MultiSearchBP: Entorno para búsqueda y agrupación de modelos de procesos de negocio

Hugo Ordoñez, Juan Carlos Corrales, Carlos Cobos

***Resumen*—El artículo presenta un entorno para búsqueda y agrupación de procesos de negocio denominado MultiSearchBP. Es basado en una arquitectura de tres niveles, que comprende el nivel de presentación, nivel de negocios (análisis estructural, la indización, búsqueda y agrupación) y el nivel de almacenamiento. El proceso de búsqueda se realiza en un repositorio que contiene 146 modelos de procesos de negocio (BP). Los procesos de indización y de consulta son similares a los del modelo de espacio vectorial utilizado en la recuperación de información, y el proceso de agrupación utiliza dos algoritmos de agrupación (Lingo y STC). MultiSearchBP utiliza una representación multimodal de los BP. También se presenta un proceso de evaluación experimental para considerar los juicios de ocho expertos evaluadores a partir de un conjunto de los valores de similitud obtenidos de comparaciones manuales efectuados con anterioridad sobre los modelos de BP almacenados en el repositorio. Las medidas utilizadas fueron la precisión gradual y el *recall* gradual. Los resultados muestran una precisión alta.**

***Palabras Clave*—Procesos de negocio, recuperación de información, búsqueda multimodal, agrupamiento.**

MultiSearchBP: Environment for Search and Clustering of Business Process Models

***Abstract*—This paper presents a Business Process Searching and Grouping Environment called MultiSearchBP. It is based on a three-level architecture comprising Presentation level, Business level (Structural Analysis, Indexing, Query, and Grouping) and Storage level. The search process is performed on a repository that contains 146 Business Process (BP) models. The indexing and query processes are similar to those of the vector space model used in information retrieval and the clustering process uses two clustering algorithms (Lingo and STC). MultiSearchBP uses a multimodal representation of BPs. It also presents an experimental evaluation process to consider the judgments of eight expert evaluators from a set of similarity scores obtained from previous manual comparisons made between the BP models stored in the repository. The measures used were graded precision and graded recall. The results show high accuracy.**

***Keywords*—Business processes, information retrieval, multimodal search, clustering.**

Manuscrito recibido el 18 de marzo de 2013; aceptado para la publicación el 27 de julio del 2013; versión final 16 de junio de 2014.

Hugo Ordoñez está con la Facultad de Ingeniería, Universidad de San Buenaventura, Cali, Colombia, y el Grupo de Ingeniería Telemática de la Universidad del Cauca, Colombia (correo: hugoeraso@gmail.com).

Juan-Carlos Corrales está con el Departamento de Telemática, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, Colombia (correo: jcorral@unicauca.edu.co).

Carlos Cobos está con el Departamento de Sistemas, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, Colombia (correo: ccobos@unicauca.edu.co).

I. Introducción

La apertura de los mercados y la globalización del comercio hacen que las empresas centren su atención en la oferta de nuevos productos y servicios con el propósito de atraer más clientes y de esta forma mantener o mejorar el nivel de ventas y su posicionamiento en el mercado [1]. Para lograr lo anterior, aplican estrategias que satisfacen la demanda y los requerimientos de clientes conocedores y expertos que cada día exigen más [2]. Entre estas demandas se encuentran: agilidad y calidad de servicio, rebaja de costos, disminución de tiempos, calidad de productos, agilidad en las transacciones, entre otras. Esto exige que las empresas se organicen entorno a funciones del negocio tales como: mercadeo, ventas, producción, finanzas y servicio al cliente, donde cada una de ellas se ejecutan de forma independiente según su propio modelo de negocio [3]. La aparición de los Business Process Management Systems (BPMS) permiten agilizar estas funciones dentro de la empresa facilitando su organización en torno a procesos de negocio (BP) [4], [5]. Lo anterior permite coordinar recursos humanos y tecnológicos para llevar a cabo los procesos de la empresa u organización de acuerdo con la estrategia de negocio definida.

Los lineamientos organizacionales definidos por las empresas se modelan por medio de BP, que son formados por procedimientos o actividades que colectivamente alcanzan un objetivo o política de negocio, definiendo roles y relaciones funcionales [6]. La organización por BP permite a las empresas adaptarse más eficientemente a las necesidades de los clientes, ya que los BP pueden ser modificados en cualquier momento y tantas veces como sea necesario [7].

Los BP en las organizaciones son normalmente modelados o creados por expertos, utilizando herramientas para el diseño de BP en donde plasman las operaciones o tareas que se necesita ejecutar en la organización. Las organizaciones que pretenden diseñar o modelar un nuevo BP tienen que empezar revisando grandes cantidades de información acerca de los BP existentes (normalmente almacenados en repositorios de BP).

Dentro de esta información están las instrucciones del trabajo a realizar, quién debe realizarlo y la descripción de las

conexiones con otros sistemas [8]. Esta información es almacenada en archivos que contienen los registros de transacciones conocidos como “logs” o trazas de ejecución [7], [9]. Posteriormente la información revisada sirve como base para el replanteamiento o remodelamiento de un nuevo BP que cumpla con los nuevos requerimientos de la organización [10]. El éxito en la búsqueda (descubrimiento) de los BP sobre los repositorios empresariales permite a los diseñadores reutilizar efectivamente los BP desarrollados previamente y, así disminuir el tiempo de desarrollo de los nuevos BP.

De acuerdo con lo anterior, es necesario contar con un mecanismo de gestión de información eficiente que permita buscar (descubrir) los datos generados por los BP con el propósito de encontrar aquellos BP que más similitud tienen con el comportamiento de las tareas ejecutadas en la organización y que se esperan usar para definir un nuevo BP, para un área del negocio especifica [11], [12].

En esta investigación se propone un entorno que permite el descubrimiento y agrupación de BP por medio de consultas, que contemplan características estructurales y componentes textuales. El entorno se evaluó con base en un repositorio de BP modelados con Business Process Modeling Notation (BPMN), representado en sintaxis XML, mediante el lenguaje Processing Description Language (XPDL). El entorno se basa en el modelo espacio vectorial para la representación de los BP, incorpora características de representación multimodal (que utiliza información estructural y textual) y usa algoritmos de clustering para realizar agrupaciones con base en la similitud de los BP recuperados en la consulta del diseñador.

El resto del documento está organizado de la siguiente manera. La sección 2 presenta trabajos relacionados. La sección 3 describe el entorno propuesto, sus algoritmos y algunas interfaces. La sección 4 muestra los resultados preliminares de la evaluación del modelo. Finalmente, se presentan las conclusiones y el trabajo futuro que el grupo de investigación espera desarrollar en el corto plazo

II. Trabajos relacionados

El tema de interés central en esta investigación es el descubrimiento de BP y la agrupación (clustering) de los mismos. A continuación se presenta un resumen de los trabajos más destacados y al final de cada sección se hace un resumen de las deficiencias de los enfoques propuestos hasta el momento.

A. Descubrimiento de BP basado en lingüística

En [11] los autores plantean un sistema de búsqueda de BP que extiende semánticamente la consulta. Cuenta con un editor de BP basado en redes de Petri e incorpora un repositorio en el cual todos los BP son etiquetados con metadatos. En este trabajo se crea un índice de búsqueda, se eliminan palabras vacías y se ponderan los términos presentes en actividades y estados del BP. El sistema cuenta con dos opciones de búsqueda, una básica y otra extendida. La búsqueda básica consulta sobre todos los modelos presentes en el repositorio o sobre un modelo en especial e incorpora WordNet como elemento de generación de sugerencias semánticas en las búsquedas. Por otra parte, la búsqueda extendida considera a cada actividad del BP como un vector de términos agregando una función de costo parcial, con la cual se calcula una función de costo total. El ordenamiento de los resultados de la consulta se realiza con los valores de la función de costo total más bajas o de menor peso.

En [13] se propone un método de compresión de lingüística basado en redes de Petri , donde se resaltan dos contribuciones realizadas, a saber: 1) un argumento teórico para establecer el grado de compresión de la lingüística, abordando la semiología (estudio de signos) de los gráficos, en donde identifican ocho variables visuales distintas que pueden ser utilizadas para codificar la información de la gráfica del BP y el color es tomado como una de las variables más eficaces para distinguir los elementos de la notación. 2) la formalización de conceptos en el modelado de flujos de trabajo (*workflows*), para lo cual toma el BP como un grafo dirigido bipartito donde P es un conjunto de nodos llamados lugares, T un conjunto de nodos llamados transiciones y Fp $(P \times T) \cup (T \times P)$ es una relación de flujo binario basado en un operador que mapea cada conjunto de nodos T. Para realizar la búsqueda del nuevo modelo ejecuta un algoritmo denominado (max-flow-min-cut) que realiza emparejamiento de nodos para encontrar el flujo máximo de coincidencias de los operadores de conexión.

En [14] se presenta un método de búsqueda basado en descomposición de BP creando un análisis híbrido entre estructura y relevancia. El algoritmo está basado en un análisis iterativo del grafo que representa al BP. La descomposición crea fragmentos de procesos reutilizables (RPF), los cuales cumplen las siguientes características: 1) Un RPF debe ser conectado de manera que todos los nodos puedan llegar desde una entrada de borde o arista, y 2) Cada RPF debe tener sólo una arista de entrada o de salida o ambos en común interconectados con otro fragmento. En este proceso se tiene como meta de búsqueda extraer la frecuencia de ocurrencia más alta en las tareas de los BP representados por los fragmentos generados.

En [15] los autores proponen un método de búsqueda de BP mediante la aplicación de reglas de asociación para información no estructurada. El proceso es llevado a cabo utilizando datos no estructurados en lugar de los registros de las aplicaciones. La ejecución del algoritmo de detección de reglas está dividida en dos: 1) la obtención de la asociación entre los documentos y procesos, 2) construcción de un modelo de lenguaje estadístico para identificación de normas relacionadas con el proceso y las actividades que se presentan en los documentos. La construcción del modelo está dividida en dos actividades principales: el algoritmo analizador, que detecta frases relacionadas con las actividades del proceso por medio de una ontología de dominio, y la identificación de patrones que utiliza una heurística, basada en los elementos de

la ontología de dominio y las sentencias del documento de búsqueda. En la recuperación de los BP se utiliza la detección de patrones, el cálculo de su frecuencia y las asociaciones de las actividades.

B. Descubrimiento de BP basado en agrupamiento (Clustering)

En [16] los autores plantean un algoritmo de clustering secuencial con el propósito de organizar una serie de objetos en un conjunto de grupos, donde cada grupo contiene objetos que son similares por un tipo de medida. Esta medida depende del tipo de objetos o datos presentes en los BP. Cada grupo está asociado con un modelo probabilístico, por lo general una cadena de Markov (al igual que el presentado en [17], [18]). Si para todos los grupos se conocen las cadenas de Markov, entonces cada secuencia de entrada es asignada a la agrupación que mejor pueda producir tal secuencia. El algoritmo desarrolla los pasos siguientes: 1) Inicializa los modelos de cluster (es decir, la cadena de Markov para cada grupo) al azar. 2) Asigna a cada secuencia de entrada el grupo que es capaz de producirlo con la mayor probabilidad. 3) La estimación de cada modelo de clúster de la serie de secuencias que pertenecen a ese grupo. Finalmente, se repiten los pasos 2 y 3 hasta encontrar los modelos de cada cluster o grupo.

En [19] plantean un enfoque de clustering que agrupa secuencias similares e identifica tópicos temáticos presentes en los BP sin la necesidad de proporcionar información de entrada. La agrupación es realizada con el propósito de encontrar información valiosa sobre el tipo de secuencias que se están ejecutando en los BP. El procedimiento de agrupación incluye: Un algoritmo alfa el cual es capaz de volver a crear el BP a través de una red de Petri, con base en las relaciones encontradas en el registro de ejecución de los BP. Métodos de inferencia que consideran el registro de ejecución como una secuencia simple de símbolos, inspirada en el modelo de Markov (al igual que el presentado en [17]) y que genera un modelo gráfico que considera cadenas de Markov de orden creciente con grafos acíclicos dirigidos. Un algoritmo de Clustering jerárquico que tiene en cuenta un amplio conjunto de trazas de ejecución de un mismo proceso, que separa las trazas en grupos y encuentra el gráfico de dependencias por separado para cada grupo. Un algoritmo genético donde las soluciones candidatas son evaluadas por una función de aptitud y cada solución es representada mediante una matriz causal, es decir, un mapa de las entradas y dependencias de salida para cada actividad.

En [18] presentan un esquema de agrupación de BP (tal como en [20], [21]) para recuperación de esquemas gráficos en grupos similares de (sub) procesos y sus relaciones. Se parte de un macro proceso para llegar hasta las actividades más sencillas, para lo cual se toma un conjunto de grafos dirigidos $G_i = <N_i, A_i>$ donde N_i es el conjunto de nodos y $A_i \subseteq N_i \times N_i$ es el conjunto de arcos posiblemente etiquetados, generando un esqueleto de agrupación típica de subestructuras. Los grafos son iterativamente analizados para descubrir en cada paso un grupo de sub-estructuras isomorfas. El clustering se utiliza para comprimir los grafos sustituyendo a cada ocurrencia de la subestructura con un nodo; este proceso se repite hasta que no haya más compresión posible.

C. Diferencias con los trabajos previos

Las propuestas anteriormente descritas en el descubrimiento lingüístico de BP se limitan al emparejamiento de entradas y/o salidas tomando como base la información textual o gráfica y las relaciones semánticas que se encuentra en la notación de estos elementos, además deja de lado el flujo de ejecución o comportamiento. En el proceso de búsqueda los resultados no tienen en cuenta similitud en patrones frecuentes, tipo de actividades, finalidad de la tarea o actividad. Por otro lado en las propuestas de descubrimiento basado en agrupación se eliminan secuencias que solo ocurren una sola vez sin tener en cuenta que pueden ser relevantes para los modelos que forman cada grupo, además la agrupación de atributos internos se mide separando su comportamiento de las propiedades estructurales y los atributos externos son medidos con datos tales como: tiempo de duración, número de errores, costo de ejecución. Esta medición de atributos hace que el costo computacional del algoritmo sea demasiado elevado.

Para alcanzar mayor relevancia de los resultados reportados en los sistemas de descubrimiento de BP, en esta propuesta se plantea un entorno que unifica en un solo espacio de búsqueda, unidades de comportamiento y características textuales de los BP, en lo que se conoce como una representación multimodal. Adicionalmente, integra el uso de algoritmos de clustering para agrupar los resultados de la búsqueda (descubrimiento) con base en la similitud de las características representadas en los modelos de BP descubiertos y lograr así una forma más efectiva de visualización de los resultados.

III. El entorno propuesto

El entorno propuesto, llamado MULTISEARCH*BP*, esta implementado sobre la tecnología Java y es soportado por una arquitectura organizada en 3 capas como se muestra en la La fig. 1. Está compuesta por: 1) un nivel de presentación desde la cual el usuario puede gestionar los BP (adicionar, eliminar, modificar y buscar BP) almacenados en el repositorio y el índice. 2) un nivel de lógica de negocio que se encarga de gestionar los BP, extraer las características estructurales y los componentes textuales de los BP e indexarlos, también responde a las opciones de búsqueda con dos tipos de respuesta: lista lineal ordenada de BP o grupos temáticos de BP que se relacionan con la consulta del usuario (diseñador) y finalmente, 3) un nivel de almacenamiento que se encarga de dar persistencia a los procesos de negocio y al índice de búsqueda. A continuación se explican cada uno de los componentes de esta arquitectura.

Formas para Adicionar / Actualizar / Eliminar: Corresponde a la interfaz grafica de usuario (GUI) usada para adicionar, modificar y eliminar BP del repositorio y del índice.

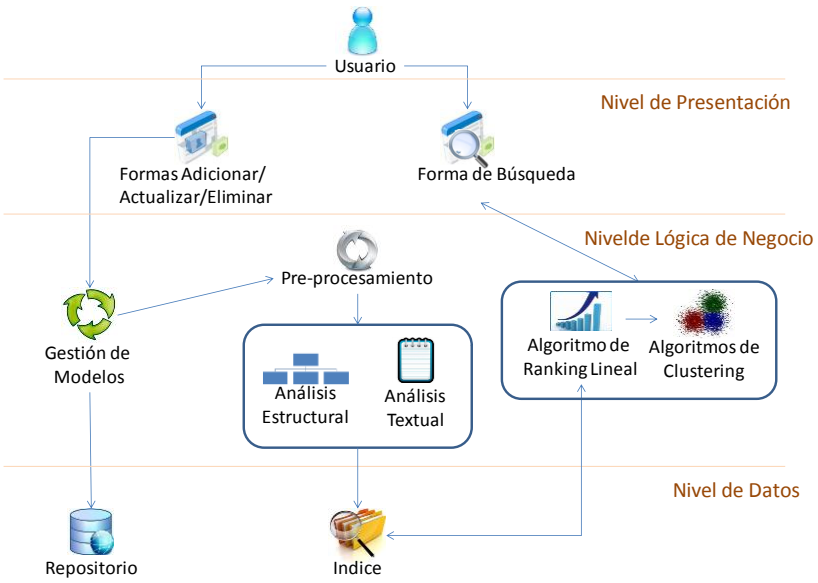

Figura 1. Arquitectura del entorno

Gestion de modelos: Este componente permite hacer gestión sobre los BP, que están en sus formatos originales XML, y representan los modelos de *BP*MN (Business Process Modeling Notation) con sintaxis XPDL (XML Process Definition Language). Estos pueden ser BP de referencia para procesos de dominio específico o BP que ejecutan un conjunto de tareas de una colección empresarial y que pueden ser reconfigurables.

Repositorio: Es la unidad central de almacenamiento y gestión, es similar a una base de datos que comparte información acerca de los artefactos de ingeniería producidos o utilizados por una empresa [10], [22]. Para la evaluación del presente entorno se usó un repositorio con 146 BP. Para cada BP se almacenan las tareas, su*b-p*rocesos y flujos de control.

Cuando la colección de BP se indexa, se realizan tres tareas fundamentales: el pre-procesamiento de cada BP, luego el analisis textual, después el analisis estructural y finalmente la creación del índice completo de la colección. Es preciso tener claro, que el índice se crea para toda la colección, pero también se puede realizar incrementalmente, es decir, uno a uno cada BP.

El **Pre-procesamiento** se encarga de convertir los términos textuales del BP a minúsculas, eliminar caracteres especiales, eliminar palabras vacías, eliminar acentos, y aplicar stemming (algoritmo de porter [27], [28]) para convertir cada uno de los componentes textuales de los BP a su raíz léxica (por ejemplo “fishing” y “fished” en “fish”).

En el **Analisis textual** se lee cada uno de los elementos del conjunto $T:\{BP \mid BP\}$ presentes en el repositorio S, para lo cual cada uno de los elementos de T es representado en forma de árbol (A) tal que ($BP_i = A_i \rightarrow (v, x)$ donde v es un nodo y x representa las aristas). El proceso inicia tomando cada A_i, para extraer las características textuales C_{tj} (nombre de actividad, tipo actividad y descripción) para formar un vector, es decir $\{C_{tj1}, C_{tj2}, \ldots, C_{tjN}\}$, que corresponde a una fila de la matriz MC_{ij} del componente de caracteristicas textuales, donde i representa los BP y j representa las características textuales de cada uno de estos.

El **Analisis estructural** incorpora una estrategia de formación y uso de libros de códigos (codebooks) para generar unidades estructurales básicas secuenciales de los BP. Estos codebooks son construidos con base en las propiedades de similitud en patrones secuenciales frecuentes en la estructura de cada uno de los BP. Generalmente los codebooks han sido empleados en el dominio de recuperación de imágenes utilizados como histogramas de patrones visuales [29] y como vocabularios o diccionarios visuales [23], [24], [25]. Además se utilizan para analizar y buscar ocurrencias de palabras en transcripciones de texto [26].

En este paso se ejecuta el algoritmo (Parser*BP*toCodebook) que analiza la estructura de los modelos de BP almacenados en el repositorio. En este proceso se recorre de manera secuencial la estructura en árbol de los archivos XPDL donde se describe cada BP, para formar una matriz MC de características textuales y una matriz MCd de componentes estructurales (usando codebooks). Este paso se realiza tomando cada $A_i \ni vt$ (vector de transiciones), donde $vt=\{t_{j1}, t_{j2},\ldots, t_{jn}\}$, de lo cual $\forall Cd_i = (vt-1, vt); \quad i \geq 2$ con esto se tiene $A_i = \sum_{i=1}^{n} Cd_i$ formando de esta manera la matriz MCd_{ij} de componentes codebook, donde i representa los BP y j representa los codebook de cada BP. LaFigura 2 hace una representación gráfica de la manera como se forma cada uno de los codebook de un BP. De lo cual es obtenido un vector de codebooks asi: {Start_TaskUser$_1$, TaskUser_ParallelRoute$_2$, ParallelRoute_TaskService$_3$, ParallelRoute_TaskService$_4$}.

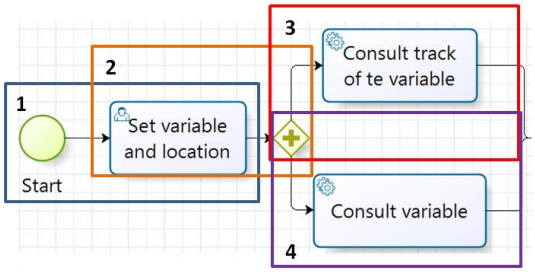

Figura 2. Estructura de cada *codebook*

	1	2	3	4	5	6	7	8	9	10	m
BP_1	1	Cd_1	Cd_2	Cd_3	Cd_4	Cd_5	Ct_1	Ct_2	Ct_3	Ct_4	Ct_m
BP_2	2	w_{ij}					w_{ij}				
BP_3	3		w_{ij}					w_{ij}			
BP_4	4			w_{ij}					w_{ij}		
BP_5	5				w_{ij}					w_{ij}	
BP_n	n					w_{ij}					w_{ij}

Figura 3. Matriz índice (MI)

El **Índice** almacena información de dos tipos: 1) Indexación de las funciones de negocios en la cual se tiene en cuenta la información textual existente en cada BP. 2) indexación estructural la cual está basada en una caracterización entre tipos de tareas, tipos de eventos y tipos de conexiones. Estas dos formas de indexación se unifican (representación multimodal) para tener una representación más exacta del objeto de estudio. El índice almacena eficientemente una estructura conceptual denominada matriz índice (*MI*) de términos por BP (similar al modelo espacio vectorial de recuperación de información [5]), que almacena en cada celda un peso (w_{ij}), el cual refleja la importancia del componente textual en su raíz léxica o codebook contra cada BP. Esta matriz se basa en la ecuación (1) propuesta por Salton [29], [27], donde $F_{i,j}$ es la frecuencia observada del componente textual o del codebook j en el BP_i. Max(F_i) es la mayor frecuencia observada en el BP_i. N es el número de BP en la colección y n_j es el número de BP en los que aparece el componente textual o codebook j. Finalmente la matriz índice $MI = \{MCd_{ij} \cup MC_{ij}\}$ puede ser resumida gráficamente como se muestra en la Figura 3. Esta figura muestra dos zonas o componentes en la *MI*, la primera, muestra el peso de los elementos de cada codebook en cada BP y el segundo el peso de los elementos textuales en cada BP.

$$w_{i,j} = \frac{F_{i,j}}{\max(F_i)} \times \log\left(\frac{N}{n_j + 1}\right) \tag{1}$$

La **Forma de búsqueda** hace referencia a un interfaz gráfica en la cual el usuario puede realizar consultas de tres formas diferentes: 1) por palabras clave (textual), 2) estructural (codebooks), y por 3) combinada de texto y estructura (es decir las dos anteriores en forma conjunta).

La consulta por palabras clave: En estas consultas el usuario puede digitar una o varias por palabras clave representadas en lenguaje natural las cuales forman un vector de consulta $qpc = \{pc_1, pc_2, \ldots, pc_n\}$. El sistema pre-procesa las palabras clave, genera un vector de consulta con los términos registrados en la *MI* y luego compara esta consulta con la parte textual del índice para entregar aquellos BP más similares a la consulta.

La consulta estructural: En esta opción el usuario tiene la posibilidad de elegir uno o varios (codebooks) de una lista de componentes estructurales formados a partir de la colección de BP existentes en el repositorio para formar el vector de consulta $qcd = \{cd_1, cd_2, \ldots, cd_n\}$. Los elementos utilizados en la consulta son comparados con la parte del índice que contiene los componentes estructurales y retorna los BP más similares a dicha consulta.

La consulta combinada de texto y estructura: Este proceso de consulta integra las dos opciones de consulta anteriores. Para realzar este proceso el sistema forma automáticamente un vector de consulta $qmg = qpc \cup qcd$, el cual se compara con cada BP registrado en la matriz *MI*, tomando las dos zonas o componentes.

Para la comparación del vector de consulta con los BP registrados en el índice se parte de los datos introducidos en la consulta, los cuales son representadas en forma de vector de términos $q = \{t_1, t_2, t_3, \ldots, t_n\}$, además se convierten todos los términos de q a minúsculas, se eliminan palabras vacías, acentos, caracteres especiales, finalmente se aplica stemming (algoritmo de porter) para convertir cada uno de los términos de q a su raíz léxica. Con la cadena de consulta procesada se ejecuta la búsqueda en el espacio elegido por el usuario, a continuación se describa cada uno de los componentes de este nivel.

Consulta: En el proceso de ejecución de la consulta el modelo ordena y filtra los BP retornados, implementando la ecuación (2) de calificación conceptual (puntuación) definida en LUCENE [28].

$$\begin{aligned} Puntuacion\ (q,d) &= coord\ (q,d) \times \\ Qnorma\ (q) &\sum_{t \in q} \left(tf\ (t \in d) + idf\ (t)^2 \times t.getBoost \times norm\ (t,d)\right) \end{aligned} \tag{2}$$

En la ecuación anterior t es un término de la consulta q y d es el documento consultando, $tf\ (t \in d)$ es la frecuencia del término en el documento, definida como el número de veces que el término t aparecen en el BP d. En esta medida los documentos de mayor puntuación son los que contiene mayor frecuencia del término, $idf(t)$ es la frecuencia inversa del término t en un BP (número de BP en los que aparece el termino t), $coord\ (q, d)$ es un factor de puntuación basado en el número de términos de la consulta que se encuentran en el BP consultado, los BP que contienen más términos de la consulta obtienen mayor puntuación, $Qnorma(q)$ es un factor de normalización utilizado para hacer las puntuaciones (para este modelo es tomado con el valor de 1 ya que no afecta la

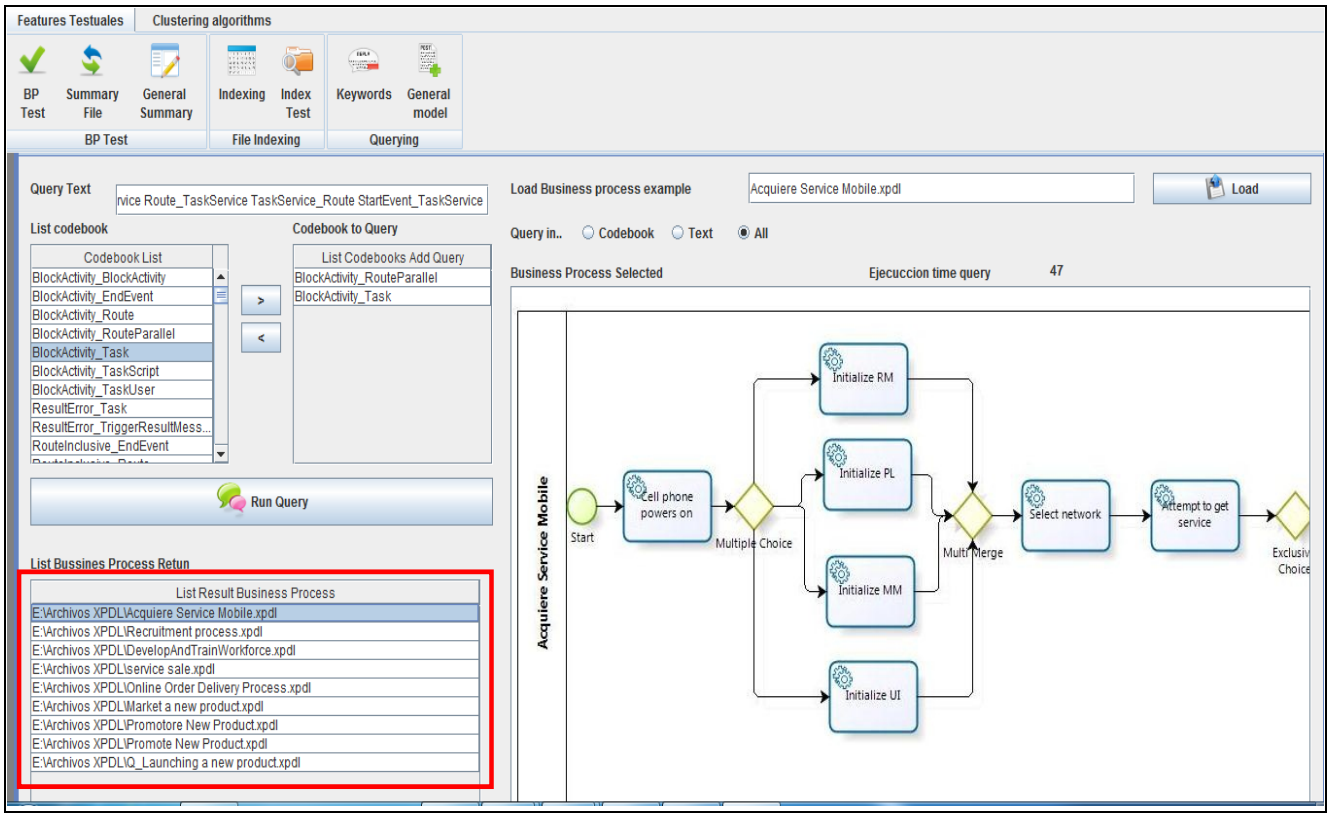

Figura 4. Opciones de consulta y despliegue de resultados en lista lineal ordenada

puntuación de cada BP evaluado). *t.getBoost*() es la ponderación del termino *t* en la consulta en este caso es igual a 1 debido que todos los términos de la consulta tienen la misma ponderación. *norm*(*t*,*d*) es un factor de ponderación en la indexación, tomado de w_{ij} en la indexación.

Una vez los resultados son ordenados y filtrados se listan en orden de acuerdo a la similitud (más similares a menos similares) que presentan con respecto a la consulta realizada por el usuario, quien puede elegir y visualizar cada uno de los modelos de BP recuperados.

Lista de resultados: Los resultados se despliegan al usuario en una lista ordenada dependiendo del nivel de relevancia, el cual es asignado obedeciendo a la puntuación definida por (2). En esta lista, el usuario puede elegir cada uno de los modelos de BP recuperados, para visualizarlos y analizarlos completamente. LaFigura 4 hace una representación gráfica de las opciones de consulta (parte izquierda central) y la lista de resultados (parte izquierda abajo enmarcada en rojo).

Nivel de agrupación: En este nivel se ejecutan los algoritmos de agrupamiento por afinidad o algoritmos de clustering [18,30] basado en las opciones de consulta explicadas en el nivel anterior, con el propósito de estructurar los resultados en grupos o familias de BP que contienen correlación en características textuales, estructurales o en ambas. Los algoritmos adaptados para este nivel son: LINGO y STC (Sufix Tree Clustering). A continuación se describen brevemente cada uno de ellos.

STC: Toma cada BP como una secuencia ordenada de términos que pueden ser textuales o estructurales, de lo cual se utiliza la información sintáctica de la secuencia para realizar la agrupación. Originalmente este algoritmo consta de tres pasos, 1) Limpiar BP, 2) Identificar clusters base y 3) Combinar clusters base. En este proyecto para aumentar el rendimiento y evitar el desarrollo de tareas redundantes del algoritmo, se eliminó el paso uno 1) Limpieza de BP, debido a que este paso se realiza previamente en el proceso de indexación.

El proceso de agrupación empieza realizando un árbol de sufijos a partir del vector que contiene todos los componentes textuales y de estructura de cada BP, se detecta una raíz, cada nodo al menos tiene dos hijos internos, las aristas entre nodos se etiquetan con una parte del texto resumen, las etiquetas de los nodos se forman uniendo el texto de las aristas, la clasificación del cluster base es realizado con la función $s(B)$, del cluster base B con frase P es: $s(B) = |B| \times f(|P|)$, donde $|B|$ = número de documentos en el cluster base B, $|P|$ = número de palabras en P que no tienen calificación 0, f = función que penaliza a las frases de una sola palabra y es lineal para frases de 2 a 7 palabras, además constante para frases mayores.

En la combinación de cluster base se tiene que en dos cluster base Bn y Bm, con tamaños $|B_m|$ y $|B_n|$. Sea $|B_m \cap B_n|$ el número de documentos comunes, La similitud entre B_n y Bm está definida como: 1 si $|B_m \cap B_n| / |B_m| > 0.5$ y $|B_m \cap B_n| / |B_n| > 0.5$ y 0 en cualquier otro caso.

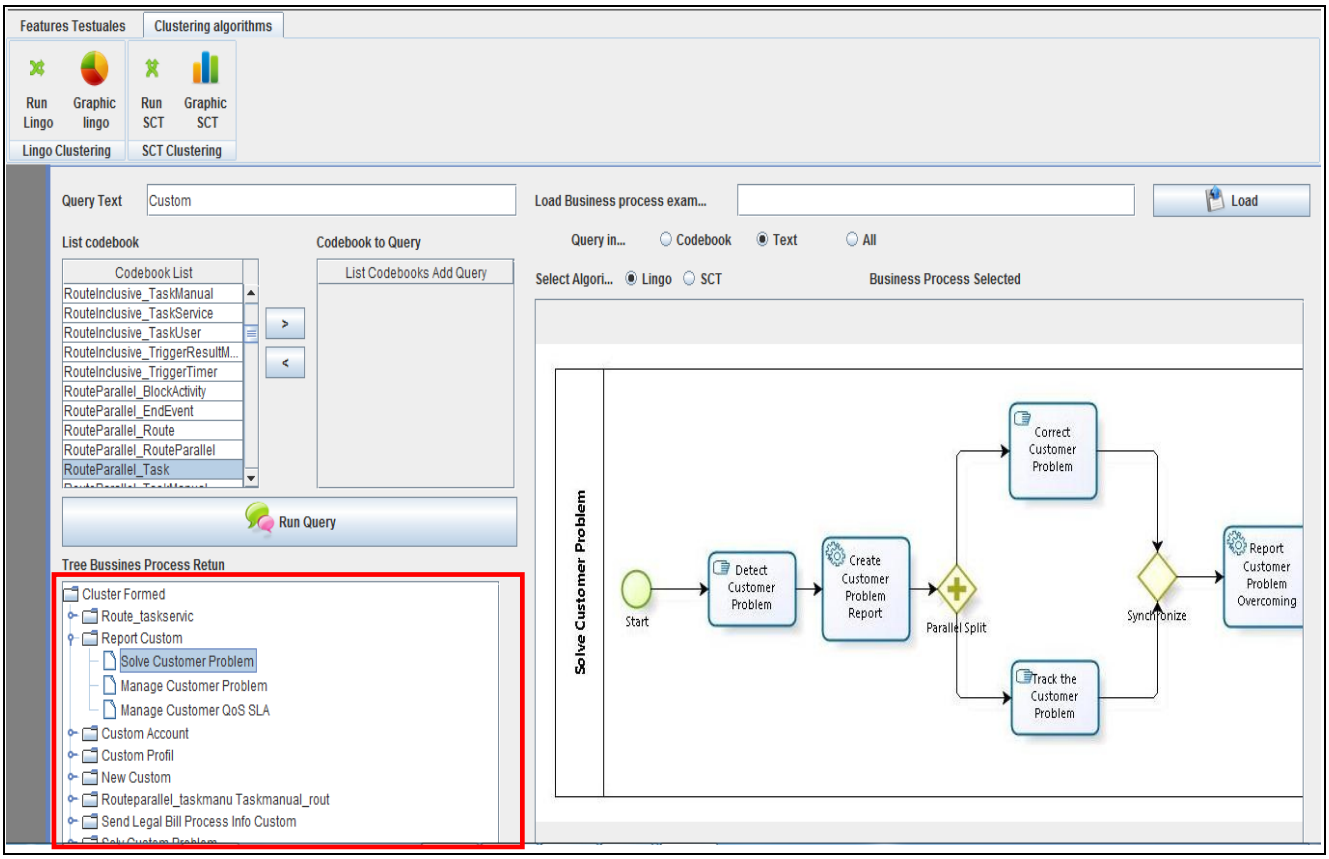

Figura 5. Opciones de consulta y despliegue de resultados en grupos temáticos

Lingo: En este algoritmo se realiza un resumen (Snippet) con los términos textuales y estructurales contenidos en cada BP descubierto en la consulta.

El algoritmo consta de cinco fases, 1) filtrado de texto, 2) extracción de características, que tiene como objeto identificar frases o términos que pueden ser candidatos para etiquetas de grupo, esto se realiza calculando el número de veces que aparecen dichas características en los BP recuperados, 3) inducción de etiquetas de cluster: en esta fase se forman descripciones significativas de grupo tomando la información de la matriz de términos por BP. Esta consta de cuatro pasos: valor del término en la matriz, el descubrimiento del concepto abstracto, la concordancia de la frase y el etiquetado, poda y evaluación, 4) descubrimiento de contenido de cada cluster: se comparan fragmentos de texto con todas y cada una de las etiquetas de grupo, para esto se forma una matriz Q en la que cada etiqueta de cluster es representada como un vector columna. De tal forma que $C=Q^TA$, donde A es el termino original de la matriz de términos por BP. De esta manera, el elemento c_{ij} de la matriz C indica el peso de adhesión del *BP j* en el grupo i, 5) formación final de clusters: se calcula con la formula valor-cluster = etiqueta-score × numero-veces, esta formación se ordena con base a la puntuación obtenida.

Al igual que en el algoritmo anterior se aumenta el rendimiento realizando la primera fase de filtrado de texto en el proceso de indexación. La Figura 5 muestra una representación gráfica de la agrupación de una consulta desplegada en forma de árbol (sección izquierda abajo enmarcada en rojo).

IV. Evaluación del entorno propuesto

Para determinar la calidad del entorno fue necesario someterlo a un proceso de evaluación experimental, con el objetivo de verificar la eficiencia en el proceso de descubrimiento de BP con base al modelo de similitud definido para las opciones de consulta que permite el entorno. Es preciso aclara que en la actualidad no se cuenta con la evaluación del proceso de agrupación. La experimentación se realizó teniendo en cuenta una colección cerrada de prueba elaborada con el juicio de ocho (8) evaluadores expertos en la temática de descubrimiento de procesos de negocio. Esta colección de prueba se realizó comparando manualmente los BP del repositorio con cada una de las consultas. En este proceso se realizaron un total de 1168 comparaciones manuales entre parejas de procesos de negocios, los cuales fueron comparados por los 8 evaluadores.

Para la evaluación se le solicitó a MultiSearchBP generar un ordenamiento (Ranking) de los 10 primeros Modelos BP (dispuestos por orden de similitud) retornados para satisfacer una necesidad definida por medio de una de las opciones de

consulta. En este sentido, es posible evaluar la calidad de los resultados obtenidos en la ejecución de esta operación del sistema, a partir de la aplicación de medidas estadísticas ampliamente empleadas en la evaluación de sistemas de recuperación de información [27], [29]. Estas medidas son la Precisión gradada (*Pg*) y el Recall gradado (*Rg*) [31], las cuales proporcionan una clasificación de los BP_t considerados similares a un BP_q de acuerdo a diferentes niveles de relevancia. De esta manera, mientras precisión y recall solo consideran la cantidad de elementos relevantes recuperados, Pg y Rg tienen en cuenta la suma total de grados de relevancia entre la consulta y los BP. En el presente trabajo se utilizaron las ecuaciones (3) y (4) [32] para evaluar *Pg* y *Rg*, relacionando el ordenamiento de los BP obtenidos por el entorno (*fe*) y el ordenamiento de las evaluaciones manuales de los expertos (*fr*). En estas ecuaciones se midió la efectividad de la recuperación de una herramienta al comparar una consulta BP_q con cada elemento de una colección BP_{ti}. Por simplicidad se considera que $BP_q = Q$ y *que* $BP_t = T$:

$$Pg = \frac{\sum_{T_i \epsilon T} \min\{f_r(Q,T_i), f_e(Q,T_i)\}}{\sum_{T_i \epsilon T} f_e(Q,T_i)}, \quad (3)$$

$$Rg = \frac{\sum_{T_i \epsilon T} \min\{f_r(Q,T_i), f_e(Q,T_i)\}}{\sum_{T_i \epsilon T} f_r(Q,T_i)}. \quad (4)$$

LaFigura 6 presenta el nivel de precisión del entorno en el descubrimiento de BP. En este proceso se desarrollaron consultas tomando como consulta 8 modelos de BP del repositorio. Los resultados de evaluación de la *Pg* en el tipo de consulta basada en la estructura alcanzaron un 41%, mientras que para las consultas realizadas por palabra clave, el entorno alcanzó un porcentaje de 76%. Finalmente las consultas realizadas con el modelo general (estructura y texto) alcanzaron el 89% de *Pg*, lo que demuestra que las consultas por modelo general (características estructurales y componentes textuales) son mucho más precisas.

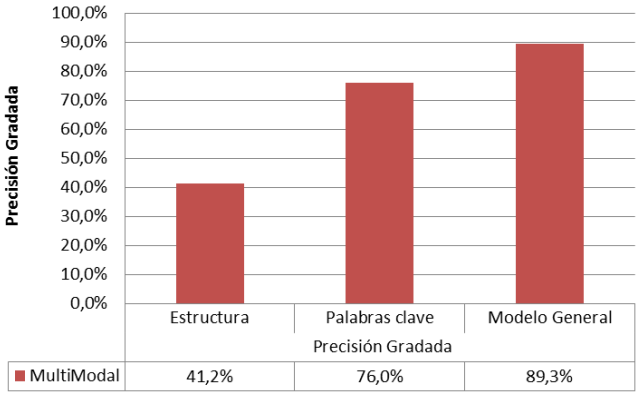

Figura 6. Grafica de precisión gradada

LaFigura 7 muestra niveles de *Rg* bajos en cada uno de los tipos de consulta, estos se encuentran en el 30% para consulta de estructura y por palabra clave (textual) mientras que el 22% para consulta por modelo general. Esto se debe a que solo se están evaluando los primeros 10 resultados y no toda la lista de resultados relevantes.

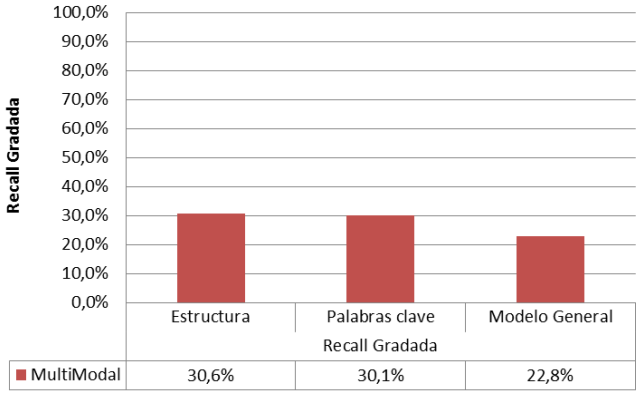

Figura 7. Grafica de recall gradada

V. Conclusiones y trabajo a futuro

En este trabajo se presentó un entorno para la búsqueda (descubrimiento) y agrupación de BP, el cual permite realizar varios tipos de consulta para ampliar el proceso de descubrimiento. Las opciones de consulta aportan flexibilidad al usuario ya que es posible replantear las búsquedas para aprovechar más el espacio de consultas y de esta forma aumentar la relevancia y pertinencia en los resultados retornados.

Los resultados obtenidos en la evaluación del entorno propuesto demuestran la eficiencia y relevancia en el proceso de descubrimiento de BP, ya que estos presentan similitud con la evaluación hecha por los expertos humanos. Alcanzando niveles de Precisión gradada que se encuentran entre el 41% como punto mínimo y 89% como punto máximo. Los resultados obtenidos en la medida de Recall gradada son bajos debido a que en el proceso de descubrimiento solo se están evaluando los primeros 10 resultados y no toda la lista de resultados relevantes, por ende no son tenidos en cuenta los BP clasificados como falsos positivos.

En el nivel de agrupación. Los grupos son formados mediante correlación y similitud directa entre características textuales, estructurales o ambas. La estructura de árbol formada permite al usuario revisar las categorías y seleccionar el grupo de mayor similitud a su consulta.

Como trabajo a futuro se propone realizar una clasificación manual de grupos de BP para poder evaluar la opción de agrupación del entorno. Evaluar la formación de grupos y comparar los resultados con otros entornos que se encuentren en el estado del arte. Incorporar ontologías de dominio específico con el propósito de realizar enriquecimiento semántico a los BP y las consultas, desarrollar un módulo de evaluación automática que genera graficas de relevancia. Ampliar la evaluación aplicando nuevas medidas para el descubrimiento de BP propuestas en [33].

AGRADECIMIENTOS

Los autores agradecen a la Universidad del Cauca y la Universidad de San Buenaventura – Cali, Colombia, por el apoyo dado al estudiante de Doctorado en Ingeniería Telemática Hugo Armando Ordóñez.

REFERENCIAS

[1] C. Cho, S. Lee, "A study on process evaluation and selection model for business process management," *Expert Systems with Applications*, vol. 38, no. 5, 2011, pp. 6339–6350

[2] Y. Gong, M. Janssen, "From policy implementation to business process management: Principles for creating flexibility and agility," *Government Information Quarterly*, vol. 29, 2012, pp. S61–S71

[3] H. Reijers, R. S. Mans, and R. van der Toorn, "Improved model management with aggregated business process models," *Data & Knowledge Engineering*, vol. 68, no. 2, 2009, pp. 221–243

[4] J. Lee, K. Sanmugarasa, M. Blumenstein, Y.-C. Loo, "Improving the reliability of a Bridge Management System (BMS) using an ANN-based Backward Prediction Model (*BP*M)," *Automation in Construction*, vol. 17, no. 6, 2008, pp. 758–772

[5] L. Xu, L. Chen, T. Chen, Y. Gao, "SOA-based precision irrigation decision support system," *Mathematical and Computer Modelling*, vol. 54, no. 3–4, 2011, pp. 944–949

[6] S. Inês, D. D. Pádua, R. Y. Inamasu, "Assessment Method of Business Process Model of EKD," vol. 15, no. 3, 2008

[7] D. Greenwood, R. Ghizzioli, "Goal-Oriented Autonomic Business Process Modelling and Execution," in: S. Ahmed and M. N. Karsiti (eds.), *Multiagent Systems*, 2009, p. 18

[8] S. Narayanan, V. Jayaraman, Y. Luo, J. M. Swaminathan, "The antecedents of process integration in business process outsourcing and its effect on firm performance," *Journal of Operations Management*, vol. 29, no. 1–2, 2011, pp. 3–16

[9] H. H. Chang, I. C. Wang, "Enterprise Information Portals in support of business process, design teams and collaborative commerce performance," *International Journal of Information Management*, vol. 31, no. 2, 2011, pp. 171–182

[10] S. Smirnov, M. Weidlich, J. Mendling, M. Weske, "Action patterns in business process model repositories," *Computers in Industry*, vol. 63, no. 2, 2012, pp. 98–111

[11] A. Koschmider, T. Hornung, A. Oberweis, "Recommendation-based editor for business process modeling," *Data & Knowledge Engineering*, vol. 70, no. 6, 2011, pp. 483–503

[12] R. Dijkman, M. Dumas, B. van Dongen, R. Käärik, J. Mendling, "Similarity of business process models: Metrics and evaluation," *Information Systems*, vol. 36, no. 2, 2011, pp. 498–516

[13] H. a. Reijers, T. Freytag, J. Mendling, A. Eckleder, "Syntax highlighting in business process models," *Decision Support Systems*, vol. 51, no. 3, 2011, pp. 339–349

[14] Z. Huang, J. Huai, X. Liu, J. Zhu, "Business Process Decomposition Based on Service Relevance Mining," *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, 2010, pp. 573–580

[15] D. Rosso-Pelayo, R. Trejo-Ramirez, M. Gonzalez-Mendoza, N. Hernandez-Gress, "Business Process Mining and Rules Detection for Unstructured Information," *2010 Ninth Mexican International Conference on Artificial Intelligence*, 2010, pp. 81–85

[16] D. R. Ferreira, "Applied Sequence Clustering Techniques for Process Mining," *Science*, April, 2009, pp. 492–513

[17] M. Qiao, R. Akkiraju, A. J. Rembert, "Towards Efficient Business Process Clustering and Retrieval: Combining Language Modeling and Structure Matching," *Lecture Notes in Computer Science*, vol. 6896, 2011, pp 199–214

[18] C. Diamantini, D. Potena, E. Storti, "Clustering of Process Schemas by Graph Mining Techniques (Extended Abstract)," *SEBD 2011*, 2011, p. 49

[19] D. Ferreira, M. Zacarias, M. Malheiros, P. Ferreira, "Approaching Process Mining with Sequence Clustering: Experiments and Findings," *Lecture Notes in Computer Science*, vol. 4714, 2007, pp 360–374

[20] J.-Y. Jung, J. Bae, L. Liu, "Hierarchical clustering of business process models," *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 12, 2009, pp. 613–616

[21] J. Melcher, D. Seese, "Visualization and Clustering of Business Process Collections Based on Process Metric Values," *10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, 2008, p. 572–575

[22] Z. Yan, R. Dijkman, P. Grefen, "Business process model repositories – Framework and survey," *Information and Software Technology*, vol. 54, no. 4, 2012, pp. 380–395

[23] H.-L. Luo, H. Wei, F.-X. Hu, "Improvements in image categorization using codebook ensembles," *Image and Vision Computing*, vol. 29, no. 11, 2011, pp. 759–773

[24] Y.-C. Hu, B.-H. Su, C.-C. Tsou, "Fast VQ codebook search algorithm for grayscale image coding," *Image and Vision Computing*, vol. 26, no. 5, 2008, pp. 657–666

[25] M. Wu, X. Peng, "Spatio-temporal context for codebook-based dynamic background subtraction," *AEU - International Journal of Electronics and Communications*, vol. 64, no. 8, 2010, pp. 739–747

[26] M. E. Fonteyn, M. Vettese, D. R. Lancaster, S. Bauer-Wu, "Developing a codebook to guide content analysis of expressive writing transcripts," *Applied nursing research: ANR*, vol. 21, no. 3, 2008, pp. 165–168

[27] C. D. Manning, P. Raghavan, H. Schütze, *An Introduction to Information Retrieval*, 2008, p. 428

[28] G. Bordogna, A. Campi, G. Psaila, S. Ronchi, "Disambiguated query suggestions and personalized content-similarity and novelty ranking of clustered results to optimize web searches," *Information Processing & Management*, vol. 48, no. 3, 2012, pp. 419–437

[29] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., 1999, p. 513

[30] F. Aiolli, A. Burattin, A. Sperduti, *A Metric for Clustering Business Processes Based on Alpha Algorithm Relations*, Technical report, 2011

[31] U. Küster, B. König-Ries, "On the Empirical Evaluation of Semantic Web Service Approaches: Towards Common SWS Test," *2008 IEEE International Conference on Semantic Computing*, 2008

[32] D. A. Buell, D. H. Kraft, "Performance measurement in a fuzzy retrieval environment," *SIGIR Forum*, pp. 56–62, 1981

[33] M. Becker, R. Laue, "A comparative survey of business process similarity measures," *Computers in Industry*, vol. 63, no. 2, 2012. pp. 148–167