quintana Ramírez, Ma José y Ferri Ramírez, Cesar. 2004. *Introducción a la Minería de Datos.* Madrid : Pearson Educación S.A., 2004.

Ramakrishnan, Srikant y Agrawal, Rakesh. 1995. *Mining Generalized Asociation Rules.* Zurish : IBM Almaden research center, 1995.

Sánchez, D. 1999. *Adquisición de relaciones entre atributos en bases de datos.* Granada : Dept. Comput Sci. Artificial, 1999.

Shaw, M. 2002. *What makes good research in software engineering.* s.l. : FOR TECHNOLOGY TRANSFER (STTT). SPRINGER BERLIN / HEIDELBERG, 2002.

—. 2003. *Writing good software engineering research papers: minitutorial.* Washington : Proceedings o the 25th International Conference on Software Engineering, ICSE, 2003.

Silverstein, C, Brin, S y Motwani, R. 1998. *Beyond market baskets: Generalizing association rules to dependence rules.* s.l. : Data Mining Knowl. Disc., 1998.

Sriphaew, Kritsada y Theeramunkong, Thanaruk. 2002. *A New Method for Finding Generalized Frequent Itemsets in Generalized Association Rule Mining.* Thailand : Information Technology Program Sirindhorn International Institute of Technology, Thammasat University, 2002.

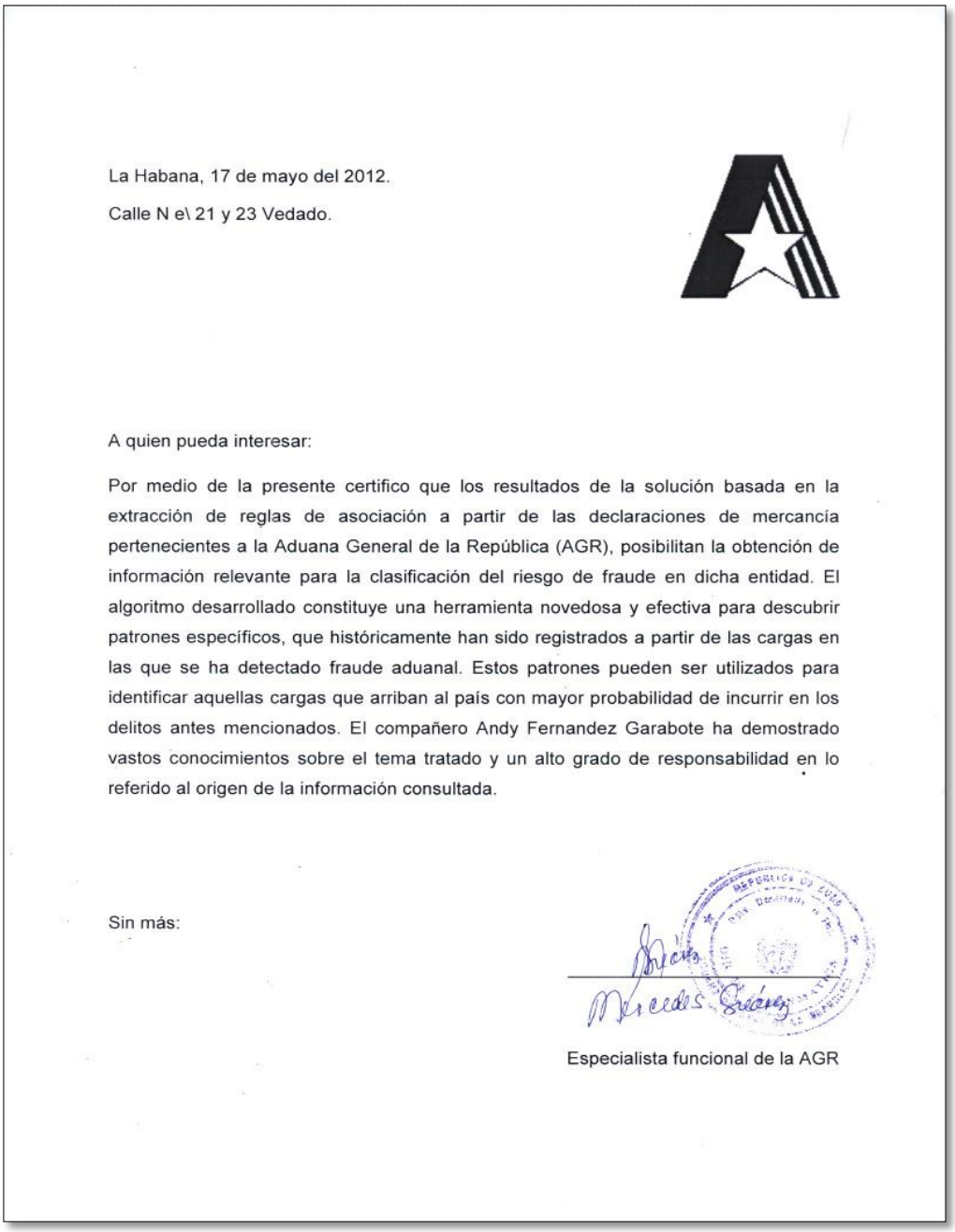
Triantaphyllou, Evangelos. 2010. *Data Mining and Knowledge Discovery via Logic-Based Methods* Louisiana : Springer, 2010.

Vaishnavi , Vijay K. y Kuechler Jr., William. 2008. *Design Science Research Methods and Patterns: Innovating Information and Communication Technology.* NewYork : Auerbach Publications is an imprint o the , an informa business , 2008.

Yueqin, Zhang, Qingwei, Yan y Lili, Zong. 2009. *An Association Rule Algorithm Based on Quotien Space.* Taiyuan : College of Computer and Software Taiyuan University of Technology, 2009.

Yu-Lu, LIU, y otros. 2009. *An Efficient Algorithm of Mining Association Rules Based on Digital Pure Subset.* Chongqing : College of Math and Computer Science, Chongqing Three Gorges University, 2009.

ANEXOS



La Habana, 17 de mayo del 2012.

Calle N e\ 21 y 23 Vedado.

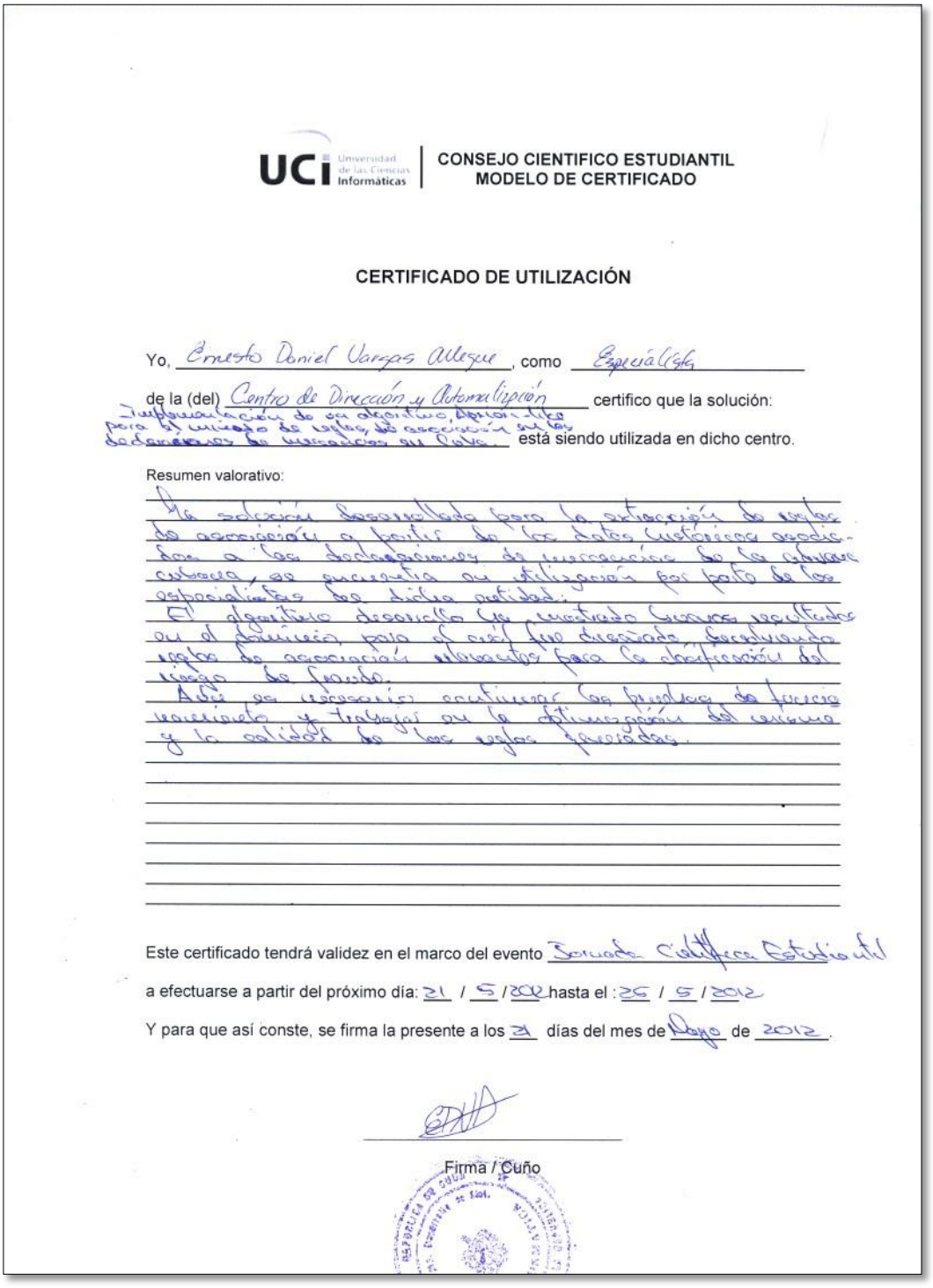
A quien pueda interesar:

Por medio de la presente certifico que los resultados de la solución basada en la extracción de reglas de asociación a partir de las declaraciones de mercancía pertenecientes a la Aduana General de la República (AGR), posibilitan la obtención de información relevante para la clasificación del riesgo de fraude en dicha entidad. El algoritmo desarrollado constituye una herramienta novedosa y efectiva para descubrir patrones específicos, que históricamente han sido registrados a partir de las cargas en las que se ha detectado fraude aduanal. Estos patrones pueden ser utilizados para identificar aquellas cargas que arriban al país con mayor probabilidad de incurrir en los delitos antes mencionados. El compañero Andy Fernandez Garabote ha demostrado vastos conocimientos sobre el tema tratado y un alto grado de responsabilidad en lo referido al origen de la información consultada.

Sin más:

Especialista funcional de la AGR

Anexo 1. Aval proporcionado por los especialistas de la Aduana General de la República de Cuba.



UCI Universidad de las Ciencias Informáticas | CONSEJO CIENTIFICO ESTUDIANTIL
MODELO DE CERTIFICADO

CERTIFICADO DE UTILIZACIÓN

Yo, Ernesto Daniel Vargas Allesque, como Especialista

de la (del) Centro de Dirección y Automatización certifico que la solución:

Implementación de un algoritmo Apriori-like para el minado de reglas de asociación en las declaraciones de mercancías en Cuba. está siendo utilizada en dicho centro.

Resumen valorativo:

La solución desarrollada para la extracción de reglas de asociación a partir de los datos históricos asociados a las declaraciones de importación de la aduana cubana, así como su utilización por parte de los especialistas de dicha entidad.
El algoritmo desarrollado fue evaluado con resultados en el dominio para el cual fue diseñado, generando reglas de asociación relevantes para la clasificación del riesgo de canal.
Además es necesario acentuar las bondades de fácil usabilidad y trabajar en la diferenciación del sistema y la validez de las reglas generadas.

Este certificado tendrá validez en el marco del evento Jornada Científica Estudiantil

a efectuarse a partir del próximo día: 21 / 5 / 2012 hasta el : 25 / 5 / 2012

Y para que así conste, se firma la presente a los 21 días del mes de Mayo de 2012.

Firma / Cuño

Anexo 2. Certificado de utilización proporcionado por el Centro de Dirección y Automatización de la Información.