novática
Revista de la Asociación de Técnicos de Informática

Novática, revista fundada en 1975 y decana de la prensa informática española, es el órgano oficial de expresión y formación continua de **ATI** (Asociación de Técnicos de Informática), organización que edita también la revista **REICIS** (Revista Española de Innovación, Calidad e Ingeniería del Software).

<http://www.ati.es/novatica/>
<http://www.ati.es/reicis/>

ATI es miembro fundador de **CEPIS** (*Council of European Professional Informatics Societies*) y es representante de España en **IFIP** (*International Federation for Information Processing*); tiene un acuerdo de colaboración con **ACM** (*Association for Computing Machinery*), así como acuerdos de vinculación o colaboración con **AdaSpain**, **AI2**, **ASTIC**, **RITSI** e **Hispalinux**, junto a la que participa en **ProInnova**.

Coordinación Editorial, Redacción Central y Redacción ATI Madrid
Plaza de España 6, 2ª planta, 28008 Madrid
Tlfn.914029391; fax.913093685 <novatica@ati.es>
Composición, Edición y Redacción ATI Valencia
Av. del Reino de Valencia 23, 46005 Valencia
Tlfn. 963740173 <novatica_prod@ati.es>
Administración y Redacción ATI Cataluña
Via Laietana 46, ppal. 1ª, 08003 Barcelona
Tlfn.934125235; fax 934127713 <secregen@ati.es>
Redacción ATI Andalucía <secreand@ati.es>
Redacción ATI Galicia<secregal@ati.es>
Suscripción y Ventas <novatica.subscripciones@atinet.es>
Publicidad Plaza de España 6, 2ª planta, 28008 Madrid
Tlnf 914029391; fax.913093685 <novatica@ati.es>
Imprenta: Derra S.A., Juan de Austria 66, 08005 Barcelona.
Depósito legal: B 15.154-1975 -- ISSN: 0211-2124; CODEN NOVAEC
Portada: "Mineral, vegetal, animal" - Concha Arias Pérez / © ATI
Diseño: Fernando Agresta / © ATI 2003

Nº 223, mayo-junio 2013, año XXXIX

sumario

editorial
El "caso Snowden" y la seguridad de las redes de telecomunicación > 02

en resumen
Soporte al negocio y práctica profesional: El sueño del buen editor > 03
Llorenç Pagés Casas

noticias de IFIP
IFIP TC6 Latin American Tutorials in Networking (LATIN 2013) > 03
Ramon Puigjaner Trepat

monografía
Minería de procesos
Editores invitados: Antonio Valle Salas y Anne Rozinat

Presentación. Una perspectiva sobre la minería de procesos > 05
Antonio Valle Salas, Anne Rozinat

Minería de procesos: La objetivación de la intuición en los procesos de toma de decisiones en los negocios, más transparentes gracias al análisis de los datos > 07
Anne Rozinat, Wil van der Aalst

Minería de procesos: Obtenga una radiografía de sus procesos de negocio > 11
Wil van der Aalst

El viaje del descubrimiento de procesos > 20
Josep Carmona Vargas

Posibilidades de uso de la minería de procesos en ITSM > 24
Antonio Valle Salas

Optimización dirigida por minería de procesos de un proceso de aprobación de préstamos al consumo > 31
Arjel Bautista, Lalit Wangikar, S.M. Kumail Akbar

Mejoramiento de procesos con técnicas de minería de procesos, simulación y optimización: Caso de estudio > 41
Santiago Aguirre Mayorga, Carlos Alberto Parra Rodríguez

Detección de cambios temporales en los procesos de negocio mediante el uso de técnicas de segmentación > 49
Daniela Lorena Luengo Mundaca, Marcos Sepúlveda Fernández

secciones técnicas

Referencias autorizadas > 57

visiones sobre Lenguajes de Programación

Cómo la metáfora de objetos llegó a ser el principal paradigma de programación > 62
Jesús J. García Molina

Elección de lenguajes de programación para la enseñanza universitaria > 67
Baltasar García Perez-Schofield

La importancia de la labor del programador. ¿Qué se espera? ¿Cómo se prepara? Análisis desde los lenguajes de programación > 70
Óscar Belmonte Fernández, Carlos Granell Canut

Para pensar > 79
Rafael Martínez Martínez

Programando caminos y resolviendo necesidades > 81
Alejandro Fuentes Penna

sociedad de la información

Programar es crear

El problema del CUIT (corrección del publicado en el número anterior) (Competencia UTN-FRC 2012, problema D, enunciado) > 82
Julio Javier Castillo, Diego Javier Serrano, Marina Elizabeth Cárdenas

Asuntos Interiores

Coordinación editorial / Programación de Novática / Socios Institucionales > 83

Tema del próximo número:
"Eficiencia energética en centros de proceso de datos"

Detección de cambios temporales en los procesos de negocio mediante el uso de técnicas de segmentación

Daniela Lorena Luengo Mundaca, Marcos Sepúlveda Fernández

Departamento de Ciencias de la Computación, Escuela de Ingeniería, Pontificia Universidad Católica de Chile

<dlluengo@uc.cl>,<marcos@ing.puc.cl>

Resumen: *Hoy en día, las organizaciones tienen la necesidad de estar constantemente cambiando para ajustarse a las necesidades del entorno. Estos cambios se reflejan en sus procesos de negocio. Por ejemplo, un supermercado debido a cambios estacionales tendrá distinta demanda en distintos meses del año, por lo que sus procesos de abastecimiento o de reposición de productos podrían ser distintos en distintas épocas del año. Una forma de analizar con profundidad un proceso y entender cómo realmente se ejecuta en la práctica a través del tiempo, es en base al análisis de sus registros históricos almacenados en los sistemas de información, lo cual es conocido como minería de procesos. Sin embargo, en la actualidad la mayoría de las técnicas que existen para analizar y mejorar procesos consideran todos los registros de un proceso de manera estática, es decir, que el proceso no cambia a través del tiempo, lo cual en la práctica es poco realista dada las naturaleza dinámica de las organizaciones. Nuestro trabajo propone una técnica de segmentación que encuentra las distintas versiones de un proceso a través del tiempo. Esta técnica se basa en una ya existente de segmentación que solo considera características estructurales del proceso (flujo de actividades). Nuestra técnica incorpora de manera adicional la característica temporal de los procesos, de tal manera que los* clusters *que se generen al realizar la segmentación tengan una similitud estructural, pero también una cercanía temporal, de tal manera que representen distintas versiones del proceso. Este documento presenta el detalle de la técnica propuesta y un conjunto de experimentos que reflejan que nuestra propuesta entrega mejores resultados que las técnicas existentes de segmentación.*

Palabras clave: Concept Drift*, dimensión temporal, minería de procesos, segmentación.*

Autores

Daniela Lorena Luengo Mundaca, nacida en Santiago de Chile, es Ingeniera Civil de Industrias mención en Tecnologías de la Información por la Pontificia Universidad Católica de Chile. Posee también el grado académico de Magíster en Ciencias de la Ingeniería de la misma casa de estudios. Adicionalmente tiene un Certificado académico en Actividad Física, Deporte, Salud y Educación. Del año 2009 al 2013 trabajó en el Centro de Estudios de Tecnologías de Información de la Pontificia Universidad Católica de Chile (CETIUC), como analista y consultora del área de excelencia de gestión de procesos, realizando levantamiento de procesos en terreno, propuestas de mejoras e investigaciones varias. Actualmente sus intereses están orientados en aplicar los conocimientos que posee, a través de distintos proyectos de emprendimiento.

Marcos Sepúlveda Fernández es Ingeniero Civil de Industrias mención en Computación de la Pontificia Universidad Católica de Chile. Posee también los grados académicos de Magíster y Doctor en Ciencias de la Ingeniería de dicha casa de estudios. Realizó un postdoctorado en la ETH Zürich, Suiza. Es profesor asociado en el Departamento de Ciencia de la Computación en la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile. Desde el año 2001 es profesor en jornada completa de la Escuela de Ingeniería de la Universidad Católica, en el área de Tecnologías de Información. Sus intereses académicos están ligados a la minería de procesos, la modelación de procesos de negocio, el uso estratégico de las TI en las empresas, y la inteligencia de negocios. Es director del Centro de Estudios de Tecnologías de Información de la Pontificia Universidad Católica de Chile (CETIUC), el cual tiene como objetivo promover las mejores prácticas en el uso de las Tecnologías de Información y generar conciencia en los actores clave de la importancia de las TI para la generación de valor y para aumentar la eficiencia de los procesos de negocio.

1. Introducción

Hoy en día, en un mundo globalizado e hiperconectado, las organizaciones tienen la necesidad de mantenerse en permanente cambio para adaptarse a las necesidades del entorno, lo que implica que sus procesos de negocio también deban estar cambiando constantemente.

Para ilustrar esto, es posible considerar el caso de una tienda de juguetes, donde sus procesos de venta pueden variar de manera radical dependiendo de si son ejecutados en Navidad o en período de vacaciones, debido principalmente a los cambios en los volúmenes de la demanda. En Navidad se podría ejecutar un proceso que priorice la eficiencia y el volumen – *throughput* –, y en vacaciones se podría ejecutar un proceso con foco en la calidad de atención al cliente.

En este ejemplo es fácil identificar los períodos en los que la demanda cambia y, por lo tanto, es posible que el responsable de la gestión del proceso tenga claridad respecto a los cambios que sufre el proceso de venta a través del tiempo. Sin embargo, si una organización tiene un proceso que se realiza en distintas oficinas autónomas, por ejemplo, porque se encuentran en distintas ubicaciones geográficas, la evolución de los cambios en el proceso en cada oficina ya no es tan evidente para el responsable central del proceso; los cambios en cada oficina pueden ser distintos y estarse aplicando en distintos momentos en el tiempo.

Entender los cambios que están ocurriendo en las distintas oficinas, podría ayudar a entender mejor cómo mejorar el diseño global del proceso. Poder entender estos cambios y modelar las distintas versiones del proceso, permiten al responsable de su gestión contar con información más precisa y completa para tomar decisiones coherentes que redunden en una mejor atención o eficiencia.

Para lograr lo anterior, se han desarrollado diversos avances en la disciplina de gestión de procesos de negocio (BPM), disciplina que combina conocimiento sobre tecnologías de información y técnicas de gestión, las cuales son aplicadas a procesos de negocio operativos, con el objetivo de mejorar su eficiencia [1].

Dentro de BPM (*Business Process Management*), la minería de procesos se ha posicionado como una disciplina emergente, proveyendo un conjunto de herramientas que ayudan a analizar y mejorar los procesos de negocio [1], en base al análisis de los registros de eventos que almacenan los sistemas de información durante la ejecución de un proceso. Sin embargo, a pesar de los avances desarrollados en este campo, aún existe un gran desafío, el cual consiste en incorporar el hecho de que los procesos cambian a lo largo del tiempo, concepto que es conocido en la literatura como *Concept Drift* [2].

Dependiendo de la naturaleza del cambio, es posible encontrar diferentes tipos de *Concept Drift*, algunos de ellos son: *Sudden Drift* (cambio repentino y significativo a la definición del proceso), *Gradual Drift* (cambio gradual en la definición del proceso, permitiendo la existencia de dos definiciones de éste

> “Actualmente, uno de los problemas en la minería de procesos es que los algoritmos desarrollados suponen la existencia de información relativa a una única versión de un proceso en el *log* de eventos”

de manera simultánea) o *Incremental Drift* (la evolución del proceso se realiza a través de pequeños cambios sucesivos a la definición del modelo).

Pese a que existen todas estas variantes de *Drift*, en la actualidad las técnicas de minería de procesos existentes están limitadas a encontrar los puntos en el tiempo en los que el proceso cambia, centrándose principalmente en cambios de tipo *Sudden Drift*. El problema de esta limitación es que en la práctica no es tan frecuente que los procesos de negocio muestren un cambio repentino de su definición.

Si aplicamos las técnicas de minería de procesos existentes en procesos que tengan cambios distintos a *Sudden Drift*, podríamos encontrarnos con resultados de poco sentido para el negocio.

En este documento proponemos un nuevo enfoque, permitiendo descubrir las versiones de un proceso cuando tiene distintos tipos de *Drift*, ayudando a entender cómo se comporta el proceso a través del tiempo. Para llevar a cabo esta tarea, se utiliza técnicas existentes de segmentación en minería de procesos, pero incorporando el tiempo como una variable adicional al control de flujo para generar los distintos segmentos o *clusters*. Se utilizan técnicas de *Trace Clustering*, las cuales, a diferencia de otras técnicas basadas en métricas para medir la distancia entre secuencias completas, tienen una complejidad lineal, permitiendo la entrega de resultados en menores tiempos [3].

El enfoque de nuestro trabajo contribuye al análisis del proceso, permitiendo al responsable de la gestión del proceso tener una visión más realista de cómo se comporta el proceso en distintos intervalos de tiempo. Con este enfoque es posible determinar las distintas versiones del proceso, las características de cada una de ellas, e identificar en qué momento ocurren estos cambios.

Este artículo está organizado de la siguiente manera. La **sección 2** presenta el trabajo relacionado. La **sección 3** presenta el método de segmentación base y el modificado. La **sección 4** presenta experimentos y resultados obtenidos, para finalmente, presentar en la **sección 5** las conclusiones y el trabajo futuro.

2. Trabajo relacionado

La minería de procesos es una disciplina que ha concentrado gran interés en la actualidad. Esta disciplina asume que la información histórica almacenada en los sistemas de información sobre un proceso se encuentra en un registro, conocido como *log* de eventos [4]. Este registro contiene información histórica de las actividades que se llevan a cabo en cada ejecución del proceso, donde cada fila del registro está compuesta, al menos, por un identificador (id) asociado a cada ejecución individual del proceso, el nombre de la actividad ejecutada, su marca de tiempo (día y hora en que ocurre la actividad) y, opcionalmente, información adicional, como el ejecutor de la actividad u otros. Adicionalmente, en la literatura [3] se define como traza de la ejecución de un proceso, a la lista ordenada de actividades invocadas por una ejecución en particular.

Actualmente, uno de los problemas en la minería de procesos es que los algoritmos desarrollados suponen la existencia de información relativa a una única versión de un proceso en el *log* de eventos. Sin embargo, esto muchas veces no se cumple, por lo que aplicar los algoritmos de minería de procesos a estos *logs* lleva a resultados poco representativos y/o de gran complejidad, que aportan poco a la tarea de análisis y mejora de procesos.

2.1. Segmentación del log de eventos

Para resolver el problema mencionado en la minería de procesos, se han propuesto técnicas de segmentación del *log* de eventos antes de aplicar técnicas de minería de procesos [5], las cuales consisten en dividir el *log* de eventos en *clusters* homogéneos, para luego aplicar de manera independiente las técnicas de minería de procesos sobre cada uno de ellos y así obtener información o modelos más representativos. La **figura 1** muestra la etapa de procesamiento del *log* y utiliza una técnica de descubrimiento como ejemplo de técnica de minería de procesos.

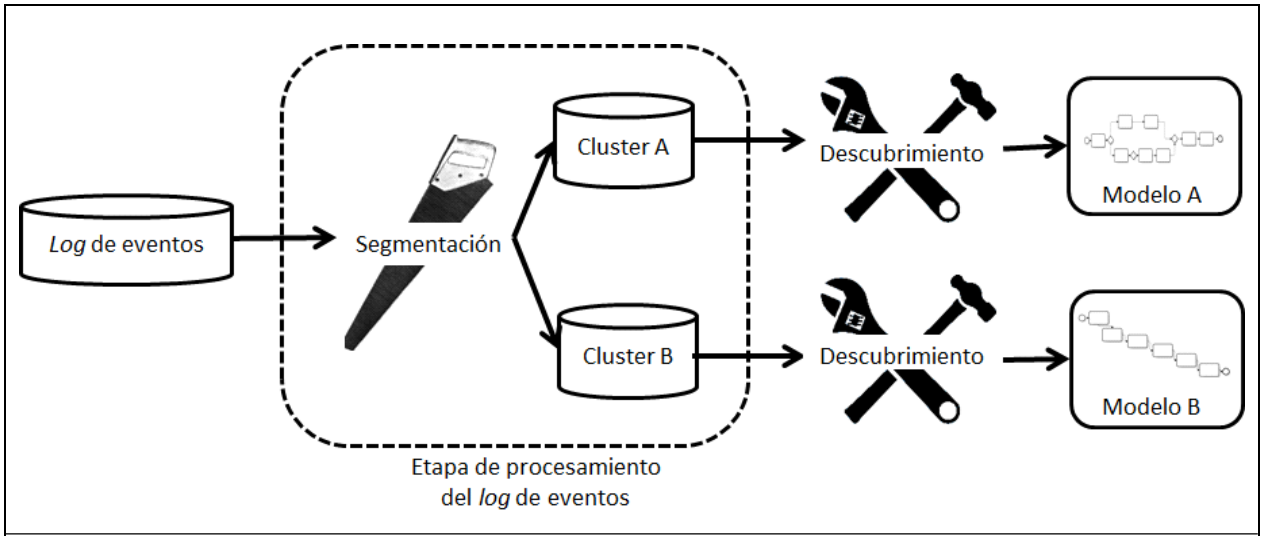

Figura 1. Etapa de procesamiento del *log* de eventos.

> *Concept Drift*, en BPM, se refiere a la situación en la cual un proceso ha sufrido cambios en su diseño dentro del periodo analizado y no se conoce el momento en que se produjeron los cambios

Para realizar esta segmentación es necesario definir una manera de representar las trazas, de manera de poder agruparlas posteriormente de acuerdo a un criterio de similitud previamente definido. Actualmente existen varias técnicas de segmentación en la minería de procesos [3][6], donde la mayoría de ellas considera principalmente información del flujo de las actividades. Estas técnicas pueden ser clasificadas en dos categorías:
1) Técnicas que transforman las trazas en un espacio vectorial, en donde cada traza se convierte en un vector. La segmentación del *log* se puede hacer utilizando una variedad de técnicas de segmentación en el espacio vectorial, como por ejemplo: *Bag of activities, K-gram model* [6] y *Trace clustering* [5]. Sin embargo, estas técnicas tienen el problema que carecen de información de contexto, lo que se ha intentado resolver con la técnica *Trace clustering based on conserved patterns* [3].
2) Técnicas que operan con la traza completa. Estas técnicas utilizan métricas de distancia como *Levenshtein* y *Generic Edit Distance* [6], en conjunto con técnicas estándar de segmentación, asignando un costo a la diferencia entre trazas.

Sin embargo, las técnicas existentes en ambas categorías, a pesar de mejorar la segmentación a través de formar *clusters* de trazas estructuralmente similares, no consideran la dimensión temporal de la ejecución de los procesos, ni cómo el proceso va cambiando en el tiempo.

2.2. El desafío de *Concept Drift*

Concept Drift, en BPM, se refiere a la situación en la cual un proceso ha sufrido cambios en su diseño dentro del periodo analizado y no se conoce el momento en que se produjeron los cambios. Estos cambios se pueden deber a varios factores, pero principalmente se deben a la naturaleza dinámica de los procesos [7].

Los cambios en un proceso en los que se ha centrado el estudio de *Concept Drift*, tienen que ver con los cambios en la perspectiva de control de flujo, y pueden ser de dos tipos: cambios permanentes o cambios momentáneos, según la duración de los cambios.

Cuando ocurren cambios en periodos cortos de tiempo y pocas instancias se ven afectadas, entonces se habla de cambios momentáneos. Estos cambios también son reconocidos en el lenguaje de procesos, como ruido o anomalías.

Por otro lado, los cambios permanentes ocurren en períodos más prolongados de tiempo y/o hay una considerable cantidad de instancias afectadas por los cambios, lo cual hace referencia a un cambio en el diseño del proceso.

Nuestro interés se centra en los cambios permanentes en la perspectiva de control de flujo, los cuales pueden dividirse en las siguientes 4 categorías:
- *Sudden Drift*: Se refiere a los cambios que ocurren de manera drástica, es decir, la forma de realizar el proceso cambia repentinamente de un momento a otro.
- *Recurring Drift*: Cuando los cambios que sufre el proceso ocurren de manera periódica, es decir, una forma de hacer el proceso se repite en otro periodo de tiempo posterior.
- *Gradual Drift*: Se refiere a cambios que no son drásticos, sino que en algún momento dos versiones del proceso se traslapan, ya que corresponde a una transición.
- *Incremental Drift:* Es cuando un proceso tiene pequeños cambios incrementales. Este tipo de cambios es más frecuente en organizaciones que adoptan metodologías ágiles de BPM.

Para resolver el problema de *Concept Drift*, han surgido nuevos enfoques que analizan el dinamismo de los procesos.

Bose [2] propone métodos para manejar el *Concept Drift*, mostrando que los cambios en el proceso están indirectamente reflejados en el *log* de eventos, y la detección de estos cambios es factible examinando la relación entre las actividades. Para ello se definen distintas métricas. A partir de estas métricas se propone un método estadístico, cuya idea base es considerar una serie sucesiva de valores e investigar si hay una diferencia significativa entre dos series. Si es que existe, ésta correspondería a un cambio en el proceso.

Stocker [8] también propone un método para manejar *Concept Drift*, el cual considera las distancias entre pares de actividades de distintos trazos como una característica estructural para generar *clusters* cronológicamente subsecuentes.

Ambos enfoques se limitan a determinar el momento en el tiempo en el que el proceso cambia, por lo que se centran en procesos con cambios repentinos, dejando fuera otro tipo de cambios.

Para resolver esto, en un artículo anterior [9] propusimos un enfoque que utiliza técnicas de segmentación para descubrir los cambios que puede sufrir un proceso a través del tiempo, pero sin limitarse a un tipo de cambio en particular. En este enfoque, la similitud entre dos trazas está definida por información del flujo de las actividades y por información del momento en el que se comienza a ejecutar cada traza.

En este trabajo, presentamos una extensión del artículo anterior [9], al incorporar una nueva forma de medir la distancia entre dos trazas.

3. Extendiendo la técnica de segmentación para incorporar la variable temporal

Como ya se mencionó, los enfoques existentes para tratar *Concept Drift* no son suficientemente efectivos para encontrar las versiones de un proceso cuando este tiene cambios de distintos tipos. Para resolver esta problemática recurrimos a la técnica *Trace Clustering* basado en conservación de patrones [3], que permite realizar segmentación del *log* de eventos considerando solo las secuencias de actividades de cada traza.

Nuestro trabajo se basa en esta técnica, y la extiende incorporando la variable temporal de manera adicional a las que ya utiliza para realizar la segmentación.

3.1. *Trace clustering* basado en conservación de patrones

La idea básica que planteamos en este artículo [3] es considerar subsecuencias de actividades que se repiten en múltiples trazos como conjuntos de características para realizar la segmentación. Cuando dos instancias tienen en común un significativo número de subsecuencias, entonces se asume que tienen una similitud estructural y estas instancias son asignadas al mismo *cluster*.

Se definen seis tipos de subsecuencias, pero haremos la definición formal solo de una (MR), ya que es la utilizada para desarrollar nuestro enfoque; el trabajo podría ampliarse y utilizar las otras subsecuencias.

- *Maximal Repeat* (MR): Un *Maximal Repeat* en una secuencia T, es definido como un par de subsecuencias idénticas, tal que los elementos inmediatamente a la derecha e inmediatamente a la izquierda sean distintos en T. Intuitivamente, una MR corresponde a una subsecuencia de actividades que se repite más de una vez en el *log*.

En la **tabla 1** se observa un ejemplo donde

> “La idea básica que planteamos en este artículo es considerar subsecuencias de actividades que se repiten en múltiples trazos como conjuntos de características para realizar la segmentación”

Secuencia	*Maximal Repeat*	*Conjunto de Características*
bbbcd-bbbc-caa	{a, b, c, bb, bbbc}	{bb, bbbc}

Tabla 1. Ejemplo de *Maximal Repeat* y Conjunto de Características.

Traza\Conjunto de Características	**bb**	**bbbc**
bbbcd	2	1
bbbc	2	1
caa	0	0

Tabla 2. Matriz de características estructurales.

se determina las MR existentes en una secuencia. Lo que hace esta técnica es construir una única secuencia a partir del *log* de eventos, la cual es obtenida concatenando todas las trazas, pero incorporando un delimitador entre ellas. Luego, sobre esta única secuencia se aplica la definición de MR. El conjunto de todos las MR descubiertas en esta secuencia, con más de una actividad, es llamado Conjunto de Características.

A partir del Conjunto de Características, se crea una matriz que nos permite calcular la distancia entre las distintas trazas. Cada fila de la matriz corresponde a una traza y cada columna a una característica del Conjunto de Características. Los valores de la matriz corresponden al número de veces que se encuentra cada característica en las distintas trazas (ver **tabla 2)**. Esta matriz la llamaremos, matriz de características estructurales.

Este enfoque de segmentación basado en patrones, utiliza como técnica de segmentación el "*Agglomerative Hierarchical Clustering*" con criterio de mínima varianza, utilizando la distancia euclidiana para medir la distancia entre trazas, la cual se define de la siguiente manera.

$$dist\,(A,B) = \sqrt{\sum_{i=1}^{n}(T_{Ai} - T_{Bi})^2}$$

Donde:

$dist\,(A,B)$ = distancia entre la traza A y la traza B
n = número de características del Conjunto de Características
T_{Ai} = número de veces que está la característica i en la traza A

3.2. Extendiendo *Trace clustering* para incorporar la variable temporal

Para identificar los distintos tipos de cambios que pueden ocurrir en los procesos de negocio, debemos buscar la manera de identificar las versiones de un proceso.

Si solo miramos las características estructurales dejamos fuera información respecto a su temporalidad. Ambas características, estructurales y temporales, son muy importantes, ya que la estructura nos indica cuán similar es una instancia a otra y la temporalidad nos indica qué tan cercanas en el tiempo están estas dos instancias. Nuestro enfoque busca identificar las distintas formas de ejecutar el proceso utilizando ambas características (estructurales y temporales) al mismo tiempo, tal como se ilustra en la **figura 2**.

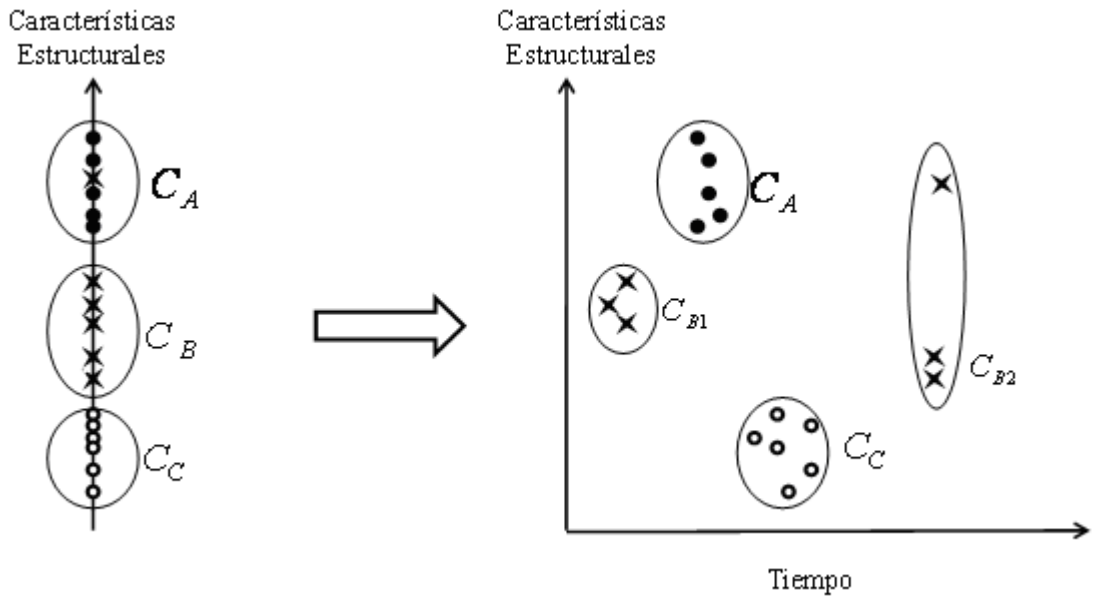

Figura 2. Ejemplo de la relevancia de considerar el tiempo en el análisis.

Traza\Conjunto de Características	bb	bbbc	Tiempo
bbbcd	2	1	tiempo 1
bbbc	2	1	tiempo2
caa	0	0	tiempo3

Tabla 3. Matriz de características estructurales más la dimensión temporal.

Consideramos que la variable temporal relevante de analizar es el tiempo en que se comienza a ejecutar cada instancia del proceso. Para cada traza, almacenamos en la dimensión tiempo, el número de días que han transcurrido desde una marca de tiempo de referencia, por ejemplo, días transcurridos desde el 1 de enero de 1970 hasta la marca de tiempo en la cual se comienza a ejecutar la primera actividad de la traza (ver **tabla 3**).

Este nuevo enfoque también utiliza "*Agglomerative Hierarchical Clustering*" con criterio de mínima varianza como técnica de segmentación.

Para calcular la distancia entre dos trazas utilizamos la distancia euclidiana, pero modificada, de tal manera que considere al mismo tiempo las características estructurales y la característica temporal.

Primero definimos T_{Ji}, como la característica i de la traza J. Si la característica i no se encuentra en la traza J, su valor será 0, de lo contrario, su valor será el número de veces que la característica i se encuentra en la traza J.

$T_{J(n+1)}$ corresponde a la característica temporal de la traza J y su valor es el número de días (podrían ser horas, minutos o segundos, dependiendo del proceso) que han transcurrido desde una marca de tiempo de referencia. Se le da el índice (n+1) para indicar que se agrega a las n características estructurales.

Definimos L como el conjunto de todas las trazas del *log*, luego la expresión $\max_{J \in L}(T_{Ji})$ a la mayor cantidad de veces que está la característica i en alguna traza del *log* de eventos. De la misma forma, $\min_{J \in L}(T_{Ji})$ corresponde a la menor cantidad de veces que está la característica i en alguna traza.

$\min_{J \in L}(T_{J(n+1)})$ y $\max_{J \in L}(T_{J(n+1)})$ corresponden al mayor y menor, respectivamente, instante de tiempo en que se comenzó a ejecutar una traza del *log* de eventos.

También, definimos $D_E(A,B)$ y $D_T(A,B)$, como la distancia estructural y la distancia temporal entre la traza A y la traza B respectivamente.

$$D_E(A,B) = \sqrt{\sum_{i=1}^{n}\left(\frac{T_{Ai} - T_{Bi}}{\max_{J \in L}(T_{Ji}) - \min_{J \in L}(T_{Ji})}\right)^2}$$

$$D_T(A,B) = \sqrt{\left(\frac{T_{A(n+1)} - T_{B(n+1)}}{\max_{J \in L}(T_{J(n+1)}) - \min_{J \in L}(T_{J(n+1)})}\right)^2}$$

Donde:

n = número de características del Conjunto de Características Estructurales

Ambas distancias, D_E y D_T están normalizadas, sin embargo el dominio de D_E es mayor al de D_T. Es por ello que también definimos: Min_E, Max_E, Min_T and Max_T:

$$Min_E = \min_{A,B \in L} \sqrt{D_E(A,B)} \quad , \quad A \neq B$$

$$Max_E = \max_{A,B \in L} \sqrt{D_E(A,B)} \quad , \quad A \neq B$$

$$Min_T = \min_{A,B \in L} \sqrt{D_T(A,B)} \quad , \quad A \neq B$$

$$Max_T = \max_{A,B \in L} \sqrt{D_T(A,B)} \quad , \quad A \neq B$$

Min_E y Max_E corresponden a la distancia mínima y máxima (normalizada) entre todas las trazas, solo considerando las características estructurales.

Min_T y Max_T corresponden a la distancia mínima y máxima (normalizada) entre todas las trazas, solo considerando las características temporales.

La nueva forma para medir la distancia entre dos trazas, *dist (AB)*, incorpora el parámetro μ, al que llamaremos ponderador de la dimensión temporal, que sirve para ponderar las características estructurales y temporales. Adicionalmente, esta nueva forma para medir la distancia ajusta D_E y D_T, de tal manera que el peso de ambas distancia, D_E y D_T, sea equivalente.

$$dist(A,B) = (1-\mu)\frac{D_E(A,B) - Min_E}{Max_E - Min_E} + \mu\,\frac{D_T(A,B) - Min_T}{Max_T - Min_T}$$

El ponderador de la dimensión temporal, μ, puede tener valores entre 0 y 1, según la relevancia que se le dé a la característica temporal.

4. Evaluación

Analizamos la técnica propuesta usando seis *log* de eventos obtenidos de distintos procesos sintéticos, los cuales se construyeron con CPN Tools [10][11].

Para medir el desempeño de la técnica utilizamos el *plug-in Guide Tree Miner* [3] disponible en ProM 6.1[1] y también una versión modificada de este *plug-in* que incorpora los cambios propuestos.

La evaluación se llevó a cabo usando distintas métricas para medir la efectividad de clasificación del nuevo enfoque versus el enfoque base.

4.1. Experimentos y resultados

En la **figura 3** se muestra la secuencia de pasos realizada en los experimentos.

1) Se usó simulación a partir de modelos diseñados, en este caso M1 y M2, para crear el *log* sintético. Para esto se utilizó CPN Tools.

2) A partir del *log* sintético se utiliza alguna técnica de segmentación para generar *clusters*, en este caso dos (C_A y C_B). En los experimentos se aplican dos técnicas de segmentación:

- Enfoque base: *Trace clustering* basada en conservación de patrones.
- Nuestro enfoque: *Trace clustering* extendida, incorporando la variable temporal, utilizando el ponderador de la dimensión temporal, μ, con distintos valores entre 0 y 1.

3) Para cada uno de estos *clusters* se realiza descubrimiento de proceso, generando dos nuevos modelos (M_A y M_B).

El desempeño del enfoque puede ser medido en dos momentos:

a) Conformidad 1: Las métricas se miden entre un modelo original y un *cluster* generado.

b) Conformidad 2: Las métricas se miden entre un modelo original y un modelo generado.

Las métricas que se utilizan para medir Conformidad 1 son las siguientes:

- *Accuracy*: Número de instancias correctamente clasificadas en cada cluster, de acuerdo a lo que se conoce del *log* de eventos original. Sus valores están entre 0% y 100%, donde 100% corresponde a cuando la segmentación se hizo de manera exacta.
- *Fitness*: Indica cuánto del comportamiento observado en un *log* de eventos (por ejemplo, cluster C_A) es capturado por el modelo original del proceso (por ejemplo, modelo M_2) [12].

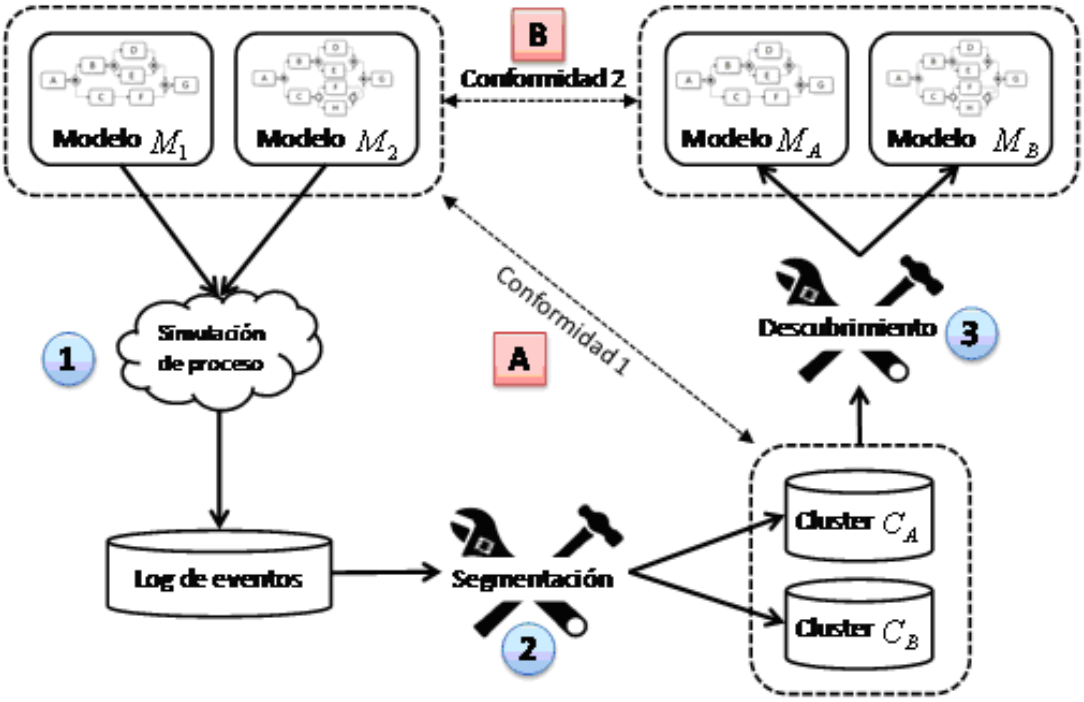

Figura 3. Pasos para realizar las pruebas experimentales.

Sus valores están entre 0 y 1, donde 1 corresponde a que el modelo es capaz de representar todas las trazas del *log*.

■ *Precisión*: Mide la generalidad del modelo, donde se prefiere modelos con un mínimo de comportamiento para representar lo mejor posible el registro del *log* de eventos. Sus valores están entre 0 y 1, donde 1 significa que el modelo no tiene comportamiento adicional a lo que indican los trazos [13].

Las métricas que se utilizan para medir Conformidad 2 son las siguientes [14]:

■ *Behavioral Precision (X)* (B_P)
■ *Structural Precision (X)* (S_P)
■ *Behavioral Recall (X)* (B_R)
■ *Structural Recall (X)* (S_R)

Estas métricas cuantifican precisión y generalidad de un modelo con respecto a otro. Sus valores están entre 0 y 1, donde 1 es el mejor valor esperado.

La **tabla 4** resume los resultados de aplicar el enfoque base y el nuevo enfoque (variando el valor del parámetro μ), a los distintos *logs* de eventos sintéticos creados. En esta tabla se muestra la métrica de a*ccuracy*, la cual nos indica el porcentaje de trazas correctamente clasificadas.

Para cada *log* utilizado, el porcentaje de *accuracy* más alto se alcanza con nuestro enfoque, pero con distintos valores de μ (varía entre 0,2 y 0,9). La razón de esto se debe a que en cada *log* no existe la misma relevancia entre la distribución temporal de las trazas versus la estructura del proceso.

Utilizamos el Log (f) para hacer un análisis más profundo de los resultados, midiendo para este *Log*, todas las métricas definidas

Enfoque	*μ*	*Log (a)*	*Log (b)*	*Log (c)*	*Log (d)*	*Log (e)*	*Log (f)*
Base	-	57%	38%	55%	86%	99%	53%
Nuevo	0,0	59%	52%	49%	82%	59%	51%
	0,1	65%	52%	49%	82%	59%	68%
	0,2	87%	52%	63%	**100%**	59%	64%
	0,3	87%	52%	63%	**100%**	59%	62%
	0,4	95%	52%	63%	**100%**	59%	63%
	0,5	**100%**	52%	63%	**100%**	59%	**95%**
	0,6	**100%**	52%	73%	**100%**	**100%**	94%
	0,7	96%	**89%**	73%	**100%**	**100%**	67%
	0,8	78%	88%	73%	74%	84%	55%
	0,9	79%	77%	**77%**	68%	77%	78%
	1,0	81%	78%	68%	73%	72%	55%

Tabla 4. Métrica de *accuracy* calculada para los 6 *logs* sintéticos de prueba.

Enfoque	*μ*	*Accuracy*	*Fitness*	*Precisión*	*Promedio* B_P *y* S_P	*Promedio* B_R *y* S_R	*Promedio*
Base	-	53%	0,93	0,78	0,77	0,81	0,76
Nuevo	0,0	51%	0,93	0,73	0,73	0,70	0,72
	0,1	68%	0,93	0,81	0,86	0,86	0,83
	0,2	64%	0,92	0,80	0,83	0,82	0,80
	0,3	62%	0,92	0,80	0,80	0,78	0,78
	0,4	63%	0,92	0,79	0,83	0,86	0,81
	0,5	**95%**	**0,95**	0,85	**0,97**	**0,96**	**0,94**
	0,6	94%	**0,95**	**0,86**	0,95	0,94	0,93
	0,7	67%	0,92	0,84	0,84	0,89	0,83
	0,8	55%	0,94	0,87	0,88	0,94	0,84
	0,9	78%	0,94	0,81	0,89	0,95	0,87
	1,0	55%	0,93	0,87	0,88	0,95	0,84

Tabla 5. Detalle de métricas al analizar el Log (f).

tanto en Conformidad 1 como en Conformidad 2 (ver **tabla 5**).

Todas las métricas calculadas para el Log (f) tienen buen desempeño cuando el parámetro μ vale 0,5 o 0,6. Al ser promediadas las cinco métricas, el promedio más alto se alcanza con μ igual a 0,5.

5. Conclusiones y trabajo futuro

En este documento se presentan las limitaciones de las actuales técnicas de segmentación en minería de procesos, las cuales se centran en agrupar ejecuciones similares (estructural-mente) de un proceso, de tal manera que se formen grupos homogéneos de ejecuciones, para que el análisis sobre cada grupo sea de mayor simplicidad que si se analiza el conjunto de datos completo.

Al centrarse solo en la estructura de las ejecuciones, dejan de lado el comportamiento del proceso a través del tiempo. Para ello surgen nuevas técnicas, pero que también presentan limitaciones, ya que se centran en encontrar los puntos en que cambia el proceso, limitándose a un tipo de cambio, el *Sudden Drift*. Dada esta situación, presentamos un enfoque que utiliza la lógica que usan las técnicas de segmentación, de manera que se encuentren las distintas versiones de un proceso cuando éste presente distintos tipos de cambios, permitiendo entender las variaciones que ocurren en el proceso y cómo realmente se está ejecutando en la práctica a través del tiempo.

Nuestro trabajo se centra en la identificación de las versiones del proceso. La técnica que proponemos, es una herramienta que ayuda a las personas involucradas en el negocio a tomar decisiones. Por ejemplo, se puede determinar si los cambios que se producen en la ejecución del proceso, son realmente los esperados, y en base a esto, tomar mediadas si se descubren comportamientos anormales. También sirve para eventualmente identificar buenas y malas prácticas, las cuales son de alta utilidad al momento de querer mejorar o estandarizar los procesos.

En este documento presentamos un conjunto de métricas para medir el desempeño del enfoque. El desempeño se considera bueno, cuando el enfoque es capaz de segmentar los datos de la misma forma en la que fueron creados. Por lo tanto, estas métricas requieren información a priori del proceso, lo cual no es aplicable en casos reales.

Cada métrica aquí presentada mide distintos aspectos que son difíciles de analizar por sí solos. Sin embargo, al utilizarlos en conjunto, permiten tener una visión de distintas perspectivas del problema, haciendo más completo el análisis.

Un aspecto clave de nuestro enfoque de segmentación, es el valor que se le da al ponderador de la dimensión temporal, μ, el cual está estrechamente relacionado con la naturaleza del proceso. Valores altos de μ le dan mayor importancia al tiempo para realizar la segmentación, mientras que valores bajos de μ, le dan más importancia a las características estructurales del proceso.

Los resultados de los experimentos muestran que el enfoque propuesto en este documento tiene un mejor desempeño en la segmentación del *log* y que existe al menos un valor del parámetro μ que permite entregar mejores resultados en comparación a utilizar solo la técnica de segmentación estructural. Esto se logra, ya que nuestro enfoque es capaz de agrupar las trazas del *log* de tal manera que se identifiquen similitud estructural y cercanía temporal al mismo tiempo.

Una de las métricas utilizadas es el *accuracy*. En algunos experimentos, esta métrica, alcanza el 100%, es decir, que todas las trazas son clasificadas correctamente.

Cuando no se alcanza el 100% de *accuracy*, se debe a que hay procesos que tienen versiones que pese a ser distintas, son similares entre sí, pudiendo incluso tener trazas ejecutables en las dos versiones del proceso, lo cual hace que la clasificación no sea exactamente igual a lo que se esperaba.

Nuestro trabajo futuro en esta línea de investigación es probar este nuevo enfoque con procesos reales. También queremos trabajar en desarrollar los algoritmos existentes para que sean capaces de determinar automáticamente el número óptimo de *clusters*; para ello es necesario definir nuevas métricas que nos permitan calcular el número óptimo de *clusters* sin saber a priori información de las versiones del proceso.

Referencias

[1] W. van der Aalst. *Process Mining, Discovery, Conformance and Enhancement of Business Processes*. Springer, 2011. ISBN 978-3-642-19345-3.

[2] R.P. Bose, W. van der Aalst, I. Zliobaite, M. Pechenizkiy. Handling Concept Drift in Process Mining. *23rd International Conference on Advanced Information Systems Engineering*. Londres, 2011.

[3] R.P. Bose, W. van der Aalst. Trace Clustering Based on Conserved Pastterns : Towards Achieving Better Process Models. *Business Process Management Workshops*, pp. 170-181, 2010.

[4] W. van der Aalst, B. van Dongen, J. Herbst, L. Maruster, G. Schimm, A. Weijters. *Data & Knowledge Engineering*, pp. 237-267, 2003.

[5] M. Song, C. Günther, W. van der Aalst. Trace Clustering in Process Mining. *4th Workshop on Business Process Intelligence* (BPI 08), pp. 109-120. Milano, 2009.

[6] R. Bose, W. van der Aalst. Context Aware Trace Clustering: Towards Improving Process Mining Results. *SIAM*, pp. 401-412, 2009.

[7] W. van der Aalst, A. Adriansyah, A.K. Alves de Medeiros, F. Arcieri, T. Baier, T. Blickle y otros. *Process Mining Manifesto*, 2011.

[8] T. Stocker. Time-based Trace Clustering for Evolution-aware Security Audits. *Proceedings of the BPM Workshop on Workflow Security Audit and Certification*, pp. 471-476. Clermont-Ferrand, 2011.

[9] D. Luengo, M. Sepúlveda. Applying Clustering in Process Mining to find different versions of a business process that changes over time. *Lecture Notes in Business Information Processing*, pp. 153-158, 2011.

[10] A.V. Ratzer, L. Wells, H.M. Lassen, M. Laursen, J. Frank, M.S. Stissing y otros. CPN Tools for editing, simulating, and analysing coloured Petri nets. *Proceedings of the 24th international conference on Applications and theory of Petri nets* pp. 450-462. Eindhoven: Springer-Verlag, 2003.

[11] A.K. Alves De Medeiros, C. Günther. Process Mining:Using CPN Tools to Create Test Logs for Mining Algorithms. *Proceedings of the Sixth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools*, pp. 177–190, 2005.

[12] A. Rozinat, W. van der Aalst. Conformance testing: Measuring the fit and appropriateness of event logs and process models. *Business Process Management Workshops*, pp. 163-176, 2006.

[13] J. Muñoz-Gama, J. Carmona. A fresh look at precision in process conformance. *Proceeding BPM'10 Proceedings of the 8th international conference on Business process management*, pp. 211-226, 2010.

[14] A. Rozinat, A.K. Alves De Medeiros, C. Günther, A. Weijters, W. van der Aalst. *Towards an Evaluation Framework for Process Mining Algorithms*. Genetics, 2007.

Nota

[1] ProM es un *framework* extensible que soporta una variedad de técnicas de minería de procesos en forma de *plug-ins*. Se puede conseguir en <http://www.processmining.org/>.