Universidad de las Ciencias Informáticas Facultad 3

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

Título: Base de conocimiento para la recomendación de algoritmos de descubrimiento de procesos.

Autor: Osiel Fundora Ramírez

Tutores: Ing. Damián Pérez Alfonso

Dr. Raykenler Yzquierdo Herrera

La Habana

Junio 2014

Declaratoria de autoría

Declaro ser autor del presente trabajo de diploma y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Autorizo a dicho centro para que haga el uso que estime pertinente con este trabajo.

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

Osiel Fundora Ramírez

_____

Firma del autor.

Ing. Damián Pérez Alfonso

_____

Dr. Raykenler Yzquierdo Herrera

_____

Firma de los tutores

Agradecimientos

A mi madre por darme todo su amor, por ser ejemplo de sacrificio y dedicación por enseñarnos tantas cosas lindas y por darnos todo lo que somos yo y mi hermana.

A mi hermanita por ser la personita más importante en mi vida, por todo su cariño y comprensión, por esperarme siempre aunque lleve tanto tiempo lejos de ella.

Mis tíos Jesús, Jorge y Gine por quererme como un hijo más, por sus enseñanzas y todo su amor.

A mis primos que más que primos son hermanos Riudi, Nailem, Jorgito, Somy y Yannielis por estar ahí en todo momento por ser los mejores primos del mundo.

A mis hermanos Alexito y Daniel Alejandro.

A Danay, Acela, Nela, Noel y Gilberto por acogerme como un hijo y darme todo su apoyo.

A Babi por ser especial y permitirme ser parte de su vida.

A David, Mabel y Mabelita por todo el cariño y apoyo, por hacerme sentir un miembro más de su familia.

A Dayamí por ser única, por todos los momentos especiales.

A Damián por ser más que un tutor, ser un amigo, por enseñarme tantas cosas y por la confianza.

A Raykenler por sus sabios consejos y su apoyo incondicional.

A Mayi por su cariño, sus consejos y sus libros.

A Eudel y Reinaldo por ser más que compañeros, ser amigos y convertir la minería de proceso en una forma de vida, en una gran familia.

A Tomás y Yohery por ser mis hermanos de corazón, mis amigos no importa la distancia y las adversidades.

A Claudia por todos los momentos vividos.

A Daylén por su amistad.

A Jessie, Gisela, Susana, Elizabeth, Marile, Alejandro, Camilo, Dianelys, Mercedes, Barbara, Lisandra, Dianiselys, Fernando, Nardelys, Raisa,Martha por estar siempre ahí, por su ayuda por soportarme estos largos años y no dejarme claudicar.

A Giselle por todas las horas y su confianza.

A mis compañeros del aula, a los compañeros del proyecto, a mis amigos que ya no están aquí.

A todos los que confiaron en mi aquí y en Santa Clara.

A todos muchas gracias

Dedicatoria

A las dos personas más importantes de mi vida y que hoy no se encuentran físicamente a mi lado pero que son un ejemplo a seguir. A mis padres Ignacio y Luis por todas sus enseñanzas y comprensión.

A ustedes va dedicado este trabajo con todo el respeto y admiración del mundo.

Resumen

La minería de proceso es la disciplina que permite descubrir, monitorear y mejorar procesos a través del análisis de registros de eventos. Para recomendar un algoritmo debe considerarse el impacto de las características de los registros de eventos y del proceso. Actualmente el conocimiento disponible sobre el tema es dominado por expertos y se encuentra disperso, por lo que esta investigación se propone contribuir a la recomendación de algoritmos de descubrimiento teniendo en cuenta el conocimiento disponible en esta disciplina. Para ello se desarrolla una base de conocimiento utilizando ProM como herramienta para el descubrimiento y la integración de la base de conocimiento; CoBeFra para la evaluación de modelos de procesos descubiertos; Eclipse como entorno de desarrollo; Generador de Log para generar los registros de eventos; Base de datos objetual como tecnología para la persistencia de la base de conocimiento y Berkeley como gestor de base de datos. Como resultado se obtiene un conjunto de casos que conforman la base de conocimiento combinando características de entornos reales. Contiene información relevante de las características que afectan el rendimiento de los algoritmos de descubrimiento, la evaluación de los modelos de procesos y los algoritmos. Además, recoge los rasgos necesarios, para aplicar técnicas de clasificación, para la recomendación de algoritmos de descubrimiento. El complemento desarrollado permite gestionar los casos de la base de conocimiento. La comparación de los resultados de la recomendación al utilizar la clasificación sobre la base de conocimiento y la evaluación empírica mostró mejores tiempos para la técnica de clasificación.

Palabras claves: minería de proceso, recomendación de algoritmos de descubrimiento, base de conocimiento.

Índice

Índice de ilustraciones

Índice de tablas

Introducción

En la actualidad los procesos de negocios (BP, por sus siglas en inglés) centran la atención de la comunidad científica. Las empresas utilizan sistemas de información que son capaces de gestionar sus procesos de negocio, trayendo múltiples beneficios entre los que se pueden encontrar: la automatización de los procesos operativos, suministro de una plataforma de información necesaria para la toma de decisiones y usabilidad de la información (Kourdi 2008). Estos sistemas son importantes para mantener la competitividad y el desarrollo de las organizaciones.

De forma general los sistemas de información poseen la capacidad de registrar en forma de trazas la ejecución de los procesos que realizan. El archivo donde se almacenan estas trazas se denomina registro de eventos. Cada entrada del registro de eventos contiene información relacionada con las instancias ejecutadas, esta información queda registrada en formato XES (Mans, Schonenberg, Song, Aalst, Bakker 2009). A partir de los registros de eventos almacenados por estos sistemas se pueden aplicar distintas técnicas de minería de procesos.

La minería de procesos es un área de investigación, que permite realizar análisis de los procesos basado en su funcionamiento real. Tiene como beneficios el descubrimiento de modelos representativos de la realidad, la detección de desviaciones, el descubrimiento de redes sociales y modelos organizacionales relacionados con el proceso que se está analizando (Van Der Aalst, Dustdar 2012). La minería de procesos incluye su diagnóstico, descubrimiento, el chequeo de conformidad y el soporte operacional. El descubrimiento de procesos se realiza a partir de los registros de eventos (AALST, 2011; AALST and WEIJTERS, 2004).

El descubrimiento de modelos representativos del proceso es el área a la que se le ha prestado mayor atención dentro de la minería de procesos. El descubrimiento de los modelos de procesos consiste en la representación de la información presente en un registro de eventos. Para ello se utilizan algoritmos de descubrimiento. Un algoritmo de descubrimiento es una función que mapea un registro de eventos y transforma esa información en un modelo de procesos (de Leoni, van der Aalst 2013).

El **ruido**, las **tareas ocultas** (ausencia de información) y **las tareas duplicadas** son

características de los registros de eventos que afectan a los algoritmos de descubrimiento (De Weerdt, De Backer, Vanthienen, Baesens 2012). La presencia de patrones de control de flujo como **lazos** y **alternativas no libres** así como la **heterogeneidad de los casos** y la **granularidad** de los eventos son características del proceso que también afectan el descubrimiento (Jan Claes, Geert Poels 2012a). Además los modelos descubiertos deben presentar un balance entre cuatro criterios de **calidad**: simplicidad, generalización, aptitud y precisión (Ly, Indiono, Mangler, Rinderle-Ma 2012).

La elección incorrecta de un algoritmo de descubrimiento conduce a demoras en el proceso de descubrimiento. Esto aumenta el costo de la aplicación de la minería de procesos, limita sus potencialidades y aumenta las probabilidades de obtener un modelo de proceso de baja calidad. Lo cual a su vez puede dificultar la aplicación otras técnicas de minería de procesos.

Con el fin de identificar los algoritmos de descubrimiento que permitan obtener modelos de proceso de mejor calidad ante ciertas situaciones, se han desarrollado un conjunto de técnicas para la evaluación de los algoritmos. Una forma de evaluar la calidad de los algoritmos es a partir de la calidad de los modelos que descubren.

Un enfoque para realizar la evaluación de los algoritmos de descubrimiento es el empírico. Este enfoque es poco adecuado ya que resulta costoso desde el punto de vista computacional y de tiempo. El tiempo necesario para la evaluación empírica de algoritmos de descubrimiento, está determinado por el tiempo de ejecución de cada uno de los algoritmos a evaluar y el tiempo de ejecución de las métricas que se utilicen. Un artículo publicado sobre evaluación empírica muestra que los tiempos necesarios para evaluar 8 registros de eventos reales, con 8 métricas, oscilan entre 22 minutos y 23 días para el algoritmo Genetic Miner. Al tiempo de ejecución se debe agregar el de configuración de los parámetros y las métricas a utilizar (Wang, Wong, Ding, Guo, Wen 2012).

Un estudio realizado en el año 2013 (Seppe K.L.M. vanden Broucke, Cédric Delvaux 2013) describe el impacto que poseen las características en los algoritmos y su influencia por cada dimensión de calidad. Esta publicación muestra la diversidad de influencias entre las características de los procesos, los registros de eventos, los algoritmos de descubrimiento con las dimensiones de calidad. Tener en cuenta estos factores para utilizar un algoritmo permite la obtención de modelos de proceso que se ajusten mejor a los entornos donde se aplique el descubrimiento. Además disminuiría el tiempo de realización del descubrimiento.

A partir de la situación problémica planteada se define el **problema a resolver**:

¿Cómo contribuir a la recomendación de algoritmos de descubrimiento de procesos teniendo en cuenta el conocimiento disponible en la minería de procesos?

Por lo cual el **objeto de estudio** de la investigación es la minería de procesos

Teniendo en cuenta el problema a resolver se define como **objetivo general:** desarrollar una base de conocimiento para la recomendación de algoritmos de descubrimiento de procesos teniendo en cuenta el conocimiento disponible en la minería de procesos.

Por tanto, el **campo de acción** está enmarcado en el descubrimiento de procesos

Objetivos específicos:

- Analizar las características de los procesos y registros de eventos, que afectan los algoritmos de descubrimiento.
- Describir las métricas para la evaluación de la calidad de los modelos descubiertos.
- Construir una base de conocimiento para la recomendación de algoritmos de descubrimiento.
- Desarrollar un complemento para la plataforma ProM que permita la gestión de la base de conocimiento construida.
- Validar la propuesta de solución mediante la aplicación de técnicas de clasificación utilizando la base de conocimiento.

Tareas de investigación:

1-Análisis de las características de los registros de eventos que afectan el rendimiento de los algoritmos de descubrimiento.

2-Caracterización de las métricas que permiten evaluar la calidad de los modelos descubiertos.

3-Selección de las herramientas que permiten la creación de registros de eventos con características que afectan el descubrimiento de procesos.

4-Selección de la tipología de base de conocimiento a utilizar a partir del estudio de las alternativas existentes y el conocimiento a almacenar.

5-Construcción de una base de conocimiento para la recomendación de los algoritmos de descubrimiento.

6-Implementación de un complemento para integrar la base de conocimiento con la plataforma ProM.

7-Evaluación de la solución propuesta.

Para el desarrollo de la investigación se utilizaron los siguientes métodos teóricos:

- Histórico-lógico.
- Hipotético-deductivo.
- Analítico-Sintético.

El método **histórico-lógico** permite dirigir la problemática planteada en la investigación asociada al descubrimiento de procesos. En la primera parte de la investigación se realiza un estudio del estado del arte asociado a las características de los registros de eventos y los algoritmos de descubrimiento, estableciendo cómo inciden las características de los primeros en los segundos. También se analizan las diferentes estructuras que puede poseer una base de conocimiento para seleccionar la más adecuada.

La investigación sigue un método **hipotético-deductivo**, pues a partir del problema trazado, se plantean objetivos específicos los cuales, en el transcurso de la investigación, son resueltos siguiendo métodos científicamente fundamentados.

El método **analítico-sintético** se utiliza para descubrir los elementos que componen la naturaleza o esencia asociada al fenómeno del descubrimiento de proceso. Se definen las causas y los efectos, para posteriormente integrar los elementos en una nueva unidad, en una comprensión total de la esencia de lo que se conoce en todos sus elementos y particularidades.

Además de los métodos teóricos se utilizaron los métodos empíricos **experimentación** y **medición**. Mediante la experimentación se crearon los registros de eventos combinando las características seleccionadas para conformar los casos de la base de conocimiento.

Para la medición se aplican pruebas estadísticas en el análisis de los registros de eventos generados. De esta forma se garantiza que los casos almacenados en la base de conocimiento contengan todas las características especificadas.

En el presente documento puede encontrarse un resumen, una introducción, tres capítulos y conclusiones. A continuación se resume el contenido de los capítulos:

Capítulo 1: Se define la minería de procesos, específicamente el descubrimiento de proceso y la recomendación de los algoritmos de descubrimiento. Se realiza un estudio crítico de las diferentes técnicas que permiten la recomendación de algoritmos de descubrimiento. Se analizan las diferentes métricas que existen para evaluar las cuatro dimensiones de calidad, así como las herramientas que se utilizan para la construcción de la base de conocimiento y el desarrollo del complemento para la gestión de la misma.

Capítulo 2: Se describe el diseño de la base de conocimiento. Se explica la metodología de experimentación y se describen las características que poseen los casos que componen la base de conocimiento. Se analizan las características que deben cumplir las métricas para ser utilizadas en la evaluación de los algoritmos de descubrimiento.

Capítulo 3: Se describe la implementación del complemento que gestiona la base de conocimiento, la cual se valida mediante el análisis de los resultados de la aplicación de la evaluación empírica y la clasificación, para realizar la recomendación de algoritmos de descubrimiento.

Capítulo 1: Fundamentación teórica

1.1 Introducción

En este capítulo se presentan los principales conceptos asociados a la minería de procesos y el descubrimiento de modelos de procesos. También se realiza un análisis de las características que afectan el rendimiento de los algoritmos de descubrimiento y las técnicas para recomendar los algoritmos. Se analizan las dimensiones de calidad y las métricas escogidas para cada una de estas. Se presentan las principales herramientas que serán utilizadas en la solución del problema.

1.2 Minería de Procesos

Las técnicas de minería de procesos, permiten extraer información no trivial y útil de los registros de eventos almacenados por los sistemas de información. Estas técnicas permiten realizar recomendaciones y predicciones, teniendo en cuenta el análisis de los datos actuales e históricos. Estas pueden clasificarse en tres grupos como se muestra en la Figura 1. El primer grupo engloba las destinadas al descubrimiento de modelo de procesos, el segundo representa el chequeo de conformidad (AALST et al. 2011) y el tercero abarca la extensión de un modelo existente. (AALST 2012).

Figura 1: Representación de los tres tipos fundamentales de técnicas de minería de proceso: descubrimiento, conformidad y mejoramiento (AALST 2012)

Las diferentes técnicas de minería de procesos permiten el descubrimiento de información a partir de los registros de eventos. De esta forma permiten conocer el flujo de ejecución de un proceso, las redes sociales y las métricas de desempeño (Aalst 2012). También permiten identificar cuellos de botella, prever problemas, registrar violaciones de políticas, recomendar contra medidas y simplificar procesos, además de disminuir los tiempos de diseño y con ello sus costos (Herrera, Castro, Cortés, Graña 2012).

Un elemento fundamental en la minería de procesos es el descubrimiento, es decir, la construcción automática del modelo de proceso asociado. Estos modelos descubiertos describen las dependencias causales entre las actividades del proceso (Aalst 2011).

1.3 Descubrimiento de procesos

El proceso de descubrimiento es una de las tareas más complejas y atendidas en la minería de procesos (van der Aalst 2012). Su objetivo es la construcción de un modelo de proceso a partir de la información obtenida de un registro de eventos (Aalst 2011). Los sistemas informáticos como Sistemas de Planificación de Recursos (ERP) y Sistemas de Gestión de Relaciones con los Clientes (CRM) son capaces de almacenar la información de los procesos de las entidades en registros de eventos. A la información almacenada en estos se le aplican técnicas de descubrimiento de procesos para obtener los modelos asociados.

El descubrimiento de un modelo de proceso requiere que el registro de eventos contenga información suficiente, es decir, posea un nivel de completitud tal que sus trazas sean representativas del comportamiento del proceso. El grado y tipo de completitud necesario para realizar el proceso de descubrimiento varía de un algoritmo a otro.

Para realizar el proceso de descubrimiento es necesario tener en cuenta el nivel de calidad que presentan las trazas. Existen 5 niveles de calidad de las trazas:

Nivel 5: Es el nivel más alto donde el registro de eventos es de excelente calidad (confiable y completo) y los eventos están bien definidos. Los eventos registrados poseen una semántica clara que implica la existencia de una o más ontologías. Los eventos y sus atributos se refieren a esta ontología. Ejemplo: registros de eventos anotados semánticamente de los sistemas BPM (Van Der Aalst, Adriansyah, De Medeiros, 2012).

Nivel 4: Los eventos se registran automáticamente y de manera sistemática y confiable. A diferencia de los sistemas operando a nivel 3, se da soporte de manera explícita a nociones tales como instancia de proceso (caso) y actividad. Ejemplo: los registros de eventos de los sistemas tradicionales de BPM/workflow (Van Der Aalst, Adriansyah, De Medeiros, 2012).

Nivel 3: Los eventos se registran automáticamente, pero no se sigue un enfoque sistemático para registrarlos. El registro de eventos es confiable pero no necesariamente completo. Aunque se necesita extraer los eventos de una variedad de tablas, se puede asumir que la información es correcta. Ejemplo: las tablas en un sistema ERP o los registros de eventos de sistemas CRM (Van Der Aalst, Adriansyah, De Medeiros, 2012).

Nivel 2: Los eventos se registran automáticamente como un subproducto de algún sistema de información. Es posible pasar por alto el sistema de información. Por lo tanto, podrían faltar eventos o estos podrían no registrarse correctamente. Ejemplo: los registros de eventos de sistemas de gestión de documentos y productos o registros de errores de sistemas embebidos (Van Der Aalst, Adriansyah, De Medeiros, 2012).

Nivel 1: Los registros de eventos son de mala calidad. Los eventos registrados podrían no corresponder a la realidad y podrían faltar eventos. Ejemplo: trazas dejadas en documentos en papel que se trasladan a través de la organización, expedientes médicos en papel, etc (Van Der Aalst, Adriansyah, De Medeiros, 2012).

Además de los niveles de calidad de las trazas antes expuestos, los registros de eventos presentan características propias que afectan el descubrimiento. Estas características suelen ser más frecuentes en registros de eventos de baja calidad.

Entre las características propias de los registros de eventos se encuentran el ruido y las tareas duplicadas que se explican a continuación:

Ruido: Según (van der Aalst 2012) es el comportamiento raro e infrecuente presente en el registro de eventos y que no es representativo del comportamiento típico o común del proceso.

Tareas duplicadas: Son los eventos que se encuentran de forma reiterada en un registro de eventos y aunque poseen el mismo identificador, hacen referencia a actividades diferentes (Van der Aalst, 2012).

Entre las características propias de los procesos que inciden en los algoritmos de descubrimiento se encuentran las tareas ocultas, lazos y alternativas no libres que consisten en lo siguiente:

Tareas ocultas: Son actividades que no quedan reflejadas en el registro de eventos (van der Aalst 2012).

Lazos: La actividad o conjunto de actividades que se repite varias veces dentro del proceso (Wil M.P. van der Aalst 2010).

Alternativas no libres: Son constructores de control de flujo donde las alternativas y la concurrencia coinciden, de tal forma que una actividad **X** depende indirectamente de una actividad **Y**. Esto significa que en un punto de división o unión, la elección puede depender de alternativas escogidas en otras partes del proceso (Wil M.P. van der Aalst 2010).

Teniendo en cuenta estas características y haciendo una selección de los algoritmos de descubrimientos más utilizados en investigaciones y en entornos reales se construye la Tabla 1. Se han señalado las características que inciden en los algoritmos de descubrimiento y si estos responden de forma positiva o negativa a estas.

Nombre	Ruido	Tareas ocultas	Tareas duplicadas	Alternativas no libres	Lazos
EnhancedWFMiner	+	+	+	+	+
AGNEsMiner	+	+	+	+	+
DT Genetic Miner	+	+	+	+	+
HeuristicsMiner	+	+	-	+	+
FuzzyMiner	+	-	-	-	+
FSM Miner/Petrify	+	+	+	+	+
GeneticMiner	+	+	-	+	+

Tabla 1: Impacto de las características de los registros de eventos en los algoritmos de descubrimiento (Fundora-Ramírez Osiel 2013).

Tener en cuenta cómo estas características inciden en los algoritmos de descubrimiento es importante para obtener modelos de proceso de calidad. Aplicar un algoritmo de descubrimiento, que no maneje estas características en un registro de eventos donde esté presente alguna de ellas, implica que los modelos de proceso obtenidos pueden no reflejar el

comportamiento real del proceso. Obtener modelos con deficiencias dificulta la aplicación de otras técnicas de minería de procesos. Por ello se debe prestar atención al realizar el descubrimiento a estas características.

Un aspecto desafiante para los algoritmos de descubrimiento es obtener modelos de proceso con calidad. Un modelo de proceso que no cumpla con determinados requerimientos puede provocar que los análisis posteriores muestren resultados no acordes con el entorno donde se aplican. Además puede conducir a interpretaciones incorrectas del proceso que representa, afectando su comprensión y dificultando la identificación de aspectos a mejorar. En siguiente epígrafe se describen los métodos para evaluar la calidad en los modelos y las dimensiones existentes.

1.4 Dimensiones de calidad

Para evaluar la calidad de los modelos de proceso existen dos métodos, modelo-modelo y modelo-log. El método modelo-modelo evalúa la paridad entre el modelo descubierto y un modelo de referencia del proceso (Ailenei, Rozinat, Eckert, Aalst 2012). El método modelo-log compara el modelo descubierto con el registro de eventos para medir el nivel de coincidencia de las actividades representadas en el modelo con las almacenadas en el registro de eventos (Ailenei 2011).

Los métodos desarrollados para evaluar los modelos de procesos evalúan cada uno de ellos en cuatro (4) dimensiones de calidad que se describen a continuación:

Aptitud: El modelo no debe obviar ningún comportamiento presente en el registro de eventos (Ma 2012).

Precisión: El modelo no debe representar ningún comportamiento que no se aprecie en el registro de eventos. La generalización de un modelo puede conducir a un ajuste insuficiente, denominado en la literatura como *underfitting*, cuando esto sucede, pueden aparecer problemas de precisión. (Aalst, Rubin, Verbeek, Van Dongen, Kindler, Günther 2010).

Generalización: Es la capacidad del modelo descubierto de lograr una macro-representación del comportamiento observado en el registro de eventos.

Simplicidad: El modelo debe ser tan simple como sea posible (De Weerdt, Baesens, Vanthienen 2013).

Existen una serie de métricas que se encargan de evaluar los resultados de los algoritmos de descubrimiento. Estas métricas se corresponden con las dimensiones de calidad anteriormente descritas. Las métricas evalúan los modelos de procesos teniendo en cuenta los métodos modelo-modelo y modelo-log. Seleccionar las métricas adecuadas para evaluar los algoritmos es fundamental para obtener buenos resultados en la recomendación. Mientras mas cercano a 1 sea el valor de la métrica, mejor es la evaluación para la dimensión.

1.4.1 Análisis de las métricas para evaluar la dimensión aptitud

Se han desarrollado un conjunto de métricas que evalúan esta dimensión. Las métricas disponibles libremente en las fuentes consultadas son las siguientes:

1-Fitness (Rozinat, van der Aalst 2008).

2-Behavioral Recall (Goedertier, Martens, Vanthienen, Baesens 2009).

3-Aligment Based Trace Fitness (Rozinat, Veloso, van der Aalst 2008).

4-Costed-Basic Fitness Metric (Adriansyah, Van Dongen, Van Der Aalst 2011).

La métrica **Fitness** está definida de la siguiente forma:

$$f = \frac{1}{2}\left(\frac{1-\sum_{i=1}^{k} n_i m_i}{\sum_{i=1}^{k} n_i e_i}\right) + \frac{1}{2}\left(1-\sum_{i=1}^{k} n_i \frac{r_i}{\sum_{i=1}^{k} n_i p_i}\right)$$

Donde:

n_i → Número de instancias del proceso.

m_i → Número de tokens ausentes.

K Número de trazas diferentes agregadas al registro de eventos.

r_i → Número de tokens restantes.

c_i → Número de tokens consumidos.

p_i → Número de tokens producidos durante la generación del registro de eventos.

Esta métrica analiza los *tokens* y las actividades que se generan durante la ejecución de un registro de eventos. Así mide la cantidad de actividades existentes, que pueden estar representadas o no, en el modelo. El valor de la métrica oscila entre 0 y 1. La métrica analiza redes de Petri y penaliza el comportamiento adicional representado.

La métrica **Behavioral Recall** se define de la siguiente forma:

$$r_B^P = \left(\frac{\sum_{i=1}^{K} n_i TP_i}{\sum_{i=1}^{K} n_i TP_i + \sum_{i=1}^{K} n_i FN_i} \right)$$

El resultado de la métrica se obtiene mediante el análisis de cada secuencia de eventos ejecutada. Los valores de TP y FN se inicializan en cero. A partir de estos valores cada secuencia es analizada. Cada vez que ocurre una transición los valores de TP se incrementan en uno. Cuando la transición no se activa, pero se obliga en el modelo el valor de FN se incrementa en uno.

Donde:

K Número de secuencias agrupadas.

n_i → Número de instancias de procesos.

TP_i → Número de actividades analizadas correctamente.

FN_i → Número de eventos disparados por cada transición en las secuencias agrupadas.

Esta métrica analiza los eventos positivos y negativos generados en el modelo. Los eventos positivos son aquellos que se encuentran en el registro de eventos, y los negativos los que genera el algoritmo de descubrimiento ante cualquier característica del registro de eventos que incide en él. Los valores de esta métrica oscilan entre 0 y 1.

La métrica **Aligment Based Trace Fitness** está definida de la siguiente manera:

$$F_{trace} = 1 - \frac{\sum_{\sigma \in E_o} m(\sigma)}{\sum_{\sigma \in E} m(\sigma)}$$

Esta métrica está basada en el análisis de redes de Markov. Estas redes de Markov tienen que ser convertidas a redes de Petri simples. Analiza las secuencias de seguidores entre los eventos generados por el algoritmo al analizar el registro de eventos.

La métrica **Costed-Based Fitness Metric** está definida de la siguiente manera:

$$f = 1 - \sum_{\alpha \in A_s} A_s(a)\, X\, k^s(a) + \frac{\sum_{e \in E_i} k^s(\alpha(e))}{\sum_{e \in E_i} k^i(a(e))}$$

Esta analiza las actividades insertadas penalizando su aparición en el resultado. El análisis en el modelo se va a centrar en las actividades insertadas y las que no se tienen en cuenta en el modelo. El resultado de la métrica oscila entre 0 y 1.

1.4.2 Análisis de las métricas para evaluar la dimensión simplicidad

Las métricas para evaluar esta dimensión indican cuan fáciles de analizar y entender son los modelos. Esta dimensión no debe sobrestimarse ya que un modelo muy simple puede obviar comportamiento presente en el proceso.

La métrica estudiada en esta dimensión fue:

1- Advanced Structural Appropriateness (Rozinat, van der Aalst 2008).

La métrica **Advanced Structural Appropriateness** está definida de la siguiente forma:

$$a_s^{\square} = (T) - \frac{(T_{DA}) + (T_{IR})}{(T)}$$

T Conjunto de transiciones en la red de Petri del modelo.

$S_f \rightarrow$ Conjunto de tareas duplicadas alternativas.

$S_p \rightarrow$ Conjunto de tareas invisibles redundantes.

La métrica para el análisis del modelo tiene en cuenta las tareas duplicadas y redundantes, ya que estas afectan la simplicidad del modelo directamente. Si las actividades duplicadas y

redundantes quedan reflejadas en el modelo este no va a presentar buenos valores de simplicidad. Los valores de la métrica oscilan entre 0 y 1.

Existe otra métrica implementada en el marco de trabajo CoBeFra que también se utiliza en la investigación. Esta, es la unión de una serie de métricas que evalúan la cantidad de transiciones, número de arcos, número de actividades, longitud de la red, entre otras características. Todas ellas inciden de forma directa en la simplicidad del modelo.

1.4.3 Análisis de las métricas para evaluar la dimensión precisión

Las métricas de esta dimensión evalúan que el comportamiento reflejado en el modelo se encuentre presente en el registro de eventos. Esta dimensión es importante cuando se quieren detectar anomalías en las empresas, ya sean fraudes o desviaciones en el proceso.

Las métricas estudiadas fueron:

1- Advanced Behavioral Appropriateness (Rozinat, van der Aalst 2008).

2- Behavioral Specificity (Goedertier, Martens, Vanthienen, Baesens 2009).

3- Behavioral Precision (De Weerdt, De Backer, Vanthienen, Baesens 2011).

4- Best Align Precision (Munoz-Gama, Adriansyah, Carmona, Dongen 2011).

La métrica **Advanced Behavioral Appropriateness** está definida de la siguiente forma:

$$a'B = \frac{S'_F \wedge S_F^m}{2 * S_F^m} + S'_p \wedge \frac{S_p^m}{2 * S_p^m}$$

$S_f \rightarrow$ Relación de seguidores.

$S_p \rightarrow$ Relación de predecesores.

La relación S_f y S_p es simétrica, cuando existe una actividad que no es seguidora ni es seguida por ninguna otra no se representa. Esta métrica se enfoca en analizar las relaciones existentes en el modelo, desechando las actividades aisladas. El valor oscila entre 0 y 1.

La métrica **Behavioral Specifity** se define:

$$S_B^n = \left(\frac{\sum_{i=1}^{k} n_i TN_i}{\sum_{i=1}^{k} n_i TN_i + \sum_{i=1}^{k} n_i FP_i} \right)$$

$k \rightarrow$ Número de secuencias agrupadas.

$n_i \rightarrow$ Número de instancias del proceso.

$TN_i \rightarrow$ Número de actividades negativas para las cuales no se ha habilitado una transición.

$FP_i \rightarrow$ Número de actividades negativas para las cuales se ha habilitado una transición.

Los valores de TN y FP se inicializan en cero. Cada vez que en el corrido de la métrica se encuentra una actividad negativa para la cual no está habilitada ninguna transición TN se incrementa en uno. En cambio sí se encuentra una actividad negativa con transiciones habilitadas el valor de FP se incrementa en uno.

Los eventos negativos son aquellos que se generan para completar el modelo, pero no se encuentran en el registro de eventos. Los eventos negativos son configurables, mediante un número aleatorio.

La métrica **Behavioral Precision** se define:

$$P_B = \left(\frac{\sum_{i=1}^{k} n_i TP_i}{\sum_{i=1}^{k} n_i TP_i + \sum_{i=1}^{k} n_i FP_i} \right)$$

La métrica Behavioral Precision tiene similitud con Behavioral Specifity. Tomando como foco a analizar en el modelo los eventos negativos generados para completarlo.

La métrica **Best-Align Precision** se define:

$$a_p^1(A^1) = \frac{\sum_{\sigma \in S} \omega(\sigma) * (e_x(\sigma))}{\sum_{\sigma \in S} \omega(\sigma) * (a_v(\sigma))}$$

$S \rightarrow$ Conjunto de estados del autómata A' .

Para calcular este indicador, se recogen todas las actividades realizadas por cada estado, ponderando la importancia del estado. Se comparara el conjunto de actividades permitidas por el modelo para el mismo estado. En general se cuantifican las imprecisiones que pueden existir entre el modelo y el registro de eventos.

1.4.4 Análisis de las métricas para evaluar la dimensión generalización

En el marco de trabajo CoBeFra están implementadas una serie de métricas de esta dimensión. En la literatura consultada se encontró la métrica **generalization**. La cual se define a continuación:

$$generalization(L,M)=1-\frac{1}{\epsilon}\sum_{e\epsilon\varepsilon} pnew(|diff(e)|,|(e)|)$$

$pnew$→ Probabilidad de que el próximo estado visitado sea un nuevo estado.

$diff$→ Número de actividades únicas en el modelo.

e→ Número de actividades diferentes en el modelo.

Si el valor de la función **generalization** está cercano a 0 es probable que los nuevos elementos presenten un comportamiento que no se reflejaba antes. Si está cerca de 1 es poco probable que el próximo evento presente un comportamiento reflejado anteriormente.

Las métricas son importantes para evaluar la calidad de los modelos de procesos. También resultan útiles para la recomendación de algoritmos de descubrimiento, ya que la evaluación de los modelos descubiertos aporta información relevante sobre el rendimiento de los algoritmos. Además de la calidad de los modelos descubiertos, para realizar la recomendación de un algoritmo de descubrimiento, se deben tener en cuenta las características que inciden en el descubrimiento.

1.5 Técnicas de recomendación

La recomendación de los algoritmos de descubrimiento es uno de los retos actuales del descubrimiento de modelos de procesos. En la actualidad no existe un mejor algoritmo de descubrimiento que se pueda aplicar en varios contextos y se obtengan buenos resultados (Pérez Alfonso, Yzquierdo Herrera 2012). En las fuentes consultadas se identificaron técnicas que permiten recomendar algoritmos de descubrimiento. Estas técnicas son: análisis de regresión, árboles de decisión y evaluación empírica.

La evaluación empírica es una de las técnicas utilizadas para la recomendación de algoritmos de descubrimiento. Fue desarrollada para evaluar el rendimiento de los algoritmos de descubrimiento. Consiste en aplicar a un conjunto de registro de eventos varios algoritmos de descubrimiento. Los resultados obtenidos son evaluados con una serie de métricas por cada dimensión de calidad. La ejecución de cada métrica y cada algoritmo tiene un costo computacional asociado que varía en función de sus características. Una de las principales limitantes de la evaluación empírica es el tiempo que demora realizarla. Los tiempos de ejecución de los algoritmos de descubrimiento son altos, en la Figura 2 se observa el tiempo de ejecución de cinco (5) algoritmos de descubrimiento. Al ejecutar el algoritmo Genetic Miner sobre el registro de eventos UFM demoró cuatro (4) días 13 horas 49 minutos y 15 segundos en descubrir un modelo de proceso. Al tiempo de ejecución de los algoritmos se le suma el tiempo que demoran las métricas en evaluar los modelos de procesos descubiertos.

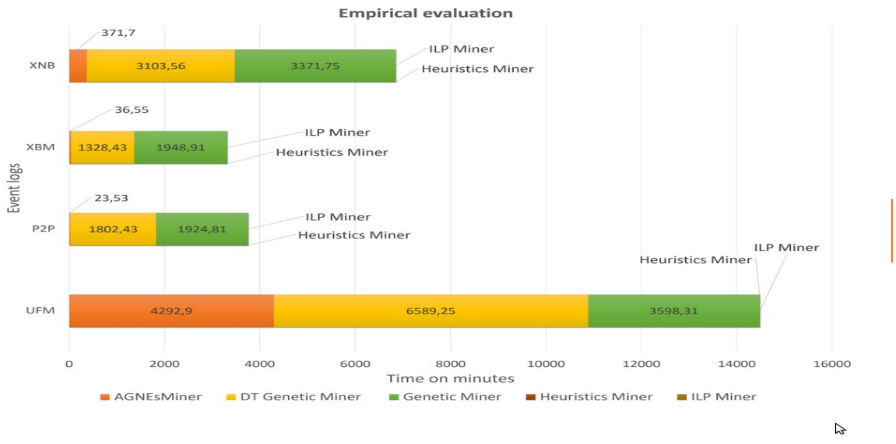


Figura 2: Tiempo de ejecución de la evaluación empírica (Pérez Alfonso, Yzquierdo Herrera 2012).

El análisis de regresión es una técnica desarrollada a partir de los análisis de regresión matemáticos, teniendo un modelo de referencia y el registro de eventos que se quiere minar. Para realizar la recomendación se seleccionan modelos de referencia de alta calidad construyendo a partir de estos un modelo de regresión que permite estimar la similitud de otros modelos de procesos sin realizarles una evaluación empírica (Wang, Zhang, Cai 2012).. Su

propuesta está compuesta por una fase de aprendizaje y una fase de recomendación, como se puede apreciar en la Figura 3.

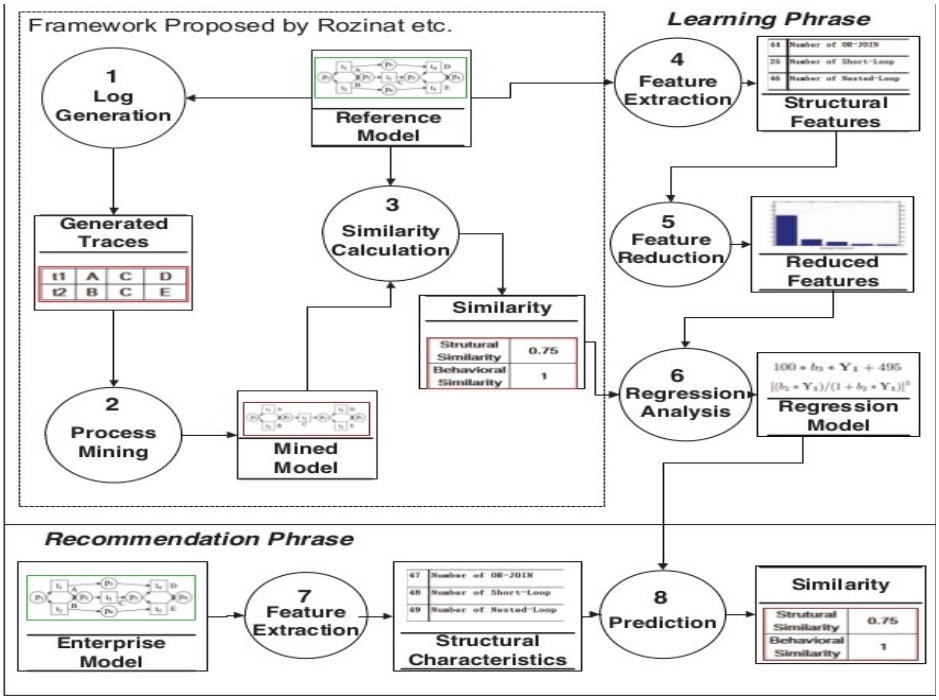


Figura 3: Análisis de regresión (Wang, Zhang, Cai 2012).

Durante la fase de aprendizaje se establece la relación entre las características estructurales de los modelos de procesos y los valores de similitud que se obtienen al evaluar los algoritmos. Primeramente se determinan los valores de similitud obtenidos al aplicar el marco de trabajo planteado en (Rozinat, Medeiros, Günther, Weijters, Aalst 2007) a una muestra aleatoria del conjunto de modelos de referencia. Los modelos significativos se identifican empleando los valores de similitud obtenidos a partir de su mayor influencia en la distancia y variación entre estos valores.

De los modelos significativos se extraen rasgos distintivos que incluyen: número de transiciones, número de lugares, número de *and-joins*, número de *and-splits*, número de *xor-joins*, densidad de aristas, número de tareas invisibles del modelo, entre otras. Utilizando PCA (Principal Component Analysis (Jolliffe 2002)) las características son reducidas a las más significativas, que a su vez son linealmente independientes entre sí.

Por último, se construye un modelo de regresión que permite establecer la relación entre las

características significativas de un modelo y los valores de similitud que se obtienen al aplicar algoritmos de descubrimiento. La similitud entre el modelo de referencia y el modelo descubierto se evalúa utilizando la métrica de similitud estructural propuesta por (Bae, Liu, Caverlee, Zhang, Bae 2007) y la métrica para similitud de comportamiento expuesta en (Wang, He, Wen, Wu, ter Hofstede, Su 2010).

En la fase de recomendación se extraen las características del conjunto de modelos para predecir los resultados de similitud empleando el modelo de regresión. A partir de estos valores estimados, se propone el algoritmo ideal para el descubrimiento de los procesos asociados al conjunto de modelos.

Este enfoque implica ciertos requerimientos que limitan su aplicación:

1- La evaluación y predicción se realiza sobre modelos de referencia especificados en redes de Petri.

2- El enfoque presupone que la ejecución real de los procesos guarden una relación cercana con sus modelos de referencia.

3- La construcción del modelo de regresión a partir de características de los modelos descarta aspectos como el ruido, la ausencia de información y el grado de completitud del registro de eventos.

Estos requerimientos traen consigo un conjunto de implicaciones negativas para el proceso de descubrimiento, estas son:

- En la mayoría de los entornos reales donde se necesitan aplicar algoritmos de descubrimiento, no están descritos los modelos de procesos o estos son inconsistentes y/o incompletos.
- En contextos donde las características del registro de eventos real difieren de las características de los registros de eventos generados artificialmente por los modelos de referencia, la probabilidad de que se obtengan resultados poco exactos es alta.
- Esto posee un impacto significativo en el rendimiento de los algoritmos de descubrimiento.

En una dirección alternativa para la solución al problema de la selección de algoritmos se encuentra el trabajo de (Lakshmanan, Khalaf 2012). Las autoras construyen un árbol de decisión a partir de la comparación de cinco algoritmos de descubrimiento: *Alpha* (Aalst, Weijters, Maruster 2004), sus sucesores (Medeiros, Dongen, Aalst, Weijters 2004); *Fuzzy Miner, Genetic Mine*r (De Medeiros, Weijters, Van Der Aalst 2007) y *Two-step approach* (Aalst, Rubin, Verbeek, Van Dongen, Kindler, Günther 2010). La comparación establece las potencialidades de los algoritmos para enfrentar retos relacionados con tareas invisibles o duplicadas, lazos, paralelismos, alternativa no libre y ruido.

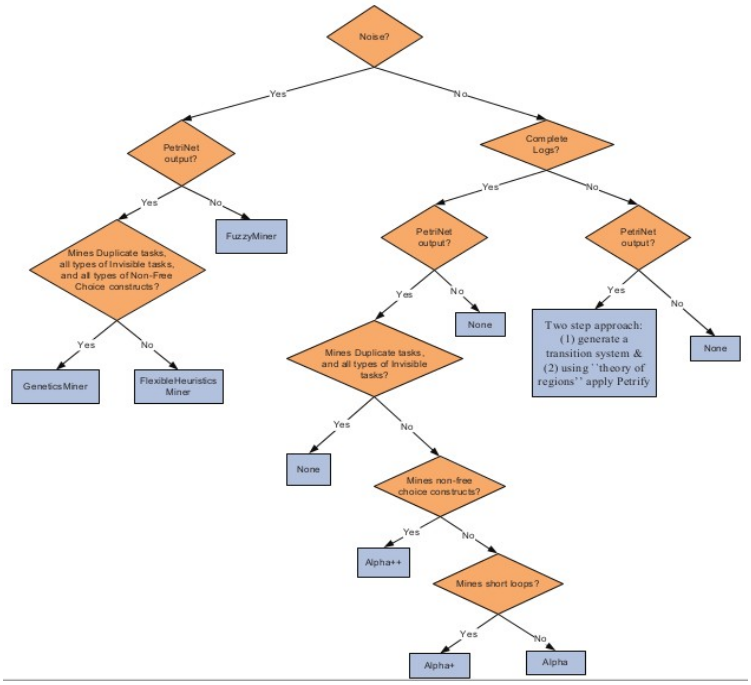


Figura 4: Árbol de decisión (Lakshmanan, Khalaf 2012).

Aunque el árbol de decisión propuesto (Figura 4) se basa en las potencialidades antes mencionadas así como en la notación del modelo descubierto, no especifica cómo identificar la presencia de situaciones desafiantes como ruido, lazos, tareas duplicadas, alternativas no libres y tareas ocultas en el proceso a minar. Es preciso señalar que la identificación de estas características a partir de un registro de eventos no es trivial. En sentido general este trabajo aporta algunos elementos teóricos importantes pero carece de aplicabilidad práctica, por el tipo de información que requieren los algoritmos.

1.6 Recomendación de algoritmos de descubrimiento como problema de clasificación.

El análisis de los trabajos relacionados con la evaluación y selección de algoritmos de descubrimiento apunta a que es necesaria una técnica de selección de algoritmos de descubrimiento que cumpla con las siguientes condiciones:

1- Debe tener en cuenta **rasgos del proceso** (patrones de control de flujo, heterogeneidad de los casos, nivel de estructuración) y **rasgos del registro de eventos** (ruido, ausencia de información, completitud, tamaño).

2- Debe identificar los rasgos antes mencionados en el registro de eventos, ya que es la principal fuente de información sobre el proceso que está disponible en todos los ambientes.

La clasificación es el problema relativo a la construcción de un procedimiento que es aplicado a una secuencia continua de casos, en la que un nuevo caso debe ser asignado a uno de los conjuntos de clases predefinidas sobre la base de atributos observados o características (Michie, D., Spiegelhalter, D.J, Taylor, C.C, Campbell, J 1994). La recomendación de los algoritmos de descubrimiento puede expresarse en términos de un problema de clasificación. Un problema de clasificación requiere, un nuevo caso a clasificar y una nueva clase para ser asignada. El nuevo caso a clasificar es el registro de eventos y el algoritmo es la clase a la que será asignado. Este procedimiento de clasificación, en el que las clases son conocidas, se ha denominado de diversas maneras: como reconocimiento de patrones o aprendizaje supervisado.

Tipificar la recomendación de los algoritmos de descubrimiento como un problema de clasificación abre el camino para la aplicación de técnicas desarrolladas en un área del conocimiento más conocida (Pérez Alfonso, Yzquierdo Herrera 2012). Técnicas del área de clasificación han sido utilizadas para la recomendación de algoritmos de descubrimiento (Wang, He, Wen, Wu, ter Hofstede, Su 2010). Sin embargo, para la concepción de un mecanismo global de recomendación que supere las limitaciones de las soluciones existentes, es importante incorporar las características anteriormente mencionadas.

Para resolver un problema de clasificación el diseño del clasificador es un elemento esencial. En términos generales existen tres enfoques diferentes para ello. El primero se basa en el

concepto de similitud, el segundo se trata de un enfoque probabilístico y el tercero en construir la decisión en límites directamente, mediante la optimización de cierto criterio de error.

Un reto importante para el problema de recomendación a través de la clasificación es la construcción del conocimiento a almacenar. El conocimiento existente en la minería de proceso debe ser recuperado y estructurado. Por ello la confección de una base de conocimiento que albergue toda esta información permitirá tratar la recomendación de algoritmos de descubrimiento como un problema de clasificación.

1.7 Base de conocimiento

Una base de conocimiento es un tipo especial de base de datos. Estas son la evolución lógica de las bases de datos tradicionales. En ellas se plasman elementos de conocimiento sobre un tema específico. Con el conocimiento almacenado pueden realizarse disímiles análisis. Las bases de conocimiento son muy utilizadas en la medicina y las organizaciones, como apoyo a la toma de decisiones (INTERCHANGE 1998) (Fayyad, Piatetsky-Shapiro, Smyth, Uthurusamy 1996). Una base de conocimiento puede almacenar un conjunto de reglas o casos que sirven para obtener información que no se encuentra almacenada de forma explícita.

1.8 Herramientas y tecnologías

Para el desarrollo de la solución se tienen en cuenta un conjunto de herramientas y tecnologías que permiten la experimentación y el desarrollo del complemento. Las herramientas Process log generator, ProM y CoBeFra son utilizadas en la realización de los registros de eventos, el descubrimiento de los modelos de proceso y la evaluación de estos. Las demás se utilizaron en la construcción del complemento para la plataforma ProM y la base de datos objetual.

1.8.1 ProM

ProM es un marco de trabajo para el desarrollo de herramientas de minería de procesos en un ambiente estandarizado (H, DONGEN B.F 2012). Está desarrollado en Java y se encuentra disponible bajo licencia GPL. ProM está concebido para admitir la adición de complementos y de esta manera posibilitar el desarrollo de nuevos algoritmos y técnicas en el campo de la minería de procesos (Jan Claes, Geert Poels 2012b). Los complementos necesitan determinada cantidad de parámetros de entrada y producen uno o varios objetos de salida. Los parámetros de entrada pueden ser registros de eventos u objetos obtenidos a partir del procesamiento de otros complementos. Mientras que los objetos de salida pueden ser empleados como parámetros de entrada de otros

complementos. Este marco de trabajo cuenta con más de 600 complementos, cada uno de los cuales posibilita realizar diferentes análisis (AALST W. M. P 2013). Las herramientas desarrolladas en ProM han sido empleadas en el análisis de procesos provenientes de diferentes dominios entre los que se encuentran gubernamental, hospitalario y sistemas ERP (Aalst 2011).

1.8.2 CoBeFra

CoBeFra es una plataforma que tiene implementada una serie de métricas que permiten la evaluación de los modelos de procesos. Las métricas están implementadas según las cuatro dimensiones de calidad. CoBeFra analiza registro de eventos en formato XES y MXML. Los modelos de proceso que analiza solo pueden estar en redes de Petri. Los resultados obtenidos pueden ser exportados en formato CSV o CBI (De Weerdt, Baesens, Vanthienen 2013).

Es un sistema implementado en Java con licencia GPL, al cual se le pueden incorporar otras métricas según se vayan desarrollando.

1.8.3 Eclipse

Eclipse es un entorno de desarrollo (IDE) de código abierto, popular en el desarrollo de aplicaciones escritas en el lenguaje de programación Java. También es utilizado para desarrollar en los lenguajes C, C++, Python, entre otros. Es útil para integrar herramientas de desarrollo, con una arquitectura abierta y basada en complementos por lo que permite agregar funcionalidades e integrar diversos lenguajes de programación. Está desarrollado en Java y es multiplataforma, así como de fácil instalación y utilización, lo que lo hace muy conveniente para el desarrollo de la propuesta de solución. Adicionalmente la comunidad de desarrollo de minería de proceso sugiere a Eclipse, por sus características, como IDE para el desarrollo de complementos para la plataforma ProM (Hou 2007).

1.8.4 Generador de Log

Process Log Generator es una aplicación que permite generar procesos de negocio especificando algunos parámetros de complejidad. Tiene la capacidad de ejecutar y generar un registro de proceso con las actividades observadas. Este sistema fue desarrollado como apoyo

en las investigaciones para la construcción de grandes conjuntos de registros de eventos. Los registros se guardan en formato MXML, formato soportado por la herramienta ProM (Burattin, Sperduti 2011).

1.8.5 Base de datos objetual

En una base de datos orientada a objetos, la información se representa mediante objetos. Un ODBMS (*object database management system* - **sistema gestor de base de datos orientada a objetos**) hace que los objetos de la base de datos aparezcan como objetos de un lenguaje de programación. Un ODBMS extiende de los lenguajes de datos persistentes de forma transparente; permite el control de concurrencia, la recuperación de datos, las consultas asociativas y otras capacidades (*Modelo Objetual - Bota del día* 2013).

Por el bajo nivel de información que se va a almacenar se utiliza una base de datos objetual.

1.8.6 Berkeley

Oracle Berkeley es una familia de bases de datos incorporadas, de código abierto. Berkeley permite a los desarrolladores incorporar en sus aplicaciones un motor de base de datos transaccional, rápido y escalable con disponibilidad y confiabilidad de clase industrial. Puede escalar por debajo de las cargas extremas, pero no requiere administración continua de la base de datos (Carretero Pastor, Marín 2011). Actualmente se encuentra en su versión 6.0.

Las principales características de este sistema son:

- Recuperación de datos en forma secuencial e indexada.

- Procesos múltiples por aplicación e hilos múltiples por proceso.

- Datos en memoria, en disco o ambos.

- Encriptación de datos por el algoritmo AES.

- Registros de hasta 4GB y tablas de hasta 256TB.

- Soporte para transacciones distribuidas.

- Respaldos en frío y en caliente.

- Replicación.

- Administración automática.

- Soporte de los lenguajes C, C++, Java, Perl, Python, PHP, TCL y Ruby.

- Disponible en los sistemas operativos: Linux, Windows, BSD Unix, Solaris, Mac OS.

Este tipo de bases de datos objetuales presentan un buen rendimiento ya que eliminan los gastos de comunicación interprocesos y SQL. Se integra a la aplicación y es invisible para los usuarios finales, por lo que no requiere administración. Además presenta flexibilidad ya que los desarrolladores pueden configurar muchos aspectos de Berkeley DB.

1.9 Conclusiones parciales

La minería de procesos es una disciplina que se encarga del descubrimiento de modelos de procesos. Para ello se utilizan algoritmos de descubrimiento que descubren modelos de proceso mediante la extracción de la información de los registros de eventos. Los modelos de procesos pueden ser evaluados mediante cuatro dimensiones de calidad, para las que existen diversas métricas.

Los procesos de descubrimiento se ven afectados por características de los procesos y registros de eventos a minar. No tener en cuenta estas características puede afectar las dimensiones de calidad de los modelos descubiertos. Por tanto es necesario elegir con criterios certeros qué algoritmo utilizar para el proceso de descubrimiento. Para ello se utilizan técnicas de recomendación, sin embargo, las técnicas identificadas implican grandes costos de tiempo y recursos. Además, algunas de ellas descartan características de los procesos y el registro de eventos, por lo que no resulta factible utilizarlas en entornos reales.

El problema de recomendación puede ser resuelto mediante la clasificación. Una base de conocimiento recupera y estructura el conocimiento disperso, por lo cual se considera factible como propuesta de solución, ya que permite tratar la recomendación de algoritmos de descubrimiento como un problema de clasificación.

Las herramientas y tecnologías seleccionadas para la propuesta de solución son las siguientes: ProM como herramienta para el descubrimiento y la integración de la base de conocimiento; CoBeFra para la evaluación de modelos de procesos descubiertos; Eclipse como entorno de

desarrollo; Process Log Generator para generar los registros de eventos; Base de datos objetual como tecnología para la persistencia de la base de conocimiento y Oracle Berkeley como gestor de base de datos.

En el capítulo 2 se describe la metodología utilizada para la generación de la información a almacenar en la base de conocimiento.

Capítulo 2: Construcción de la base de conocimiento

2.1 Introducción

En el presente capítulo se describe la estructura de la base de conocimiento para la recomendación de algoritmos de descubrimiento. La base de conocimiento está conformada por un conjunto de casos donde se reflejan las características que afectan el descubrimiento y los resultados de las métricas para los modelos descubiertos. Almacenar esta información en una base de conocimiento permite utilizar técnicas de clasificación para solucionar el problema de la recomendación.

2.2 Características de las métricas a utilizar

Las métricas para conformar la base de conocimiento para la recomendación deben cumplir un conjunto de características para su utilización. Estas características están clasificadas según su funcionamiento y de acuerdo a las necesidades de la propuesta de solución (R.P.J.M. van ArendonkBSc 2011).

Las propiedades de funcionamiento que deben cumplir las métricas son:

1. **Validez:** Las métricas se comportan de manera predecible. Cuando mejora la característica, mejora el resultado.
2. **Estabilidad:** No se afecta el valor de la métrica cuando varía un parámetro que no se mide.
3. **Operacionabilidad:** El valor de la métrica se debe poder medir. En la práctica el valor debe oscilar entre 0 y 1.
4. **Reproducibilidad:** Cuando se utilice la métrica por diferentes usuarios los valores deben ser comparables.
5. **Localizabilidad:** El valor que devuelve la métrica debe indicar el problema que presente el modelo.

La propiedad que exige la investigación es:

6. **Soporte:** Las métricas a utilizar deben estar implementadas en CoBeFra o ProM.

A continuación se analizan las métricas descritas por cada dimensión teniendo en cuenta las características enunciadas:

Aptitud

Nombre	P-1	P-2	P-3	P-4	P-5	P-6
Fitness	*	*	*	*	*	*
Behavioral Recall	*	*	*	*	*	*
Aligment Based Trace Fitness	*		*		*	
Costed-Basic Fitness Metric	*	*	*	*	*	*

Tabla 2: Métricas de aptitud.

Simplicidad

Nombre	P-1	P-2	P-3	P-4	P-5	P-6
Advanced Structural Appropriateness	*	*	*	*	*	*
Varias Métricas	*	*	*	*	*	*

Tabla 3: Métricas de simplicidad.

Precisión

Nombre	P-1	P-2	P-3	P-4	P-5	P-6
Advanced Behavioral Appropriateness	*	*	*	*	*	*
Behavioral Specificity	*	*			*	*
Behavioral Precision	*	*	*		*	*
Best Align Precision (ETC)	*	*	*	*	*	*

Tabla 4: Métricas de precisión.

Generalización

Nombre	P-1	P-2	P-3	P-4	P-5	P-6
Generalization	*	*	*	*	*	*

Tabla 5: Métricas de generalización.

Para la evaluación de una métrica, el tiempo máximo de ejecución debe ser menor a las 2 horas. Se selecciona como límite máximo 2 horas, teniendo en cuenta la cantidad de modelos a evaluar y el tiempo disponible para realizar la investigación. Si se excede este tiempo, el valor de las métricas que no se obtengan, se almacenan como (-∞).

2.3 Construcción de la base de conocimiento

Para realizar la recomendación aplicando técnicas de clasificación se deben tener en cuenta los siguientes factores (Pérez Alfonso, Yzquierdo Herrera 2012):

1- El registro de eventos es la principal fuente de información disponible.

2- Las características del proceso y del registro de eventos influyen en el rendimiento de los algoritmos de descubrimiento.

3- La evaluación de los modelos descubiertos brinda una medida del rendimiento de los algoritmos de descubrimiento ante las características del registro de eventos.

Para la confección de la base de conocimiento se han tenido en cuenta los rasgos que caracterizan a los casos que representan el conocimiento. La base de conocimiento está conformada por un conjunto de casos. Cada caso está compuesto por:

- las características del registro de eventos,
- las características del proceso,
- el algoritmo utilizado para obtener el modelo y,
- los valores de las métricas, obtenidos para cada dimensión de calidad.

Los casos que conforman la base de conocimiento fueron generados artificialmente. En la siguiente figura se muestra la secuencia de fases utilizada para la generación de los casos. La salida de cada una de las fases es la entrada de la fase siguiente.

Figura 5: Generación de los casos para la base de conocimiento.

2.3.1 Generación de los registros de eventos

En la primera fase se tienen en cuenta las características que afectan el proceso de descubrimiento, tanto del proceso como del registro de eventos. Para generar los registros de eventos se utilizó la herramienta “Process log generator”. Esta herramienta permite crear registros de eventos artificiales combinando características que afectan el descubrimiento. La combinación de estas características permite la obtención de registros de eventos similares a registros de eventos reales. Las características del proceso que se tienen en cuenta para generar los registros de eventos son los lazos, las alternativas no libres y las tareas ocultas. De estas características se almacena la información acerca de si aparecen o no en los registros de eventos generados.

Las dos características del registro de evento que se consideran son ruido e intervalo. El indicador a medir es el por ciento de aparición en el registro de eventos, el cual puede tomar cualquiera de los valores siguientes: 0%, 25%, 50%, 75%, 100%; debido a que la herramienta ¨Process log generator¨ permite la creación de registros de eventos con estas especificaciones. Las demás características que afectan el descubrimiento de procesos no se tienen en cuenta ya que la herramienta no permite la creación de registros de eventos con tareas duplicadas.

Para la generación de los registros de eventos se realizó un diseño experimental factorial completo, donde se combinan las características seleccionadas. Este diseño factorial cuenta con cinco clases, ruido, intervalos, lazos, alternativas no libres y tareas ocultas.

$$F = C1 * C2 * C3 * C4 * C5$$

Siendo:

$C1 - Ruido$

$C2 - Intervalos\ de\ ruido$

$C3 - Lazos$

$C4 - Alternativas\ no\ libres$

$C5 - Tareas\ ocultas$

El ruido y el intervalo del ruido se dividen en 5 subclases cada uno. Por otra parte, el lazo, alternativas no libres y tareas ocultas, en 2 subclases cada uno.

Siendo:

$$FC = 5 * 5 * 2 * 2 * 2$$

$$FC = 200$$

Como la herramienta ¨Process log generator¨ permite generar los registros de eventos con diferentes cantidades de trazas, se crearon 3 grupos según la cantidad de trazas.

Cantidad de registros de eventos	Cantidad de trazas	Clasificación
67	500	Pequeños
67	1000	Medianos
67	1500	Grandes

Tabla 6: Cantidad de trazas por registro de eventos.

Los grupos generados son pequeños, medianos y grandes, el primero está compuesto por registros de eventos con 500 trazas, los medianos con 1000 y el grupo de los registros de eventos grandes engloba a los que tienen 1500 trazas. Se generaron 201 registros de eventos en total para garantizar la misma cantidad de registros de eventos por cada grupo.

Los registros de eventos generados por la herramienta presentan un nivel 4 de calidad porque los eventos se registran automáticamente y de manera sistemática y confiable. La combinación de estas características permite la obtención de registros de eventos que muestran diversidad de comportamiento con peculiaridades de entornos reales. A partir de estos registros se aplican los algoritmos de descubrimiento para obtener los modelos de proceso.

2.3.2 Descubrimiento de los modelos de procesos

En la segunda fase de la construcción de los casos se aplica, a los registros de eventos generados, una selección de algoritmos de descubrimiento. Para la selección de los algoritmos se tienen en cuenta investigaciones que evalúan el rendimiento y los resultados de su aplicación en diferentes entornos. Otra característica a tener en cuenta para esta selección es que representen los modelos de procesos descubiertos en redes de Petri o puedan convertirse a estas. En la siguiente tabla se muestra la evaluación de varios algoritmos aplicados a registros de eventos con características similares a los almacenados en la base de conocimiento.

Fitness		Generalization
Alignment Based Fitness	*Behavioral Recall*	*Alignment Based Pr. Generalization*
ILP Miner (9.75)	**ILP Miner (11.11)**	**AGNEs Miner (9.36)**
AGNEs Miner (9.64)	**Heuristics Miner (9.61)**	**ILP Miner (9.34)**
Causal Miner (9.31)	**Alpha Miner+ (9.45)**	**TS Miner (8.53)**
DWS Miner (8.72)	**AGNEs Miner (8.89)**	**Heuristics Miner (8.42)**
Heuristics Miner (8.59)	**DWS Miner (8.77)**	**DWS Miner (8.28)**
TS Miner (8.48)	TS Miner (6.69)	**Causal Miner (7.95)**
Region Miner (6.80)	Genetics Miner (6.56)	**Alpha Miner (7.22)**
Process Tree Miner (6.34)	Alpha Miner (6.28)	**Process Tree Miner (7.06)**
Alpha Miner (6.30)	Alpha Miner++ (6.17)	Alpha Miner++ (6.28)
Genetics Miner (5.77)	Causal Miner (5.28)	Genetics Miner (6.00)
Alpha Miner++ (5.11)	DT Genetics Miner (4.77)	Region Miner (5.83)
DT Genetics Miner (4.83)	Region Miner (4.08)	DT Genetics Miner (4.80)
Alpha Miner+ (1.36)	Process Tree Miner (3.34)	Alpha Miner+ (1.92)

Simplicity	Precision	
Weighted P/T Average Arc Degree	*One Align Precision*	*Behavioral Precision*
Alpha Miner+ (11.07)	**AGNEs Miner (10.34)**	**DWS Miner (10.09)**
DT Genetics Miner (8.52)	**DWS Miner (9.20)**	**AGNEs Miner (9.38)**
AGNEs Miner (7.12)	**Alpha Miner (8.75)**	**Causal Miner (8.97)**
Causal Miner (6.93)	**ILP Miner (8.67)**	**Alpha Miner (8.78)**
DWS Miner (6.72)	**Heuristics Miner (8.61)**	**Heuristics Miner (8.44)**
ILP Miner (6.72)	**Causal Miner (7.83)**	**ILP Miner (8.27)**
Alpha Miner++ (5.43)	**TS Miner (7.78)**	Alpha Miner++ (7.45)
Region Miner (5.26)	Alpha Miner++ (7.48)	TS Miner (7.11)
Heuristics Miner (5.24)	Region Miner (6.20)	Genetics Miner (6.30)
Alpha Miner (4.79)	Genetics Miner (5.52)	Process Tree Miner (5.14)
Genetics Miner (4.41)	Process Tree Miner (5.09)	Region Miner (4.11)
Process Tree Miner (4.09)	DT Genetics Miner (4.33)	DT Genetics Miner (4.09)
TS Miner (3.59)	Alpha Miner+ (1.19)	Alpha Miner+ (2.88)

Tabla 7: Evaluación de los algoritmos de descubrimiento (Sepp e K.L.M. vanden Broucke, Cédric Delvaux 2013).

En este estudio (Seppe K.L.M. vanden Broucke, Cédric Delvaux 2013) se ilustra el impacto que tienen las características de los registros de eventos y los procesos a minar, en los algoritmos de descubrimiento, por cada una de las dimensiones de calidad. En la siguiente tabla se muestra el resultado del estudio, las flechas representan si afectan de forma positiva o negativa a los algoritmos:

Characteristics	Alpha Miner	Alpha Miner +	Heuristic Miner	Genetic Miner	DWS Miner	AGNEs Miner	TS Miner	ILP Miner	Causal Miner
Choice	S ↓	P ↓ S ↓	F↑ P↑	P↑	F↑			F↑ P↑ G↑	
Parallelism		P ↓	P↑		F↑				
Loop	F ↓ P ↓ S ↑	F ↓ S ↑	F ↓ P ↓ S ↑ G ↓	S ↑	F ↓ P ↓ S ↑	S ↑	F ↓ P ↓ S ↑	F ↓ P ↓ G ↓	S ↑
Invisible Task									
Duplicate Task			F ↓ P ↓		F ↓ P ↓	P ↓	F ↓ P ↓ S ↑	F ↓ G ↓	
Non-free choice	P ↓ G ↓	F↑ P↑	F↑	F↑ P↑	F↑ P ↓	F ↓ P ↓	P ↓ S ↑ F ↑	F ↓ P ↓ G ↓	F ↓ P ↓ S ↑
Nested Loop	F ↓ P ↑ S ↑	F ↓ S ↑	F ↓		F ↓ P ↑ S ↑	F ↓ S ↑	P ↓ S ↓	F ↓ P ↓ G ↓	S ↑
Number of traces			P ↓	F↑ P↑		F↑			
Number of distinct traces				F↑	F↑ P ↓		F↑ P↑	S ↑ G ↑	
Noise	S ↑	F ↓ P ↑	F ↓ P ↓	F↑	S ↑ P ↓			S ↑ P ↓	

Figura 6: Impacto de las características en los algoritmos de descubrimiento, por dimensión (Seppe K.L.M. vanden Broucke, Cédric Delvaux 2013).

A partir de los resultados en diferentes entornos y por ser los recomendados en estudios similares (Jochen De Weerdt, Manu De Backer, Jan Vanthienen, Bart Baesens 2012) (Fundora-Ramírez Osiel 2013), se seleccionan los siguientes algoritmos de descubrimiento:

1- Heuristic Miner (Weijters, van der Aalst, de Medeiros 2006).

2- ILP (Verbeek, van der Aalst 2012).

3- Inductive Miner (Weijters, Aalst 2003).

4- Genetic Miner (De Medeiros, Weijters, Van Der Aalst 2007).

5- Alpha Miner (Aalst, Rubin, Verbeek, Van Dongen, Kindler, Günther 2010).

Para el proceso de descubrimiento se utiliza la herramienta ProM en su versión 6.3 donde se encuentran implementados los algoritmos de descubrimiento seleccionados.

El proceso de descubrimiento se detiene si el algoritmo sobrepasa las 5 horas de ejecución. Se establece este umbral de tiempo porque la cantidad de modelos a evaluar es elevado y el tiempo disponible es limitado. En estudios similares se observa que el tiempo de descubrimiento es elevado (Jochen De Weerdt, Manu De Backer, Jan Vanthienen 2012) (Sepp e K.L.M. vanden Broucke, Cédric Delvaux 2013). Si al finalizar ese tiempo el algoritmo no descubrió un modelo de proceso se asume que los valores de las métricas no están definidos.

A partir de los modelos descubiertos por los algoritmos, se realiza su evaluación utilizando el método modelo-log y las métricas seleccionadas.

2.3.3 Evaluación de los modelos de procesos descubiertos

En la tercera fase se evalúan los modelos de procesos descubiertos. Para ello se utilizan un conjunto de métricas que cumplen con los requerimientos establecidos.

Las métricas utilizadas por cada dimensión de calidad se muestran en la siguiente tabla:

Dimensión	Nombre
Aptitud	Fitness
Precisión	ETC
Simplicidad	ARC Average
Generalización	Generalization

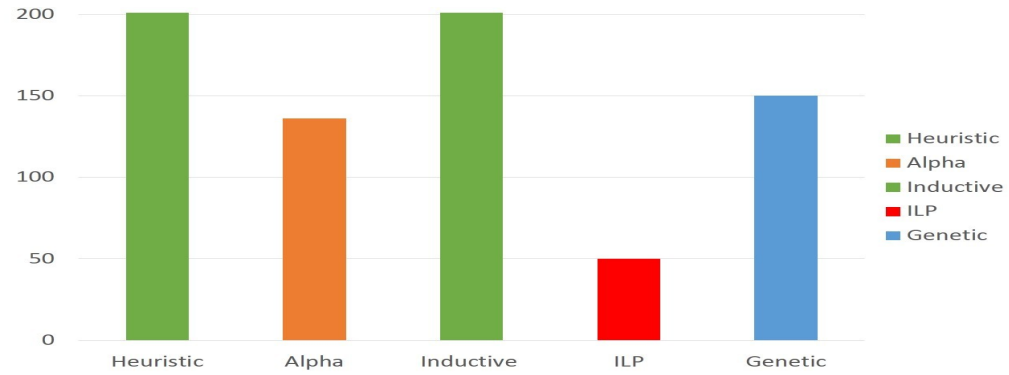
Tabla 8: Métricas seleccionadas.

Para realizar el proceso de evaluación se utiliza el marco de trabajo CoBeFra, en el mismo se encuentran implementadas las métricas seleccionadas. CoBeFra evalúa los modelos de procesos siguiendo el método modelo-log. A partir del registro de eventos minado y el modelo de procesos descubierto se aplican las métricas seleccionadas por cada dimensión.

Al terminar esta fase se construye la base de conocimiento. Con esta información se pueden realizar análisis que permiten determinar los algoritmos que tienen un mejor rendimiento en el descubrimiento.

2.4 Análisis de la base de conocimiento construida

Algunos algoritmos seleccionados poseen limitantes en el descubrimiento. Como se muestra en la gráfica 1, tres de los algoritmos no descubrieron la totalidad de los modelos asociados a los registros de eventos generados.



Gráfica 1 Cantidad de modelos descubiertos por los algoritmos de descubrimiento.

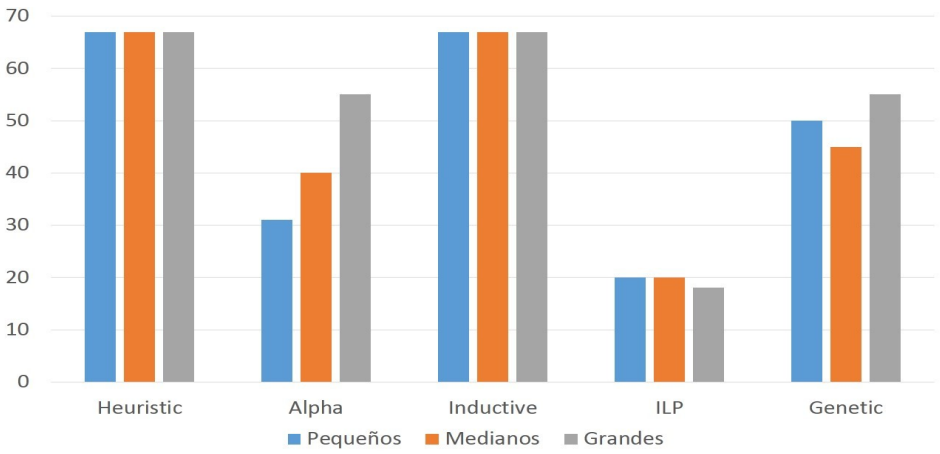
Al aplicar los algoritmos Heuristic Miner e Inductive Miner en los registros de eventos generados se obtiene la totalidad de los modelos de procesos asociados. Los tiempos de descubrimiento del algoritmo Heuristic Miner oscila entre 5 segundos y un 1 minuto. En los registros de eventos con más de 100 actividades los modelos descubiertos por el algoritmo Heuristic Miner son de tipo *spaguetti*[1]*.* Los tiempos de descubrimiento del algoritmo Inductive Miner oscilan entre 10 segundos y 2 horas. De forma general los algoritmos presentan buenos resultados y en la totalidad de los casos se descubre un modelo de procesos.

Por su parte los algoritmos ILP y Alpha Miner al ser aplicados en los registros de eventos generados no obtienen todos los modelos de procesos asociados. En registros de eventos con más de 24 actividades el algoritmo ILP no encuentra un modelo de procesos. En algunos registros de eventos menores de 24 actividades pero con niveles de ruido elevados tampoco es capaz de descubrir el modelo de proceso. Los tiempos de ejecución del algoritmo ILP oscilan entre 10 segundos y 5 horas, donde es frecuente el descubrimiento de modelos de procesos por encima de la hora de ejecución. El algoritmo Alpha Miner al enfrentarse a registros de eventos con más de 66 actividades no encuentra modelos de procesos en la mayoría de los casos. Se exceptúa un caso con 321 actividades donde descubre un modelo de proceso, este caso tiene poca relación entre las actividades y bajos niveles de ruido. Los tiempos de ejecución del algoritmo Alpha Miner oscilan entre 5 segundos y 5 horas. En general, se evidencia que estos algoritmos de descubrimiento tienen fuertes limitantes para enfrentar registros de eventos complejos y con mucha relación entre las actividades.

Al aplicar el algoritmo Genetic Miner en los registros de eventos generados se obtienen 150 modelos de proceso. Los tiempos de ejecución del algoritmo oscilan entre 1 minuto y 5 horas. Los modelos de proceso descubiertos presentan un elevado nivel de aptitud y precisión. En los casos donde no se descubren modelos de proceso se debe a que el algoritmo demora más de 5 horas y se detiene el proceso de descubrimiento.

Los registros de eventos generados poseen diferentes cantidades de trazas. En la siguiente gráfica se observa la cantidad de modelos descubiertos por cada algoritmo según la cantidad de trazas.

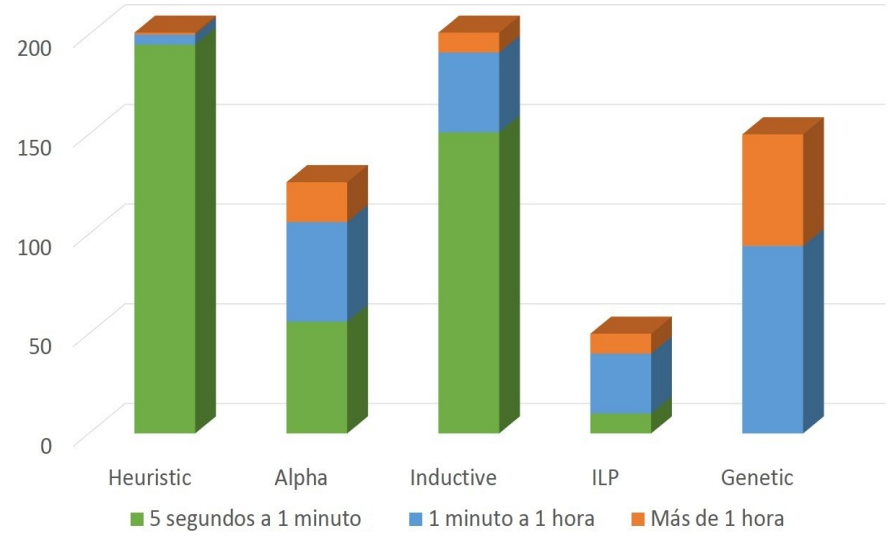
1 Modelos de procesos con baja estructuración.



Gráfica 2 Por ciento de modelos descubiertos por algoritmo.

Los algoritmos Heuristic Miner e Inductive Miner descubren todos los modelos de procesos, independientemente de la cantidad de trazas que poseen los registros de eventos. El algoritmo de menor rendimiento es el ILP. De manera general la cantidad de trazas presente en los registros de eventos no impacta significativamente en el proceso de descubrimiento.

Otro aspecto significativo de análisis es el tiempo de ejecución de cada algoritmo. En la siguiente gráfica se muestran los tiempos de ejecución de cada uno de ellos:



Gráfica 3 Tiempo de descubrimiento.

En la gráfica se observa cómo los algoritmos descubren los modelos de procesos en menos de 1 minuto en la mayoría de los casos. El algoritmo Genetic Miner demora un tiempo significativamente mayor en descubrir los modelos de proceso.

2.5 Ejemplo de los casos construidos

A continuación se ilustra el proceso de confección de 5 casos a partir de un registro de eventos para la base de conocimiento. Para ello se toma como ejemplo un registro de eventos que posee 500 trazas, 25 por ciento de Ruido, 75 por ciento de ruido por Intervalos, 1 And, 0 Xor, no posee Lazos ni Tareas ocultas y cuenta con un total de 7 actividades.

Para este registro de eventos el algoritmo Heuristic Miner descubrió el siguiente modelo:

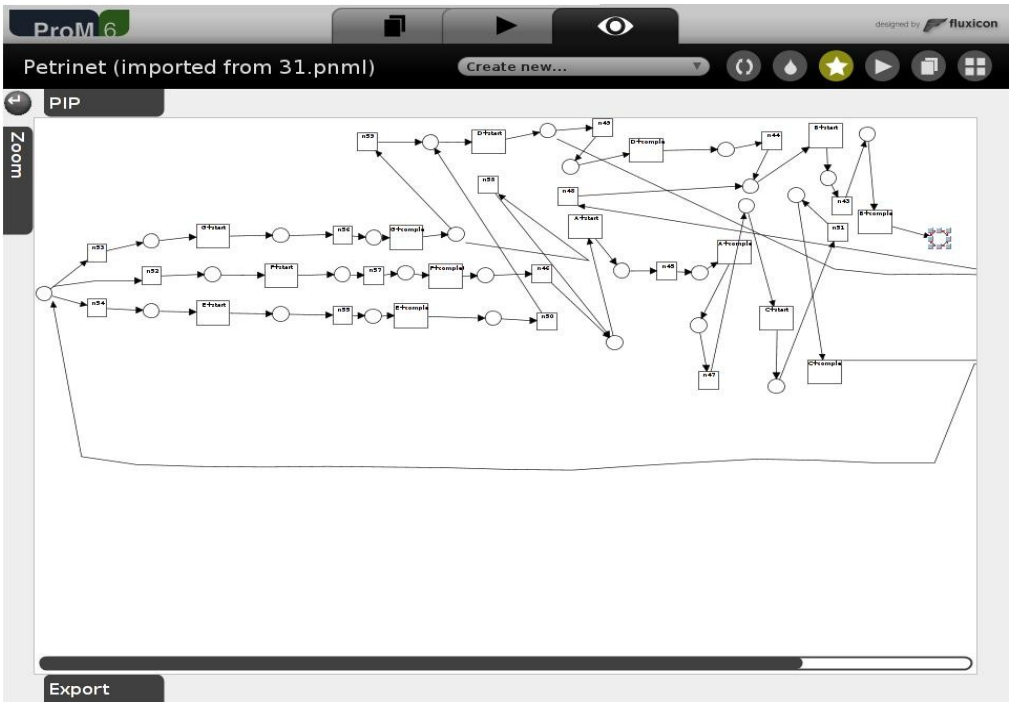


Figura 7: Modelo descubierto por el algoritmo Heuristic Miner.

La Figura 7 muestra un modelo estructurado y fácil de comprender. Este algoritmo tiene la particularidad de que no descubre los modelos en redes de Petri, sino en redes causales. Sin embargo las redes causales pueden ser convertidas a redes de Petri utilizando otro complemento dentro de ProM. Los valores de la evaluación y el tiempo de ejecución de las métricas por cada dimensión para el modelo descubierto, son los siguientes:

Dimensión	Evaluación	Tiempo (milisegundos)
Simplicidad	0.21	0.019
Aptitud	1	124.178
Precisión	0	5.756
Generalización	0.19	30006.63

Tabla 9: Evaluación del modelo descubierto.

El tiempo de ejecución del algoritmo para descubrir el modelo es de 15 segundos. Se puede observar que el algoritmo descubre el modelo con rapidez y los valores de evaluación de las métricas son aceptables en las dimensiones aptitud, generalización y simplicidad. Los tiempos de ejecución de las métricas son relativamente bajos, exceptuando la evaluación de la generalización.

Al aplicar el algoritmo Alpha Miner, sobre el registro de eventos descrito se descubre el modelo que se muestra en la Figura 8:

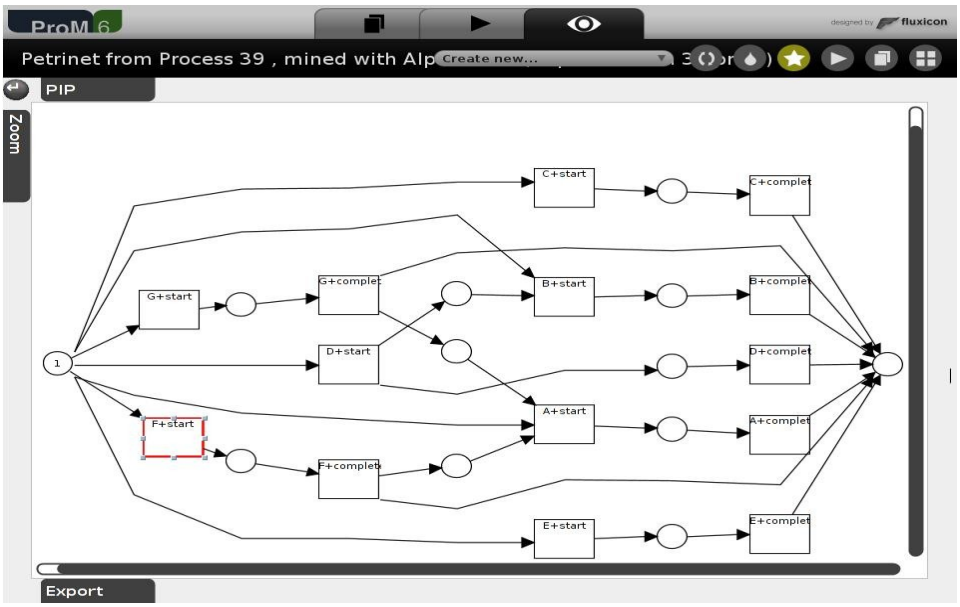


Figura 8: Modelo descubierto por el algoritmo Alpha Miner.

Es un modelo fácil de comprender por la baja dependencia existente entre las actividades. Los valores de la evaluación y el tiempo de ejecución de las métricas por cada dimensión, para el modelo descubierto, son los siguientes:

Dimensión	**Evaluación**	**Tiempo (milisegundos)**
Simplicidad	0.263	0.036
Aptitud	1	55.597
Precisión	1	55.952
Generalización	0	—

Tabla 10: Evaluación del modelo descubierto.

El tiempo de ejecución del algoritmo para descubrir el modelo de proceso asociado es de 5 horas. El algoritmo no descubre los modelos de procesos con la rapidez del algoritmo Heuristic Miner. El modelo descubierto muestra altos valores de aptitud y precisión, necesarios para el análisis de desviación y detección de anomalías. Los tiempos de ejecución de las métricas son mayores con respecto al algoritmo Heuristic Miner.

La ejecución del algoritmo Genetic Miner, sobre el registro de eventos genera el siguiente modelo:

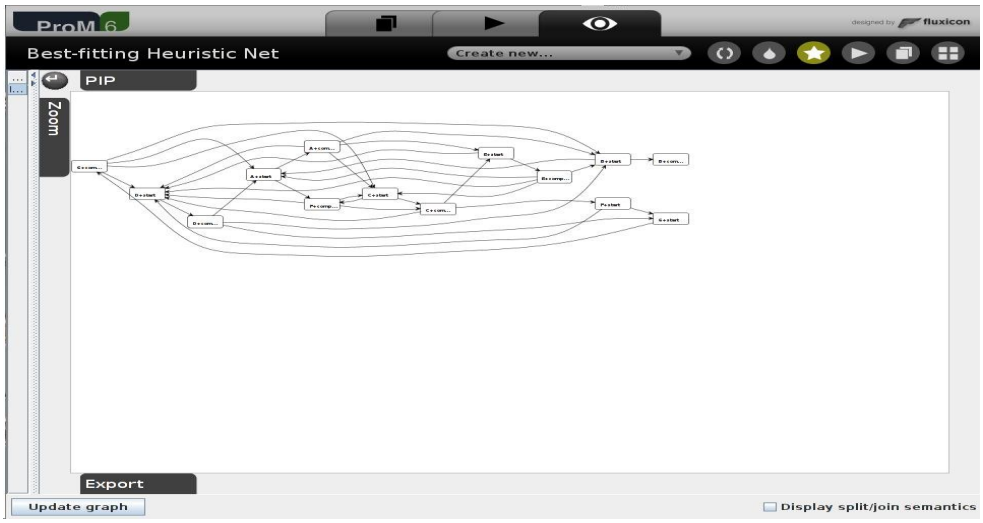


Figura 9: Modelo descubierto por el algoritmo Genetic Miner.

La Figura 9 muestra el modelo descubierto por el algoritmo Genetic Miner, este algoritmo tiene la particularidad de descubrir varios modelos de proceso según se crean los nuevos individuos. Otra de las particularidades del algoritmo es que no descubre los modelos en redes de Petri, pero estas pueden ser convertidas a redes de Petri utilizando un complemento dentro de la plataforma ProM. Los valores de evaluación de las métricas y el tiempo de ejecución de las mismas se muestran en la siguiente tabla:

Dimensión	**Evaluación**	**Tiempo (milisegundos)**
Simplicidad	0.14	0.436
Aptitud	1	255.897
Precisión	1	545.957
Generalización	0	___

Tabla 11: Evaluación del modelo descubierto.

El tiempo de ejecución del algoritmo para descubrir el modelo de proceso asociado fue de 3 horas. El algoritmo descubre modelos aptos y precisos. El tiempo de ejecución de las métricas es el más elevado de los algoritmos utilizados.

Mediante el algoritmo ILP se descubre el siguiente modelo de procesos:

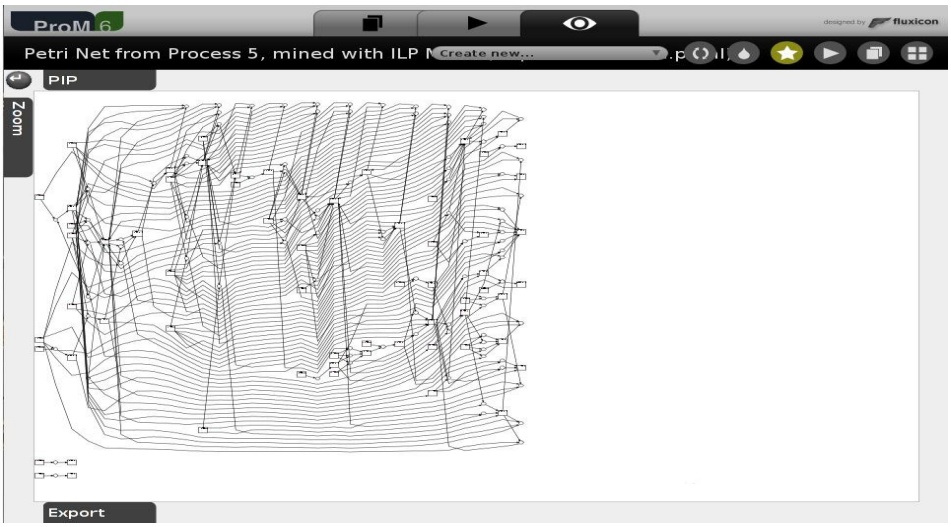


Figura 10: Modelo descubierto por el algoritmo ILP.

El modelo de procesos descubierto por el algoritmo que se muestra en la Figura 10 es un modelo poco estructurado, donde pueden observarse actividades que no se relacionan con el modelo general. El modelo resulta incomprensible a simple vista por la gran interrelación que se muestra entre las actividades. Los valores de la evaluación y el tiempo de ejecución de las métricas por cada dimensión, para el modelo descubierto, son los siguientes:

Dimensión	**Evaluación**	**Tiempo (milisegundos)**
Simplicidad	0.405	0.026
Aptitud	1	1.827
Precisión	0.04	16.883
Generalización	0	—

Tabla 12: Evaluación del modelo descubierto.

El tiempo de ejecución del algoritmo para descubrir el modelo de proceso asociado es de 12 minutos. La aptitud del modelo descubierto puede clasificarse como buena. Los tiempos de ejecución de las métricas son aceptables. En general ILP es un algoritmo que presenta dificultades a la hora de descubrir modelos de procesos sobre registros de eventos con muchas actividades y con gran dependencia entre ellas.

Utilizando el algoritmo Inductive Miner sobre el registro de eventos de ejemplo se obtiene el siguiente modelo de proceso:

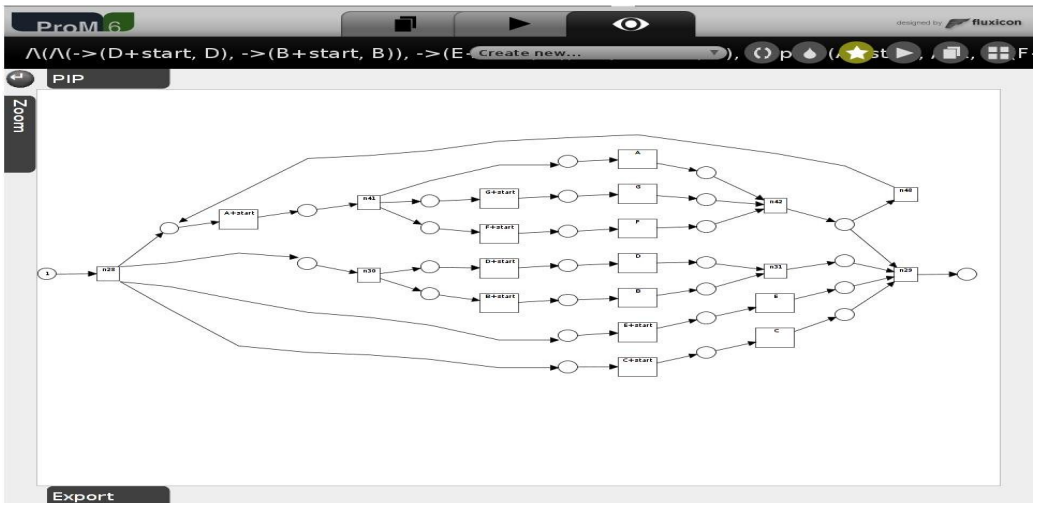


Figura 11: Modelo descubierto por el algoritmo Inductive Miner.

En la Figura 11 se observa el modelo descubierto por el algoritmo. Es un modelo simple, fácil de comprender y bien estructurado. Los valores de evaluación de las métricas y los tiempos de ejecución se muestran en la siguiente tabla:

Dimensión	**Evaluación**	**Tiempo (milisegundos)**
Simplicidad	0.228	0.003
Aptitud	1	1.353
Precisión	-	1.834
Generalización	0.55	3569.254

Tabla 13: Evaluación del modelo descubierto.

El tiempo de ejecución del algoritmo para descubrir el modelo de proceso asociado es de 10 segundos. El modelo descubierto presenta buena aptitud y generalización. Los tiempos de ejecución de las métricas son bajos. En general el algoritmo muestra buenos resultados ante

cualquier registro de eventos.

Después de finalizado el proceso de descubrimiento y la evaluación, se obtienen los datos para conformar los casos de la base de conocimiento, estos quedan conformados de la siguiente manera:

Algoritmo	ID	Trazas	And	Xor	Lazos	Tareas ocultas	Ruido	Intervalos	Total de actividades	Simplicidad	Aptitud	Precisión	Generalización
Heuristic Miner	1	500	1	0	0	0	25	75	7	0.21	1	0	0.19
Alpha Miner	2	500	1	0	0	0	25	75	7	0.263	1	1	0
Genetic Miner	3	500	1	0	0	0	25	75	7	0.14	1	1	0
ILP	4	500	1	0	0	0	25	75	7	0.405	1	0.04	0
Inductive Miner	5	500	1	0	0	0	25	75	7	0.228	1	__	0.55

Tabla 14: Ejemplo de casos confeccionados

2.6 Conclusiones parciales

La metodología seguida para la generación de la base de conocimiento permite generar registros de eventos con características de entornos reales, descubrir los modelos de procesos asociados a estos y evaluar los modelos descubiertos. La experimentación permite combinar en los registros de eventos las características seleccionadas. Las propiedades de las métricas descritas facilitan su selección por cada una de las dimensiones de calidad. La base de conocimiento contiene información relevante que permite aplicar técnicas de clasificación. Para la gestión de la base de conocimiento se implementa un complemento que se integra a la plataforma ProM, lo cual se aborda con mayor detalle en el siguiente capítulo.

Capítulo 3: Implementación y validación de la solución.

3.1 Introducción

En el presente capítulo se describe la arquitectura del complemento que se integra a la plataforma ProM y sus principales funcionalidades. Se muestran sus vistas y resultados. Además se visualizan los resultados obtenidos a partir del análisis de los datos proporcionados por la base de conocimiento. Se valida la solución propuesta mediante la aplicación de las técnicas de recomendación “evaluación empírica” y “clasificación” utilizando la base de conocimiento. Finalmente se comparan los resultados brindados por ambas técnicas.

3.2 Complemento de ProM para la gestión de la base de conocimiento

Se desarrolla un complemento para la plataforma ProM que permite gestionar la base de conocimiento creada. Esto permite interactuar con los casos almacenados en la misma. El complemento sirve de soporte al proceso de recomendación, permitiendo consultar los casos existentes, las características de los mismos, los valores de las métricas y los algoritmos utilizados en el descubrimiento. También brinda un conjunto de indicadores estadísticos sobre los casos almacenados. Además permite importar y exportar los casos de la base de conocimiento en formato CSV.

El complemento se desarrolla utilizando el lenguaje de programación Java y se integra a la plataforma ProM, ya que esta es la plataforma líder dentro de la minería de proceso. ProM brinda una interfaz amigable donde se integran múltiples complementos, todos vinculados con la minería de procesos. Integrar la base de conocimiento a la misma facilita el proceso de consulta y utilización.

El complemento facilita a los investigadores del área consultar la información contenida en la base de conocimiento. La interacción con la base de conocimiento permite que se realice una mejor selección de los algoritmos de descubrimiento.

Para persistir los datos se utiliza una base de datos objetual embebida en el complemento. La base de datos objetual permite almacenar la información en forma de objetos. Para ello se utiliza Oracle Berkerley DB como sistema gestor de base de datos objetual.

En la Figura 12 se muestra una vista de los componentes presentes en el complemento.

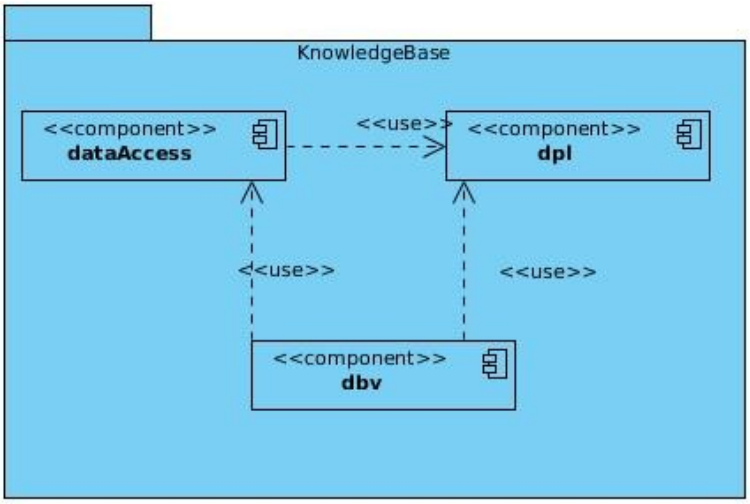


Figura 12: Diagrama de componentes del complemento.

El complemento está conformado por tres componentes. El componente dataAccess es el encargado de gestionar el acceso a la base de datos objetual. El componente dpl agrupa las clases que almacenan los objetos. Por su parte el componente dvp se encarga de la visualización de la base de conocimiento en la plataforma ProM.

La base de datos objetual está conformada por tres clases, las cuales son las encargadas de construir y almacenar los objetos con la información de cada caso. Este tipo de base de datos tiene la peculiaridad que el modelo de datos se representa como un diagrama de clases. En la Figura 13 se observa el diagrama de clases que representa la base de datos objetual.

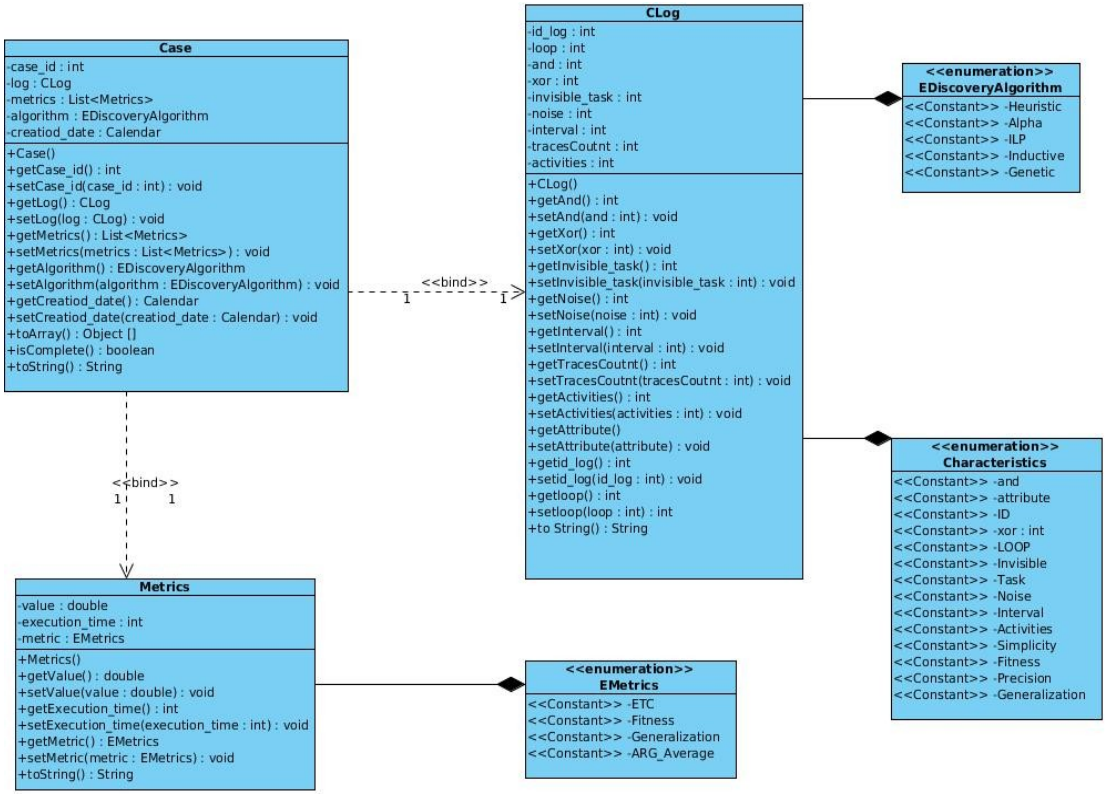
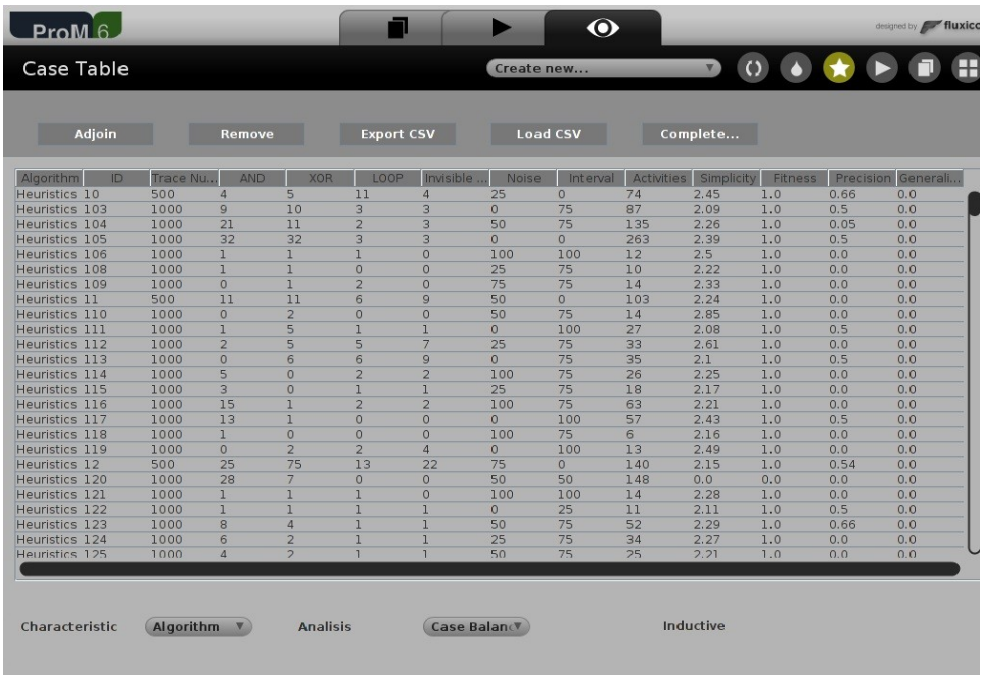


Figura 13: Diagrama de clases de la base de datos objetual.

El diagrama de clases muestra la relación existente entre las clases que componen la base de datos objetual. Cada clase representa uno de los objetos que componen un caso de la base de conocimiento. Las métricas para realizar la evaluación, los algoritmos utilizados y las características no varían en el tiempo por lo que se declaran como clases tipo enum.

El complemento permite visualizar en una tabla los casos existentes en la base de conocimiento. Para cada caso se muestra, su identificador, el algoritmo utilizado, las características del registro de eventos y los valores de la evaluación de las métricas. En la siguiente figura se muestra una vista general del complemento.



Algorithm	ID	Trace Nu...	AND	XOR	LOOP	Invisible ...	Noise	Interval	Activities	Simplicity	Fitness	Precision	Generali...
Heuristics	10	500	4	5	11	4	25	0	74	2.45	1.0	0.66	0.0
Heuristics	103	1000	9	10	3	3	0	75	87	2.09	1.0	0.5	0.0
Heuristics	104	1000	21	11	2	3	50	75	135	2.26	1.0	0.05	0.0
Heuristics	105	1000	32	32	3	3	0	0	263	2.39	1.0	0.5	0.0
Heuristics	106	1000	1	1	1	0	100	100	12	2.5	1.0	0.0	0.0
Heuristics	108	1000	1	1	0	0	25	75	10	2.22	1.0	0.0	0.0
Heuristics	109	1000	0	1	2	0	75	75	14	2.33	1.0	0.0	0.0
Heuristics	11	500	11	11	6	9	50	0	103	2.24	1.0	0.0	0.0
Heuristics	110	1000	0	2	0	0	50	75	14	2.85	1.0	0.0	0.0
Heuristics	111	1000	1	5	1	1	0	100	27	2.08	1.0	0.5	0.0
Heuristics	112	1000	2	5	5	7	25	75	33	2.61	1.0	0.0	0.0
Heuristics	113	1000	0	6	6	9	0	75	35	2.1	1.0	0.5	0.0
Heuristics	114	1000	5	0	2	2	100	75	26	2.25	1.0	0.0	0.0
Heuristics	115	1000	3	0	1	1	25	75	18	2.17	1.0	0.0	0.0
Heuristics	116	1000	15	1	2	2	100	75	63	2.21	1.0	0.0	0.0
Heuristics	117	1000	13	1	0	0	0	100	57	2.43	1.0	0.5	0.0
Heuristics	118	1000	1	0	0	0	100	75	6	2.16	1.0	0.0	0.0
Heuristics	119	1000	0	2	2	4	0	100	13	2.49	1.0	0.0	0.0
Heuristics	12	500	25	75	13	22	75	0	140	2.15	1.0	0.54	0.0
Heuristics	120	1000	28	7	0	0	50	50	148	0.0	0.0	0.0	0.0
Heuristics	121	1000	1	1	1	0	100	100	14	2.28	1.0	0.0	0.0
Heuristics	122	1000	1	1	1	1	0	25	11	2.11	1.0	0.5	0.0
Heuristics	123	1000	8	4	1	1	50	75	52	2.29	1.0	0.66	0.0
Heuristics	124	1000	6	2	1	1	25	75	34	2.27	1.0	0.0	0.0
Heuristics	125	1000	4	2	1	1	50	75	25	2.21	1.0	0.0	0.0

Figura 14: Visualización de la base de conocimiento.

El complemento permite cargar los casos en formato CSV. Esto brinda la posibilidad de aumentar los casos almacenados en la base de conocimiento y así enriquecer la misma. Para incorporar los datos a la base de conocimiento se extrae la información del fichero CSV. La información almacenada en el fichero debe tener la estructura de un caso de la base de conocimiento para poder ser incorporado a la misma. En la Figura 15 se muestra la interfaz que se ocupa de cargar los ficheros CSV.

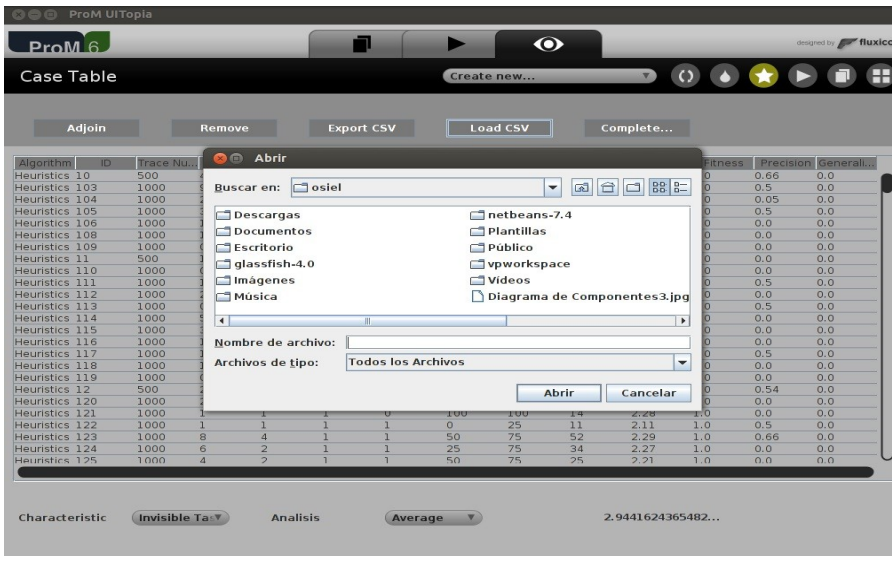


Figura 15: Cargar base de conocimiento en formato CSV.

El complemento permite adicionar un nuevo caso a la base de conocimiento. Para ello se muestra un formulario donde es posible agregar las características del caso, ruido, intervalos de ruido, and, xor, lazos, cantidad de actividades, cantidad de trazas, valores de las métricas por cada dimensión y algoritmo utilizado.

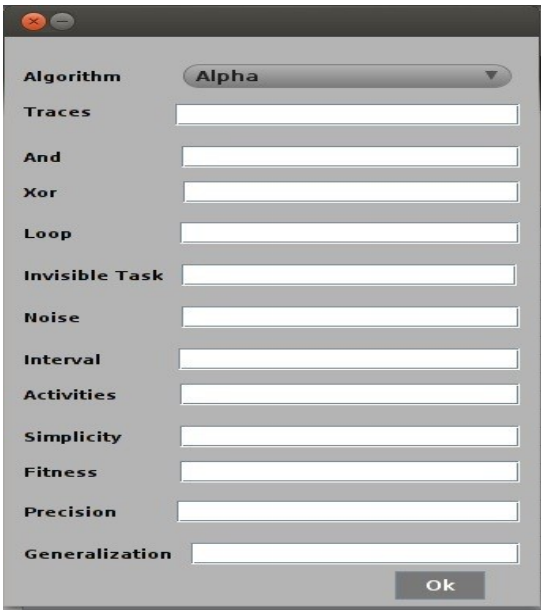


Figura 16: Adicionar un nuevo caso

Otra funcionalidad del complemento es exportar la base de conocimiento a un fichero CSV. Lo cual permite extraer los casos y poder utilizarlos para realizar el entrenamiento y clasificación de la base de conocimiento. El complemento además permite eliminar un caso de la base de conocimiento. Para eliminar un caso se selecciona en la tabla y se acciona el botón eliminar en la parte superior de la ventana. Este es eliminado de la base de datos objetual y de la tabla donde se muestra.

A partir de la información presente en la base de conocimiento es posible visualizar un conjunto de indicadores estadísticos: media, moda y mediana. Esto permite detectar diferentes características en los casos almacenados. Los indicadores se implementaron para permitir al usuario observar en el complemento la visualización de sus resultados. Estos indicadores estadísticos pueden ser aplicados a las características del registro de eventos y a la evaluación de las métricas.

En la siguiente figura se muestra cómo se visualiza en el complemento la funcionalidad de análisis estadístico para los indicadores media, moda y mediana. Es muy similar para los tres, solo cambian los textos que nombran la característica y el indicador seleccionados.

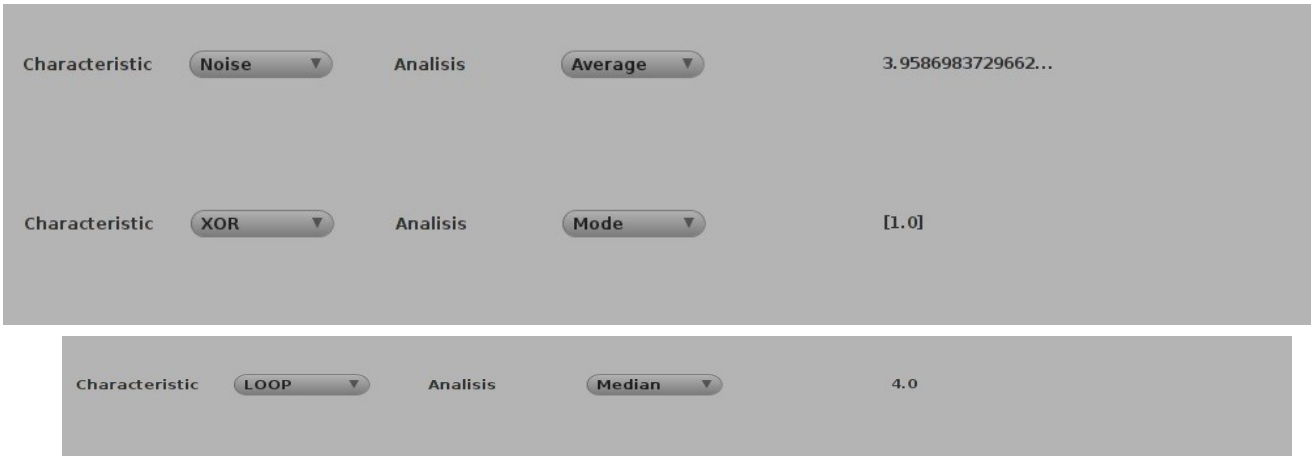


Figura 17: Indicadores estadísticos

Los dos análisis adicionales que permite el complemento son el balance de clases y la cantidad de casos incompletos en la base de conocimiento. El balance de clases permite mostrar el algoritmo que mayor cantidad de modelos descubiertos posee en la base de conocimiento. Por su parte la opción “cantidad de casos incompletos” muestra la cantidad de casos que se encuentran en la base de conocimiento donde no aparecen todos los valores de las métricas.

El balance de clases se visualiza en la parte inferior del complemento, en la Figura 18 se observa la visualización del mismo:

Figura 18: Balance de clases.

La cantidad de casos incompletos se implementa para que al accionar el botón se muestre un mensaje con la información, tal y como se muestra en el ejemplo:

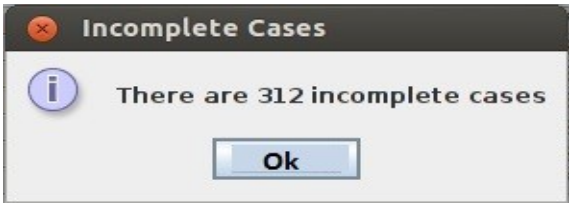


Figura 19: Casos incompletos.

Las funcionalidades presentes en el complemento, permiten manejar y visualizar los casos dentro de la base de conocimiento.

3.3 Validación de la propuesta de solución

Para validar la base de conocimiento se aplicaron las técnicas de evaluación empírica y clasificación a un conjunto de registros de eventos, con el objetivo de comparar los resultados brindados por ambas. Esto permite comprobar si el tiempo que se utiliza para realizar la recomendación utilizando la base de conocimiento es menor al de la evaluación empírica y si los algoritmos recomendados son similares.

Para realizar este proceso se seleccionaron cinco registros de eventos con diferentes características, los cuales se muestran en la tabla 15.

ID	Trazas	And	Xor	Lazo	Ruido	Intervalos	Tareas invisibles	Total de actividades
1	1000	1	2	1	25	75	2	17
2	1500	2	1	2	0	75	2	18
3	500	4	1	0	100	25	0	25
4	500	6	9	2	25	25	2	25
5	500	2	4	1	0	50	0	26

Tabla 15: Características de los registros de eventos a minar.

3.3.1 Evaluación empírica

Con los cinco registros de eventos seleccionados se realizó la evaluación empírica para obtener en cada caso el algoritmo más adecuado. Para ello se obtuvieron los modelos de procesos asociados utilizando los algoritmos utilizados en la confección de la base de conocimiento.

El tiempo que demora cada algoritmo en descubrir el modelo de procesos se muestra en la siguiente tabla:

Algoritmo	1	2	3	4	5
Heuristics Miner	45 s	10 s	20 s	25 s	5 s
Inductive Miner	10 s	10 s	2 m	15 s	10 s
ILP	11 m	47 m	12 h 50 m 10 s	12 h 50 m 55 s	25 m
Alpha Miner	2 m	25 m	1 h	12 h 10 m	45 s
Genetic Miner	12 h	9 m	11 h 25 m 45 s	12 h	12 h 50 m

Tabla 16: Tiempo de descubrimiento de los algoritmos.

El siguiente paso en la realización de la evaluación empírica es evaluar los modelos de procesos con las métricas de calidad. Para ello se utilizaron las métricas seleccionadas para la confección de la base de conocimiento. El resultado obtenido para cada dimensión se muestra en la siguiente tabla:

Algoritmos	Registro de eventos	Simplicidad	Aptitud	Precisión	Generalización
Heuristic Miner	1	0.23	1	0	0
	2	0.2	1	0.5	0
	3	0.2	1	0.5	0
	4	0.23	1	0.55	0
	5	0.2	1	0.5	0
Inductive Miner	1	0.44	1	0.5	0
	2	0.25	1	0.25	0
	3	0.31	1	0.42	0
	4	0	0	0	0
	5	0.1	1	0.2	0
ILP	1	0.44	4	0.5	0
	2	0.25	1	0.25	0
	3	0.31	1	0.42	0
	4	0	0	0	0
	5	0.1	1	0.2	0
Alpha Miner	1	0.46	1	0.73	0
	2	0.229	1	1	0
	3	0.229	1	0.76	0

	4	0	0	0	0
	5	0.37	1	1	0
Genetic Miner	1	0	0	0	0
	2	0.1	1	0.87	0
	3	0	0	0	0
	4	0	0	0	0
	5	0.1	1	0.4	0

Tabla 17: Resultado de la evaluación de las métricas de calidad.

El tiempo de ejecución de las métricas fue en todos los casos de cinco horas por algoritmo. La ejecución de la evaluación se realiza utilizando un algoritmo con los cinco registros de eventos y modelos de procesos asociados.

Al concluir el proceso de descubrimiento y evaluación de los registros de eventos con los modelos de procesos se tienen los datos suficientes para recomendar el algoritmo a utilizar a partir de la evaluación empírica. Para ello se analizan los valores obtenidos por las métricas de calidad y el tiempo de ejecución de los algoritmos. Para cada uno de los registros de eventos minados, la técnica recomienda utilizar los siguientes algoritmos:

Registro de Eventos	**Algoritmo**
1	Alpha Miner
2	Alpha Miner
3	Alpha Miner
4	Heuristic Miner
5	Alpha Miner

Tabla 18: Recomendación utilizando la técnica de evaluación empírica.

3.3.2 Clasificación a partir de la base de conocimiento construida

Después de obtener los resultados aplicando la evaluación empírica se realiza la recomendación

aplicando la técnica de clasificación con los cinco registros de eventos y la base de conocimiento construida.

El primer paso para aplicar la técnica de clasificación es entrenar los clasificadores utilizados con la base de conocimiento desarrollada. Para ello se utilizan ocho algoritmos de clasificación implementados en la plataforma ProM, como un complemento que permite realizar recomendación, aplicando técnicas de clasificación. La base de conocimiento posee un total de 796 casos en el momento de realizar el entrenamiento. El tiempo que se cronometra en el entrenamiento para cada clasificador y la cantidad de casos clasificados se muestra en la siguiente tabla:

Algoritmos	**Tiempo de Clasificación**	**Índice de casos clasificados**
MultiClassClassifier	1 m 30 s	795
MLP	5 m 42 s	794
Simple Logic	1 m 31 s	794
Logistic	1 s	793
FilteredClassifier	1 s	792
ClassificationViaRegression	1 s	792
PART	1 s	792
J48	1 s	792

Tabla 19: Tiempo de entrenamiento y cantidad de casos correctamente clasificados.

Si se añaden nuevos casos a la base de conocimiento se vuelven a entrenar los clasificadores. De manera general los primeros ocho clasificadores de la tabla clasificaron correctamente el 99.24 % de los casos existentes. Después de entrenar la base de conocimiento se realiza la clasificación para los nuevos casos permitiendo comprobar el resultado de la recomendación realizada mediante la evaluación empírica y la clasificación.

Para el registro de eventos # 1 los resultados de los clasificadores fueron los siguientes:

Algoritmo de clasificación	**Algoritmo recomendado por el clasificador**
ClassificationViaRegression	Alpha Miner
FilteredClassifier	Alpha Miner
J48	Alpha Miner
Logistic	Heuristics Miner
MLP	Alpha Miner
MultiClassClassifier	Alpha Miner
PART	Alpha Miner
Simple Logic	Alpha Miner

Tabla 20: Resultados de la recomendación utilizando la técnica de clasificación para el registro de eventos 1.

Para este registro de eventos el resultado de los clasificadores coincide con los resultados de la evaluación empírica. El tiempo de ejecución que utilizó cada clasificador para recomendar un algoritmo de descubierto fue de un segundo en cada caso. En la siguiente tabla se muestra el tiempo que demora aplicar la técnica de evaluación empírica y la técnica de clasificación sobre el registro de eventos # 1.

Técnica de recomendación	**Tiempo total de la recomendación**
Evaluación empírica	10 h 14 m 15 s
Clasificación	8 m 48 s

Tabla 21: Comparación de los tiempos de ejecución de las técnicas de recomendación analizadas.

Para el registro de eventos # 2 se obtienen los siguientes resultados al aplicar los clasificadores:

Algoritmo de clasificación	Algoritmo recomendado por el clasificador
ClassificationViaRegression	Alpha Miner
FilteredClassifier	Alpha Miner
J48	Alpha Miner
Logistic	Heuristics Miner
MLP	Alpha Miner
MultiClassClassifier	Alpha Miner
PART	Alpha Miner
Simple Logic	Alpha Miner

Tabla 22: Resultados de la recomendación utilizando la técnica de clasificación para el registro de eventos 2.

Para el registro de eventos analizado, la técnica de clasificación muestra los mismos resultados que la evaluación empírica. El tiempo que se utiliza para realizar la recomendación por cada técnica se muestra en la siguiente tabla:

Técnica de recomendación	Tiempo total de la recomendación
Evaluación empírica	9 h 13 m 24 s
Clasificación	8 m 48 s

Tabla 23: Comparación de los tiempos de ejecución de las técnicas de recomendación analizadas.

Para el registro de eventos # 3 se obtienen los siguientes resultados:

Algoritmo de clasificación	Algoritmo recomendado por el clasificador
ClassificationViaRegression	Alpha Miner
FilteredClassifier	Alpha Miner
J48	Alpha Miner
Logistic	Heuristics Miner
MLP	Alpha Miner
MultiClassClassifier	Alpha Miner
PART	Alpha Miner
Simple Logic	Alpha Miner

Tabla 24: Resultados de la recomendación utilizando la técnica de clasificación para el registro de eventos 3.

Los resultados al aplicar la técnica de clasificación son similares a los arrojados por la evaluación empírica. El tiempo que demoran ambas técnicas en realizar la recomendación se muestra en la siguiente tabla:

Técnica de recomendación	Tiempo total de la recomendación
Evaluación empírica	20 h 28 m 30 s
Clasificación	8 m 48 s

Tabla 25: Comparación de los tiempos de ejecución de las técnicas de recomendación analizadas.

Al aplicar la técnica de clasificación al registro #4 se obtienen los siguientes resultados:

Algoritmo de clasificación	Algoritmo recomendado por el clasificador
ClassificationViaRegression	Heuristics Miner
FilteredClassifier	Alpha Miner
J48	Alpha Miner
Logistic	Heuristics Miner
MLP	Alpha Miner
MultiClassClassifier	Heuristics Miner
PART	Alpha Miner
Simple Logic	Inductive Miner

Tabla 26: Resultados de la recomendación utilizando la técnica de clasificación para el registro de eventos 4.

Los resultados observados al aplicar la técnica de clasificación al registro de eventos es el más diverso. En algunos casos el algoritmo recomendado por la clasificación coincide con el recomendado por la evaluación empírica. Las restantes recomendaciones del clasificador siguen siendo los mejores algoritmos recomendados por la evaluación empírica. Por tanto, utilizar cualquiera de esos algoritmos brinda buenos resultados al obtener modelos de procesos asociados.

El tiempo de ejecución de cada una de las técnicas se muestra a continuación:

Técnica de recomendación	Tiempo total de la recomendación
Evaluación empírica	31 h 1 m 35 s
Clasificación	8 m 48 s

Tabla 27: Comparación de los tiempos de ejecución de las técnicas de recomendación analizadas.

El último de los registros de eventos clasificados fue el # 5, los resultados obtenidos al aplicar la

técnica de clasificación fueron:

Algoritmo de clasificación	**Algoritmo recomendado por el clasificador**
ClassificationViaRegression	Alpha Miner
FilteredClassifier	Alpha Miner
J48	Alpha Miner
Logistic	Heuristics Miner
MLP	Alpha Miner
MultiClassClassifier	Alpha Miner
PART	Alpha Miner
Simple Logic	Alpha Miner

Tabla 28: Resultados de la recomendación utilizando la técnica de clasificación para el registro de eventos 4.

Los resultados observados al analizar el registro de eventos coinciden con los arrojados por la evaluación empírica. El tiempo de ejecución de las técnicas se muestra a continuación:

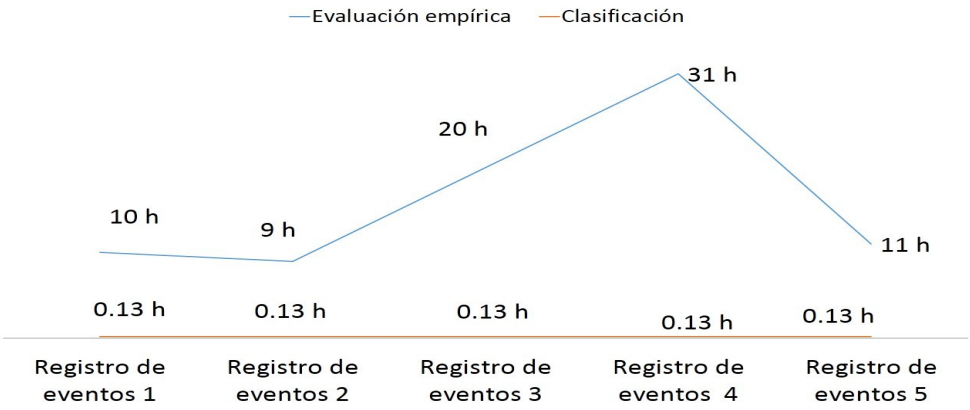
Técnica de recomendación	**Tiempo total de la recomendación**
Evaluación empírica	11 h 55 m 51 s
Clasificación	8 m 48 s

Tabla 29: Comparación de los tiempos de ejecución de las técnicas de recomendación analizadas.

De manera general los resultados arrojados por la técnica de clasificación coinciden con los resultados de la evaluación empírica. Lo cual valida que los algoritmos recomendados por los clasificadores tiene la calidad suficiente para aplicarse en un entorno determinado. En los casos donde el resultado del clasificador no coincidió con el algoritmo recomendado por la evaluación empírica, su recomendación coincide con el segundo o tercer algoritmo con mejor evaluación de calidad.

La principal diferencia entre ambas técnicas es el tiempo que demoran en realizar la recomendación.

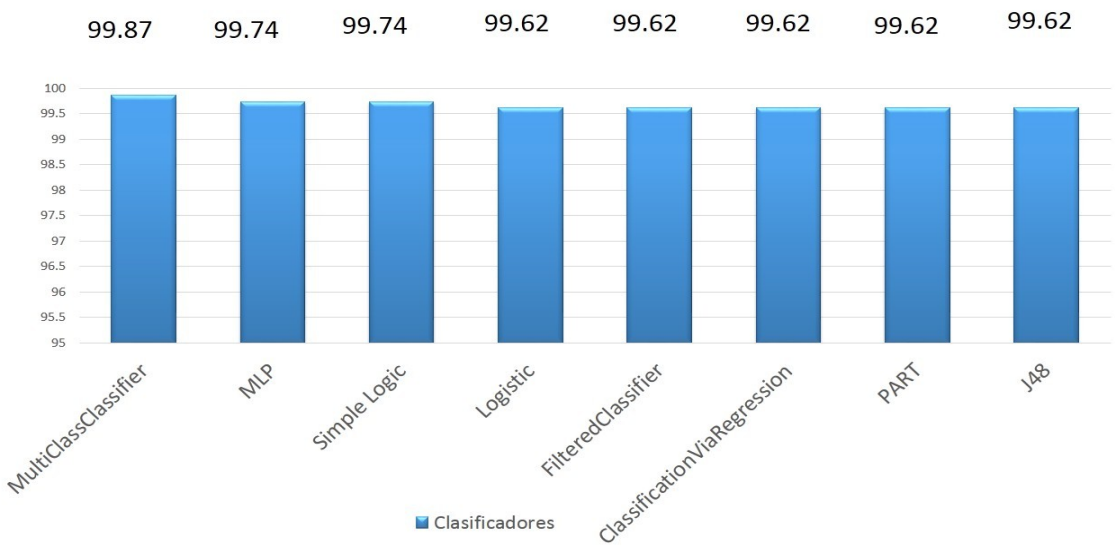
En la siguiente gráfica se muestran los tiempos por cada uno de los algoritmos al aplicar ambas técnicas.



Gráfica 4 Tiempo de ejecución de las técnicas de clasificación.

La técnica de clasificación utilizando la base de conocimiento construida mejora significativamente los tiempos de obtención de los resultados. Para cada registro de eventos minado la técnica no supera los nueve minutos. Esto mejora significativamente el tiempo de realización de la recomendación, lo que facilita su uso en diferentes entornos, mejorando así el descubrimiento de modelos de proceso.

Los clasificadores utilizados para realizar la recomendación de los algoritmos de descubrimiento muestran buenos resultados. Durante el proceso de entrenamiento clasificaron correctamente la mayoría de los casos almacenados en la base de conocimiento. En la siguiente gráfica se muestra el por ciento de casos correctamente clasificados por cada clasificador.



Gráfica 5 Por ciento de casos correctamente clasificados.

Los clasificadores utilizados muestran un buen rendimiento durante el entrenamiento de la base de conocimiento, permitiendo obtener mejores resultados en la recomendación. El elevado por ciento de casos correctamente clasificados valida la robustez de la base de conocimiento para utilizarla en la recomendación de algoritmos de descubrimiento.

3.4 Conclusiones parciales

El complemento desarrollado para la plataforma ProM permite la gestión de la información contenida en la base de conocimiento. La validación de la base de conocimiento desarrollada aplicando sobre ella algoritmos de clasificación y comparando sus resultados con los obtenidos mediante la evaluación empírica permitió comprobar el rendimiento de la técnica. Los resultados obtenidos son alentadores, lo que permite comprobar que la aplicación de clasificadores a la base de conocimiento disminuye el tiempo de la recomendación de algoritmos de descubrimiento de proceso.

Conclusiones generales

A partir de la investigación realizada y los resultados obtenidos puede arribarse a las siguientes conclusiones:

- Las principales características que afectan el rendimiento de los algoritmos de descubrimiento así como la evaluación de sus modelos son: ruido, tareas ocultas, tareas duplicadas, alternativas no libres y lazos. Por ello es necesario tenerlas en consideración para la selección de algoritmos de descubrimiento de modelos de proceso.
- La descripción de las métricas de calidad por cada una de las dimensiones permite determinar qué propiedades deben poseer para evaluar los modelos de proceso.
- Seleccionar las métricas y las características de los registros de eventos permite crear registros de eventos diversos con comportamiento de entornos reales.
- El diseño del experimento para la generación de los casos permite combinar las características deseadas, obteniendo registros de eventos complejos con características de entornos reales.
- La base de conocimiento construida recoge los rasgos necesarios, para aplicar técnicas de clasificación, para la recomendación de algoritmos de descubrimiento.
- El complemento desarrollado permite gestionar los casos de la base de conocimiento, permitiendo la interacción con la misma.
- La comparación de los resultados de la recomendación al utilizar la clasificación sobre la base de conocimiento y la evaluación empírica mostró mejores tiempos para la técnica de clasificación.

Recomendaciones

Con el objetivo de darle continuidad a la investigación y mejorar los resultados de la propuesta de solución se recomienda:

1. Incrementar el número de casos de la base de conocimiento mediante el incremento de la cantidad y diversidad de algoritmos de descubrimiento.
2. Incorporar casos provenientes de registros de eventos de entornos reales.

Referencias bibliográficas

AALST, Wil M. P. van der, 2011, Mining Additional Perspectives. In:*Process Mining* [online]. Springer Berlin Heidelberg. p.215–240. [Accessed 5 December 2013]. ISBN 978-3-642-19344-6, 978-3-642-19345-3. Available from: http://link.springer.com/chapter/10.1007/978-3-642-19345-3_8

AALST, W M P van der, WEIJTERS, A J M M y MARUSTER, L, 2004. Workflow Mining: Discovering process models from event logs. En: *IEEE Transactions on Knowledge and Data Engineering* [en línea]. 2004. Vol.16, no.9, pp.1128–1142. DOI 10.1109/TKDE.2004.47. Disponible desde: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1316839.

AALST, W.M.P. Van Der, 2012. *Decomposing process mining problems using passages* [en línea]. Hamburg: s.n. 33rd International Conference on Application and Theory of Petri Nets and Concurrency, PETRI NETS 2012. ISBN 03029743 (ISSN); 9783642311307 (ISBN). Disponible desde: http://hinari-gw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-84862503604&partnerID=40&md5=dca5ca92267c6ff5800170e6769321fc.

AALST, W.M.P. Van Der, RUBIN, V., VERBEEK, H.M.W., VAN DONGEN, B.F., KINDLER, E. y GÜNTHER, C.W., 2010. Process mining: A two-step approach to balance between underfitting and overfitting. En: *Software and Systems Modeling* [en línea]. 2010. Vol.9, no.1, pp.87–111. Disponible desde: http://www.springerlink.com/index/U43V780550278H4L.pdf.

AALST, W M P Van der, 2007. Towards an Evaluation marco de trabajo for Process Mining Algorithms. En: *BPM Center Report* [en línea]. 2007. Disponible desde: BPMcenter.org.

AALST W. M. P, 2013, Mine your own business: Using process minig to turn big data into real value. Eindhoven. In: 2013.

AALST, W M P van der, 2011, *Process Mining. Discovery, Conformance and Enhancement of Business Processes*. Springer, Heidelberg, Dordrecht, London et. al. ISBN 978-3-642-19344-6. 0179

ADRIANSYAH, A., VAN DONGEN, B.F. and VAN DER AALST, W.M.P., 2011, Conformance checking using cost-based fitness analysis. In:*Proceedings - IEEE International Enterprise Distributed Object Computing Workshop, EDOC* [online]. Helsinki. 2011. p.55–64. 15th IEEE International EDOC Enterprise Computing Conference, EDOC 2011. ISBN 15417719 (ISSN); 9780769544250 (ISBN). Available from: http://hinari-gw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-80054889108&partnerID=40&md5=cc84c3cb85aded396d7dd5548b9df336

AILENEI, I. M., 2011, *Process Mining Tools: A Comparative Analysis* [online]. EINDHOVEN UNIVERSITY OF TECHNOLOGY. [Accessed 7 September 2012]. Available from: http://alexandria.tue.nl/extra1/afstversl/wsk-i/ailenei2011.pdf

AILENEI, Irina, ROZINAT, Anne, ECKERT, Albert and AALST, Wil M. P., 2012, Definition and Validation of Process Mining Use Cases. In:*Business Process Management Workshops* [online]. Springer Berlin Heidelberg. p.75–86. Lecture Notes in Business Information Processing. [Accessed 7 September 2012]. ISBN 978-3-642-28108-2. Available from: http://www.springerlink.com/content/n08155ghg567h528/abstract/

ALBIOL, Roses, 2004, *Hibernate_Introduccion*. 2004.

BAE, J., LIU, L., CAVERLEE, J., ZHANG, L. J. y BAE, H., 2007. Development of distance measures for process mining, discovery and integration. En: *International Journal of Web Services Research (IJWSR)* [en línea]. 2007. Vol.4, no.4, pp.1–17. [Accedido 17 septiembre 2012]. Disponible desde: http://hinari-gw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-51749096433&partnerID=40&md5=e4575b4d1762321a3038349373dd410a.

BURATTIN, Andrea and SPERDUTI, Alessandro, 2011, PLG: Framework for the Generation of Business Process Models and Their Execution Logs. In:*Business Process Management Workshops* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg. p.214–219. [Accessed 7 February 2014]. ISBN 978-3-642-20510-1, 978-3-642-20511-8. Available from: http://www.processmining.it/sw/plg

CARRETERO PASTOR and MARÍN, Pastor, 2011, *BERKELEY DB*. 2011.

COSTILLA, Carmen, 2009, *Características Objeto-Relacionales de: Sistema de Gestión de Bases de Datos Oracle*. 2009.

DE MEDEIROS, A.K.A., WEIJTERS, A.J.M.M. and VAN DER AALST, W.M.P., 2007, Genetic process mining: An experimental evaluation. *Data Mining and Knowledge Discovery*. 2007. Vol.14, no.2, p.245–304.

DE WEERDT, J., DE BACKER, M., VANTHIENEN, J. and BAESENS, B., 2011, *A critical evaluation study of model-log metrics in process discovery* [online]. Hoboken, NJ. 8th International Workshops and Education Track on Business Process Management, BPM 2010. ISBN 18651348 (ISSN); 9783642205101 (ISBN). Available from: http://hinari-gw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-79957506648&partnerID=40&md5=b6c68c4b35a21eeb986f94f1ff2f1950DE

DE WEERDT, Jochen, BAESENS, Bart and VANTHIENEN, Jan, 2013, A Comprehensive Benchmarking marco de trabajo (CoBeFra) for conformance analysis between procedural process models and event logs in ProM. In:*Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining (CIDM 2013), part of the IEEE Symposium Series in Computational Intelligence 2013* [online]. 2013. [Accessed 7 December 2013]. Available from: http://eprints.qut.edu.au/57682/

DE WEERDT, J., DE BACKER, M., VANTHIENEN, J. y BAESENS, B., 2012. A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs. En:

InformationSystems [en línea]. 2012. Vol.37, no.7, pp.654-676. Disponible desde: http://hinari-gw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-84861093725&partnerID=40&md5=b73a289aad8600e57203262c9aebb872.

DELVAUX, C., FREITAS, J., ROGOVA, T., VANTHIENEN, Jan and BAESENS, Bart, 2013, Uncovering the relationship between event log characteristics and process discovery techniques. In:*Business Process Management Workshops* [online]. 2013. [Accessed 7 December 2013]. Available from: https://lirias.kuleuven.be/handle/123456789/412085

DE LEONI, Massimiliano y VAN DER AALST, Wil MP, 2013. Data-Aware Process Mining: Discovering Decisions in Processes Using Alignments. En: 2013.

FAYYAD, Usama M., PIATETSKY-SHAPIRO, Gregory, SMYTH, Padhraic and UTHURUSAMY, Ramasamy, 1996, Advances in knowledge discovery and data mining. [online]. 1996. [Accessed 7 December 2013]. Availablefrom: http://www.citeulike.org/group/2902/article/1550195

FUNDORA-RAMÍREZ OSIEL, 2013, Impacto de las características de un registro de evento en algoritmos de descubrimiento de la minería de proceso. In: VIII Peña Tecnológica. VI Taller Científico provincial de jóvenes de la especialidad de tecnologías y sistemas. 2013.

GOEDERTIER, Stijn, MARTENS, David, VANTHIENEN, Jan and BAESENS, Bart, 2009, Robust Process Discovery with Artificial Negative Events. *Journal of Machine LearningResearch*. 6 June 2009. Vol.10, no.6, p.1305–1340.

GÜNTHER, C W y AALST, W M P van der, 2007. Fuzzy Mining: Adaptive Process Simplification Based on Multi-Perspective Metrics. En: ALONSO, G, DADAM, P y ROSEMANN, M (eds.), *International Conference on Business Process Management (BPM 2007)* [en línea]. S.l.: Lecture Notes in Computer Science. Springer, Berlin. 2007. pp.328–343. Disponible desde: http://dl.acm.org/citation.cfm?id=1793114.1793145

HERRERA, Raykenler Yzquierdo, CASTRO, Rogelio Silverio, CORTÉS, Manuel Lazo and GRAÑA, Adrian Torres, 2012, Minería de procesos como herramienta para la auditoria. *Ciencias de la Información* [online]. 2012. [Accessed 13 April 2012]. Available from: http://intranet2.uci.cu/node/12558/postgrado/maestrias

H, YANG and DONGEN B.F, 2012, Estimating completeness of event logs. In: 2012.

HOU, Daqing, 2007, Studying the evolution of the Eclipse Java editor. In:*Proceedings of the 2007 OOPSLA workshop on eclipse technology eXchange* [online]. 2007. p.65–69. [Accessed 7 December 2013]. Available from: http://dl.acm.org/citation.cfm?id=1328293

INTERCHANGE, KNOWLEDGE, 1998, The DARPA knowledge sharing effort: Progress report. *Readings in Agents*. 1998. P.243.

JOLLIFFE, I. T., 2002. *Principal component analysis* [en línea]. S.l.: Wiley Online Library. [Accedido 17 septiembre 2012]. Disponible desde: http://onlinelibrary.wiley.com/mrw_content/esbs/articles/bsa501/image_n/bsa501.pdf

JAN CLAES and GEERT POELS, 2012a, Process Mining and the ProM Framework: An Exploratory Survey. In: *LNBIP* [online]. Springer. 2012. [Accessed 6 September 2012]. Available from: http://processmining.ugent.be/pdf/ClaesPoels2012BPI@BPM.pdf

JOCHEN DE WEERDT, MANU DE BACKER, JAN VANTHIENEN and BART BAESENS, 2012, A multi- dimensional quality assessment of state-of-the- art process discovery algorithms using real- life event logs. *Information Systems*. 5 March 2012. Vol.37, p.654–676.

KOURDI, Jeremy, 2008, *Estrategia: Claves para tomar decisiones en los negocios* [online]. Cuatro Media. [Accessed 7 December 2013]. Available from: http://virtual.urbe.edu/artectexto/TEL/TEL-031/TEL-031-008/texto.pdf

LAKSHMANAN, Geetika and KHALAF, Rania, 2012, Leveraging Process Mining Techniques to Analyze Semi-Structured Processes. *IT Professional*. 2012. Vol.99, no.PrePrints, p.1–1. DOI 10.1109/MITP.2012.88.

LY, LinhThao, INDIONO, Conrad, MANGLER, Jürgen y RINDERLE-MA, Stefanie, 2012. Data Transformation and Semantic Log Purging for Process Mining. En: *24th International Conference on Advanced Information Systems Engineering (CAiSE'12)* [en línea]. S.l.: Springer. junio 2012. Disponible desde: http://dbis.eprints.uni-ulm.de/796/.

MA, L., 2012, *How to Evaluate the Performance of Process Discovery Algorithms* [online]. Master Thesis. Netherlands: Eindhoven University of Technology. [Accessed 3 October 2012]. Available from: http://alexandria.tue.nl/extra1/afstversl/wsk-i/ma2012.pdf

MANS, R. S., SCHONENBERG, M. H., SONG, M., AALST, W. M. P. and BAKKER, P. J. M., 2009, Application of process mining in healthcare–a case study in a dutch hospital. *Biomedical Engineering Systems and Technologies*. 2009. P.425–438.

DE MEDEIROS, A.K.A., WEIJTERS, A.J.M.M. and VAN DER AALST, W.M.P., 2007, Genetic process mining: An experimental evaluation. *Data Mining and Knowledge Discovery*. 2007. Vol.14, no.2, p.245–304.

MEDEIROS, A K Alves de, DONGEN, B F van, AALST, W M P van der y WEIJTERS, A J M M, 2004. Process Mining: Extending the Alpha algorithm to Mine Short Loops. En: *BETA Working Paper Series*. 2004.

ModeloObjetual - Bota del día, 2013. [online], [Accessed 7 February 2014]. Available from: www.modeloonjetual-botadeldia.es

MUNOZ-GAMA, J., ADRIANSYAH, A., CARMONA, J. and DONGEN, B. F. van, 2011, Alignment Based Precision Checking in Process Mining. 2011. 0002

MICHIE, D., SPIEGELHALTER, D.J, TAYLOR, C.C and CAMPBELL, J, 1994, Machine learning, neural and statistical classification. 1994.

PÉREZ-CASTILLO, R., DE GUZMÁN, I.G.-R., PIATTINI, M., WEBER, B. and PLACES, Á.S., 2011, An empirical comparison of static and dynamic business process mining. In:*Proceedings of the ACM Symposium on Applied Computing* [online]. TaiChung. 2011. p.272–279. 26th Annual ACM Symposium on Applied Computing, SAC 2011. ISBN 9781450301138 (ISBN). Available from: http://hinari-gw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-79959314832&partnerID=40&md5=4273c892fb28632d16e1a99bb80321a5 © 2011 ACM.

PÉREZ ALFONSO and YZQUIERDO HERRERA, 2012, Recommendation of Process Discovery Algorithms: a Classification Problem. In: 2012.

R.P.J.M. VAN ARENDONK BSC, 2011, *A Benchmark Set for Process Discovery Algorithms*. Eindhoven University of Technology.

ROZINAT, A. and VAN DER AALST, W.M.P., 2008, Conformance checking of processes based on monitoring real behavior. *Information Systems*. 2008. Vol.33, no.1, p.64–95.© 2007 Elsevier B.V. All rights reserved.

ROZINAT, A., VELOSO, M. and VAN DER AALST, W. M. P., 2008, Using hidden markov models to evaluate the quality of discovered process models. *Extended Version. BPM Center Report BPM-08-10, BPMcenter.org* [online]. 2008. [Accessed 6 December 2013]. Available from: http://www.researchgate.net/publication/228670148_Using_hidden_markov_models_to_evaluate_the_quality_of_discovered_process_models/file/9c960517a55f6f3378.pdf

ROZINAT, A, MEDEIROS, A K Alves de, GÜNTHER, C W, WEIJTERS, A J M M y WANG, J., HE, T., WEN, L., WU, N., TER HOFSTEDE, A. y SU, J., 2010. A behavioral similarity measure between labeled Petri nets based on principal transition sequences. En: *On the Move to Meaningful Internet Systems: OTM 2010* [en línea]. 2010. pp.394–401.[Accedido 17 septiembre 2012]. Disponible desde: http://www.springerlink.com/index/8017076272360022.pdf.

SEPPE K.L.M. VANDEN BROUCKE and CÉDRIC DELVAUX, 2013, Uncovering the Relationship between Event Log Characteristics and Process Discovery Techniques. 2013.

VAN DER AALST, W., ADRIANSYAH, A., DE MEDEIROS, A.K.A., ARCIERI, F., BAIER, T., BLICKLE, T., BOSE, J.C., VAN DEN BRAND, P., BRANDTJEN, R., BUIJS, J., BURATTIN, A., CARMONA, , 2012, *Process mining manifesto* [online]. Clermont-Ferrand. 9th International Conference on Business Process Management, BPM 2011P. ISBN 18651348 (ISSN); 9783642281075 (ISBN). Available from: http://hinari-gw.who.int/whalecomwww.scopus.com/whalecom0/inward/record.url?eid=2-s2.0-84863011087&partnerID=40&md5=07d0f3bad2de1ffec274ccdb8c971b00 © 2012 Springer-Verlag.

VAN DER AALST, W.M.P. and DUSTDAR, S., 2012, Process mining put into context. *IEEE Internet Computing*. 2012. Vol.16, no.1, p.82–86. © 2006 IEEE.

VAN DER AALST, Wil, 2012, Process Mining: Overview and Opportunities. *ACM Trans. Manage. Inf. Syst.* July 2012. Vol.3, no.2, p.7:1–7:17. DOI 10.1145/2229156.2229157.

VAN DONGEN, Boudewijn F., DE MEDEIROS, Ana Karla A., VERBEEK, H. M. W., WEIJTERS, AJMM and VAN DER AALST, Wil MP, 2005, The ProM marco de trabajo: A new era in process mining tool support. In: *Applications and Theory of Petri Nets 2005* [online]. Springer. p.444–454. [Accessed 7 December 2013]. Available from: http://link.springer.com/chapter/10.1007/11494744_25

VERBEEK, H. M. W., BUIJS, JCAM, VAN DONGEN, B. F. and VAN DER AALST, Wil MP, 2010, Prom 6: The process mining toolkit. *Proc. of BPM DemonstrationTrack*. 2010. Vol.615, p.34–39.

VERBEEK, H. M. W. and VAN DER AALST, W. M. P., 2012, An Experimental Evaluation of Passage-Based Process Discovery. *BPM Center Report BPM-12-14, BPMcenter. org* [online]. 2012. [Accessed 19 September 2012]. Available from: http://www.win.tue.nl/~hverbeek/downloads/preprints/Verbeek12.pdf

WANG, Xiaodong, ZHANG, Li y CAI, Hongming, 2012. Using Suffix-Tree to Identify Patterns and Cluster Traces from Event Log. En: *Signal Processing and Information Technology* [en línea]. S.l.: Springer Berlin Heidelberg. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. pp.126–131. [Accedido 4 septiembre 2012]. ISBN 978-3-642-32573-1. Disponible desde: http://www.springerlink.com/content/qh73ppu501426n07/abstract/.

WESTERGAARD, Michael and KRISTENSEN, Lars Michael, 2009, The Access/CPN marco de trabajo: A Tool for Interacting with the CPN Tools Simulator. In: *Applications and Theory of Petri Nets* [online]. Springer Berlin Heidelberg. p.313–322. Lecture Notes in Computer Science, 5606. [Accessed 7 December 2013]. ISBN 978-3-642-02423-8, 978-3-642-02424-5. Available from: http://link.springer.com/chapter/10.1007/978-3-642-02424-5_19

WEIJTERS, A J M M and AALST, W M P Van der, 2003, Rediscovering Workflow Models from Event-Based Data using Little Thumb. *Integrated Computer-Aided Engineering*. 2003. Vol.10, no.2, p.151–162.

WEIJTERS, A., VAN DER AALST, W. M. P. and DE MEDEIROS, A. K. A., 2006, Process mining with the heuristics miner-algorithm. *Technische Universiteit Eindhoven, Tech. Rep. WP* [online]. 2006. Vol.166. [Accessed 1 December 2012]. Available from: http://cms.ieis.tue.nl/Beta/Files/WorkingPapers/Beta_wp166.pdf

WIL M.P. VAN DER AALST, 2010, *Process_Mining__Discovery__Conformance_and_Enhancement_of_Business_Processes* [online]. Springer-Verlag Berlin Heidelberg 2011. Springer Heidelberg Dordrecht London New York: Springer Berlin / Heidelberg. ISBN 978-3-642-19344-6, e-ISBN 978-3-642-19345-3. ISBN 978-3-642-19344-6. Available from: www.springer.com H.4.1, H.2.8, I.2.6, F.3.2, D.2.2, J.1