ARTÍCULO

Chronic and Temporal Patterns retrieval in large data collections

D. M. Llidó, J. M. Pérez, R. Berlanga and M. J. Aramburu

Departament de Llenguatges i Sistemes Informàtics
Universitat Jaume I, E-12071 Castellón. Spain.

e-mail:{dllido,martinej,berlanga,aramburu}@uji.es

The goal of our work is to present an algorithm to automatically generate the temporal metadata of a large collection of text documents, and how to exploit this resource on current Information Retrieval Systems (IRS) and Topic Detection Systems (TDS). More specifically, we propose a method to automatically obtain the document event-time periods from the temporal references that appear in the document texts. Afterwards, we define the concept of chronicle, which is intended to detect and retrieve topics with current IRS and TDS. Finally, we apply the concept of chronicle to discovering temporal constrains between topics by using CSP techniques.

Recuperación de Crónicas y de Patrones Temporales en grandes colecciones de documentos

D. M. Llidó, J. M. Pérez, R. Berlanga and M. J. Aramburu
Departament de Llenguatges i Sistemes Informàtics,
Universitat Jaume I,
E-12071 Castellón. Spain.
email:{dllido,martinej,berlanga,aramburu}@uji.es

Resumen

En este artículo vamos a analizar cómo generar automáticamente metadatos temporales en grandes colecciones de documentos y cómo explotar este recurso para la búsqueda y detección de sucesos o para descubrir patrones temporales entre distintos sucesos. Específicamente primero proponemos un método que automáticamente extrae de los documentos las referencias temporales y las traduce a expresiones temporales en un modelo formal. En un segundo paso proponemos un algoritmo para obtener de cada documento su periodos de suceso. A continuación definiremos el concepto de crónica, la cual nos va a permitir detectar o buscar sucesos con sistemas de detección de sucesos (TDT) o sistemas de recuperación de información(IRS). Por último veremos como utilizando la definición de crónica también podemos descubrir patrones temporales entre sucesos aplicando las técnicas de CSP.

Palabras clave: Metadatos temporales para documentos, Recuperación de Información temporal.

1. Introducción

Muchos de los documentos digitales que existen en la Web poseen información temporal muy relevante para su comprensión. Los periódicos, los informes médicos, los textos legales y económicos son ejemplos de documentos cuyo contenido transcurre en un tiempo concreto, y este nos permite relacionarlos. En el campo periodístico, el conocer cuándo se producen los sucesos, permite relacionarlos con propósito de obtener información para obtener sucesos similares, crónicas sobre sucesos o ampliar información mediante la ayuda de sistemas de recuperación de información (Information Retrieval System *IRS*) [Bae99] o de detección de sucesos (Topic Detection and Tracking *TDT*)[Yan98, Pap99]. Para un economista sería más interesante poder reconocer los patrones temporales entre sucesos, en cuánto tiempo afectan unos sucesos a otros, para por ejemplo saber cuándo invertir en ciertas acciones de bolsa.

Tanto en los sistemas de recuperación de información como en los sistemas de detección de tópicos, el conocimiento de la temporalidad presente en el contenido de los documentos puede ser muy valiosa [Pap99]. De hecho, en la iniciativa de Dublín Core Metadata se han definido distintos elementos para expresar los aspectos temporales de los documentos (e.j. la fecha de creación, la fecha de publicación y la cobertura temporal del documento). Posteriormente en la *MUC entity task* [Chi97], se propuso el uso de la etiqueta TIMEX para identificar referencias temporales dentro del texto. Los sistemas de recuperación de información actuales no hacen uso de las características temporales de los documentos, principalmente porque no soportan

un conjunto de operadores temporales [Ara01]. Además, este tipo de metadatos necesita asignarse manualmente, lo cual es impracticable en aplicaciones donde el flujo de documentos es muy grande. Como resultado, los sistemas disponibles sólo tienen en cuenta la fecha de publicación, la cual en la mayoría de los casos no expresa correctamente la localización temporal del contenido de los documentos.

En este documento describiremos un método para la generación automática de metadatos temporales y la posibilidad de explotar esta información para la obtención de crónicas con los sistemas actuales de recuperación de información y detección de tópicos, y también para obtener patrones temporales entre sucesos relacionados, un problema de estudio de *Constraint Solving Problem (CSP)*. Para el primer objetivo proponemos un método para identificar automáticamente las referencias temporales que aparecen en un documento, que posteriormente serán traducidas en expresiones temporales según un modelo de tiempo formal. Para el segundo objetivo proponemos un algoritmo que calcula el *período de suceso* [Ara01] de un documento a partir de sus referencias temporales. Esta propiedad temporal nos permite simplificar la representación de la temporalidad de los documentos y su manipulación tanto en los sistemas de recuperación de información como en los de detección de tópicos. Para el último objetivo propondremos algoritmo basado en las técnicas de CSP para descubrir patrones temporales entre sucesos.

El resto del artículo está organizado como sigue. La sección 2 describe el modelo de tiempo formal y el método de extracción de las referencias temporales de los documentos. La sección 3 esta dedicada a explicar distintas características temporales que se pueden obtener de los documentos. La sección 4 define una medida de similitud entre los documentos que tiene en cuenta las características temporales de los documentos. La sección 5 está dedicada a los experimentos de agrupamiento para la obtención de crónicas que evalúan la efectividad comparando distintas aproximaciones mediante sistemas de recuperación o de detección. En la sección 6 vamos a descubrir patrones temporales entre sucesos, introduciendo previamente ciertos conceptos de CSP.

2. Extracción de referencias temporales

Generalmente las expresiones temporales en lenguaje natural involucran distintas unidades de tiempo, días, meses, años, etc. Estas unidades temporales o granularidades permiten definir el tiempo con mayor o menor precisión.

En esta sección vamos a describir un modelo formal del tiempo que tiene en cuenta las entidades temporales que aparecen en las expresiones lingüísticas. Posteriormente, presentamos la herramienta *TimeExtractor*, la cual identifica las referencias temporales presentes en los documentos de texto, las codifica según un modelo de tiempo, y resuelve las referencias temporales codificadas para tratar de obtener fechas o intervalos de fechas.

2.1. El Modelo de Tiempo

2.1.1. Granularidades

En nuestro modelo de tiempo vamos a adoptar un modelo de tiempo lineal discreto, de modo que utilizando una unidad de medida podemos establecer una isomorfía entre éste y el conjunto de números naturales. Así, cada instante de tiempo absoluto está asociado con un único número natural, y la relación $\leq$ representa una relación de orden entre instantes de tiempo.

Sobre el eje temporal, el modelo de tiempo define un conjunto de granularidades que permiten expresar el tiempo con distintos niveles de abstracción siguiendo el modelo del Calendario Gregoriano. Específicamente hemos identificado las siguientes granularidades: *día(i), díames(d), díasemana(x), semana(w), mes(m), trimestre(t), cuatrimestre(q), semestre(e), año(y), década(z), siglo(s) y milenio(l)*. Las letras entre paréntesis corresponden con los códigos que hemos utilizado para representar las granularidades. En la tabla del cuadro 1 se definen las relaciones que pueden existir entre las granularidades.

A partir de estas granularidades y sus relaciones, definiremos el calendario Gregoriano como sigue [Bet00]:

Relación	Semántica
$\overline{\wedge}$	partition
$\leq$	order
$\trianglelefteq$	group
$\preceq$	finer
$\propto$	groups periodically into
$\sqsubseteq$	sub-granularity
$\rightarrow$	equivalent

Cuadro 1: Relaciones entre granularidades

$$C = (\mathcal{G}, \mathcal{E}) \quad (1)$$

donde:

- $\mathcal{G} = \{(i, \mathbb{Z}), (x, 1..7), (y, \mathbb{Z}), (z, 1..10), (s, \mathbb{Z}), (l, \mathbb{Z}), (m, 1..12), (e, 1..2), (c, 1..3), (t, 1..4), (d, 1..\mathcal{L}_d), (w, 1..\mathcal{L}_w)\}$, donde cada par $(g, \mathcal{L}_g)$ representa a la granularidad g a lo largo de su domino $\mathcal{L}_g$,
- la granularidad día *i* es terminal, según la relación de partición $\overline{\wedge}$ y,
- $\mathcal{E}$ son las reglas que relacionan las granularidades entre sí para formar el calendario, las cuales se describen en la tabla del cuadro 2.

Granularidad	Regla
(x,1..7)	$x \rightarrow i,\ \exists\ j \in \mathbb{Z},\ k = mod(j/7) + 1$
(y,$\mathbb{Z}$)	$x(y, i) = 400$
(z,1..10)	$x(\mathbb{Z}, y) = 1, 10$
(s,$\mathbb{Z}$)	$x(s, y) = 1, 100$
(l,$\mathbb{Z}$)	$x(l, y) = 1, 1000$
(m,1..12)	$x(m, y) = 13$
(e,1..2)	$x(m, e) = 6$
(c,1..3)	$x(m, c) = 4$
(t,1..4)	$x(m, t) = 3$
(d,1..$\mathcal{L}_d$)	$d \rightarrow i$, *and* $\mathcal{L}_d \in$ [28.,31] year dependent
(w,1..$\mathcal{L}_w$)	$w \rightarrow i,\ \mathcal{L}_w \in$ [1.,7] month dependent

Cuadro 2: Reglas de generación del calendario.

Las granularidades de nuestro modelo se pueden clasificar en absolutas y relativas. Las granularidades absolutas son aquellas cuyos valores se han definido en el calendario en un dominio absoluto $\mathcal{L} = Z$, es decir, son aquellas unidades de medida que permiten expresar un instante de tiempo en el eje temporal unívocamente. En nuestro modelo son granularidades absolutas el día, año, siglo y milenio. Las granularidades relativas el dominio se define en función de otra granularidad, tienen un dominio relativo, y para definir un instante de tiempo concreto deben combinarse con una granularidad absoluta. Cabe destacar que en algunos contextos las granularidades absolutas pueden tener dominios relativos (e.j. ' el segundo año del siglo'). Hemos introducido tres granularidades equivalentes a día, *i*, *x* y *d* lo cual nos ayuda a distinguir entre el día en el dominio absoluto o con dominio relativo con respecto a semana y mes respectivamente.

2.1.2. Entidades temporales y operadores

Primero vamos a definir las entidades temporales de nuestro modelo:

- Un *Punto de Tiempo* T es secuencia alterna de granularidades y números enteros,

  $T = g_1 n_1 g_2 n_2 \ldots g_k n_k$
  donde $g_i \in \mathcal{G}$, $n_i \in \mathcal{L}_{g_i}$ y
  si $i < j$ entonces $g_j \overline{\wedge} g_i$.

  Como consecuencia las granularidades deben estar ordenadas por la relación de partición $\overline{\wedge}$. Vamos a definir una relación de orden entre dos puntos temporales con la misma secuencia de granularidades $\leq$ como :

  Sean $T = g_1 n_1 g_2 n_2 \ldots g_k n_k$ y
  $T' = g_1 n'_1 g_2 n'_2 \ldots g_k n'_k$,
  diremos que $T \leq T'$ si se cumple
  que $n_i \leq n'_i$ *con* $1 \leq i \leq k$

- Un *Intervalo de Tiempo* I es el espacio temporal entre dos puntos temporales que tienen la misma secuencia de granularidades.

  $I = [T1, T2]$, donde $T1 \leq T2$,

- Una *Duración de Tiempo*, es un espacio temporal no fijo en el dominio de tiempo, el cual se mide en función de una granularidad y que puede referenciar a un periodo futuro (+) o pasado (-), formalmente:

  $S = \pm ng$, con $g \in \mathcal{G}$ y $n \in \mathcal{L}_g$.

El conjunto de operadores temporales necesarios para extraer y resolver las referencias temporales se muestran en la tabla del cuadro 3, sus descripciones más detalladas se pueden consultar en [Lli01].

Operador	Semántica
$start(I)$	Punto inicial de I
$end(I)$	Punto final de I
$format(T, g_1..g_k)$	Formatea T con el patrón $g_1..g_k$
$refine(T, g)$	Refina T a una g más fina
$abstract(T, g)$	Abstrae T a una g más gruesa
$shift(T, S)$	Desplaza T con la duración S

Cuadro 3: Operadores Temporales.

2.2. TimeExtractor

TimeExtractor es una herramienta que detecta y analiza las expresiones temporales existentes en los documentos. Esta herramienta devuelve el documento original con las expresiones temporales etiquetadas con la etiqueta XML TIMEX [Chi97].

Para poder tratar la flexibilidad y complejidad de las expresiones del lenguaje natural, *TimeExtractor* define una representación superficial del lenguaje denominada *CodSem* la cual incluye todos los elementos del modelo de tiempo formal, además de símbolos extras para capturar la semántica de las expresiones lingüísticas.

Por ejemplo, este lenguaje incluye el código r para denotar una referencia de tiempo relativa a una fecha, mientras el código R indica una referencia relativa a un evento, el código 0 denota el presente, etc. Más detalles de este lenguaje se pueden encontrar en [Lli01].

TimeExtractor está compuesto por dos módulos que trabajan en cascada, denominados *Tagtime*, el cual extrae las expresiones temporales y las codifica según el lenguaje *CodSem*, y *ModelTimex*, el cual analiza cada expresión codificada y trata de calcular la fecha o intervalo de tiempo relacionado con la expresión.

EL módulo *Tagtime* utiliza un léxico con todos los elementos gramaticales que pueden aparecer en una expresión temporal. En este léxico las palabras están clasificadas en tres categorías:

- *núcleos temporales*, que son palabras asociadas con las granularidades de tiempo y su dominio (ej. día, semestre, junio, lunes, etc.),
- *cuantificadores*, que son los adjetivos numerales ordinales y cuantificadores (ej. Primero, segundo, dos, etc.) ,
- y *modificadores* que comprenden todas aquellas palabras que permiten identificar la dirección temporal, conocer si la expresión temporal denota una fecha, listas o intervalos de fechas, o bien un tiempo relativo a un evento o a una expresión temporal.

Todos estos elementos tienen asociado un código semántico según la descripción del lenguaje *CodSem*. *TagTime* reconoce las expresiones temporales de forma similar a los actuales sistemas de extracción de información, según se describe con más detalle en [Lli01].

El módulo *ModelTimex* utiliza las relaciones y operadores definidos en el modelo del tiempo para trasladar las referencias temporales codificadas en fechas concretas. Este módulo también utiliza la fecha de publicación de los documentos para resolver las expresiones temporales relativas al punto de referencia del lector (ej. ayer, este lunes, dentro de dos días, etc.).

La división de *TimeExtractor* en dos módulos mejora la portabilidad de esta herramienta a otros lenguajes. Así, para admitir un nuevo idioma sólo es necesario reconstruir el léxico y generar la gramática que reconoce las expresiones temporales del lenguaje. Gracias a que el modulo *ModTimex* es independiente del lenguaje del texto original, no se ha de modificar cuando se quiere admitir otro idioma.

A continuación presentamos algunos ejemplos de salida del primer módulo.

```
<TIMEX Value='-1ny$nm$nd'> ayer </TIMEX>
<TIMEX Value='0-w'>última semana</TIMEX>
<TIMEX Value='$0y$m7$d1'>día 1 de julio</TIMEX>
<TIMEX Value='I$y1992'> desde 1992 </TIMEX>
```

Estas expresiones temporales codificadas son procesadas por el segundo módulo, que produce la siguiente salida final (siendo la fecha de publicación el 1/6/99)

```
<TIMEX type='DATE' Value='y1999m06d31'> ayer </TIMEX>
<TIMEX type='DATE' Value='y1999m5w4'>última semana</TIMEX>
<TIMEX type='DATE' Value='y1999m7d1'>día 1 de julio</TIMEX>
<TIMEX type='DATE' Value='[y1992,y1999]'>desde 1992</TIMEX>
```

3. Características temporales de los documentos: Periodo de Suceso

Para la indexación de los documentos necesitamos extraer todas aquellas características que se van a utilizar en la herramienta de consulta. En nuestra aproximación cada documento d^i se representará con los siguientes metadatos:

- La fecha de publicación del documento pd^i (*publication date*).
- Un vector de términos: $T^i = (w_1^i, \ldots w_n^i)$, siendo w_k^i el peso de de cada término t_k dentro del documento d^i. Tradicionalmente este peso se corresponde con la frecuencia del término en el documento (TF), combinada con la inversa de frecuencia total del término en la colección (IDF).
- Un vector de entidades temporales: $F^i = (TF_{f_1^i} \ldots TF_{f_k^i})$ donde $TF_{f_m^i}$ representa la frecuencia de la entidad temporal f_m en el documento d^i. Las entidades temporales son extraídas de los documentos de texto utilizando las herramientas descritas en las secciones anteriores.
- El *periodo de suceso* del documento: et^i (*event time*), el cual representa el periodo de tiempo en el que se producen los hechos más relevantes descritos en el documento [Ara01].

Para determinar de forma automática el *periodo de suceso* de un documento nos vamos a apoyar en las siguientes hipótesis de partida:

- Las noticias se publican en fechas próximas a la fecha del suceso. Así pues, las referencias temporales lejanas a la fecha de publicación son referencias a sucesos indirectamente relacionados con la acción principal del artículo.
- La relevancia de las fechas que abarca un intervalo temporal es inversamente proporcional a la duración del intervalo. A modo de ejemplo, las fechas que aparecen dentro del intervalo 'de lunes a jueves' son más relevantes a las fechas que aparecen en el intervalo 'este verano'.
- Cuando los sucesos no son puntuales, generalmente se utilizan las fechas más destacadas de los sucesos, entre las cuales puede haber huecos de varios días, de modo que el *periodo de suceso* debe ser el intervalo que recubra las fechas destacadas.

El algoritmo propuesto para generar el periodo de suceso (ver algoritmo 1) tiene en cuenta estas hipótesis. Básicamente el algoritmo determina el intervalo más relevante y próximo a la fecha de publicación del documento. Para ello primero construye una lista de intervalos con las entidades temporales contiguas, permitiendo algunos huecos entre ellos. Para determinar la relevancia de cada intervalo el algoritmo utiliza la frecuencia de las entidades temporales que se solapan con el intervalo. El algoritmo solamente considera aquellos intervalos que superan un determinado umbral sobre su frecuencia de aparición en el texto.

Algorithm 1 Event Time Algorithm

Require: pd^i, D^i, I^i, th, gap, $maxdist$
{pd^i: publication date ; D^i: list of extracted points entities ; I^i: list of extracted intervals entities ; th: lowest frequency for a relevant time entity ; gap: allowed days-distance between relevant time-entities ; $maxdist$: maximal days-distance between two proximal dates}
Ensure: $Event\_Time$
for all $[f_1, f_2] \in I^i$ **do**
  if $dayDistance(f_1, f_2) \leq maxdist$ **then**
    add to D^i all dates between f_1 and f_2
    delete $[f_1, f_2]$ from I^i
  else
    add to D^i the dates f_1 and f_2
  end if
end for
$F0 = '\ '$; $cont = 0$; $F1 = ''$; $FI = ''$; $FF = ''$
for $F1 \in D^i$ **do**
  if $F0 == '\ '$ **then**
    continue
  else
    if $dayDistance(F1, F0) < gap$ **then**
      $cont+ = D^i[F1]$; $FF = F1$
    else
      $I^i[FI, FF] = cont$
      $F0 = F1$; $FI = F1$; $FF = F1$; $cont = 0$
    end if
  end if
end for
Event_Time=ExtractMostRelevantInterval($I^i, maxdist, pd^i, th$)
{Obtain most relevant interval near to the pd^i}
if not $Event\_Time$ **then**
  $Event\_Time = [pd^i - 1, pd^i]$
end if

4. Crónicas de sucesos

El concepto de *crónica* fue inicialmente definido en el modelo TOODOR [Ara01] como el resulta-

do de aplicar un operador temporal que agrupa aquellos documentos cuyos tiempos de suceso se solapaban en el tiempo. La principal idea de este operador es que cada uno de los grupos generados, denominado *crónica*, representase distintos sucesos de un tópico dado (consulta). Como se muestra en [Ber01], este operador se puede implementar sobre un sistema de recuperación de información para representar gráficamente los resultados de la consulta organizados a lo largo del eje temporal.

Nuestra experiencia nos ha mostrado que el inconveniente de este operador es que los documentos con poca relevancia y periodos de suceso largos pueden agrupar distintos sucesos en una misma crónica. Para evitar este problema, hemos redefinido el concepto de crónica añadiendo un umbral de relevancia β entre los documentos que pertenecen a la misma crónica.

Dado un conjunto de documentos D, vamos a definir como *secuencia de crónicas* $CH^g_{q,\beta}$ al resultado de agrupar aquellos documentos relevantes a la temática q, que superen cierto umbral de relevancia β, y que cumplan cierta relación temporal a nivel de la granularidad g. Cada crónica Ch de la secuecia vendrá caracteriza por un trio $(D_{Ch}, et_{Ch}, rel_{Ch})$, donde:

- D_{Ch} es el conjunto de documentos que conforman la crónica.
- et_{Ch} es el intervalo que recubre los periodos de suceso de los documentos de la crónica, formalmente:

$$et_{Ch} = [min(start(et^i))_{\forall d^i \in D_{Ch}}, max(end(et^i))_{\forall d^i \in D_{Ch}}]$$

- y rel_{Ch} es la relevancia de la crónica, la cual se determina a partir valor medio de la relevancia máxima y media de los documentos de la crónica, teniendo además en cuenta el tamaño relativo de la misma, formalmente:

$$rel_{Ch} = \frac{|Ch| \cdot max(rel(d^i,q))_{\forall d^i \in D_{Ch}} + \Sigma_{\forall d^i \in D_{Ch}} rel(d^i,q)}{2 \cdot max(|Ch_j| \cdot)_{\forall Ch_j \in CH^g_{q,\beta}}}$$

En las siguientes secciones vamos a describir dos métodos alternativos para la construcción de secuencias de crónicas sobre una gran colección de documentos. El primer método consiste en utilizar un sistema de recuperación de la información existente, sobre el cual se lanzarán las consultas de interés, y con el resultado se obtendrán las correspondientes crónicas. El segundo método consiste en utilizar un algoritmo de detección de sucesos para agrupar los documentos que estén relacionados con los mismos temas y en tiempos próximos.

4.1. Obtención de Crónicas con un sistema de recuperación de la información

Supongamos que un sistema de recuperación de información nos devuelve como respuesta a la consulta q un conjunto de documentos d, y su relevancia con respecto a la consulta denotado con $rel(d, q)$. Para la obtención de la secuencia de crónicas con un sistema de recuperación de información hemos adaptado el algoritmo presentado en [Ber01]. A modo de ejemplo la figura 1 muestra un ejemplo de la secuencia de crónicas obtenidas para la consulta sobre "Méjico", y como se ha realizado el agrupamiento de los documentos a lo largo del tiempo.

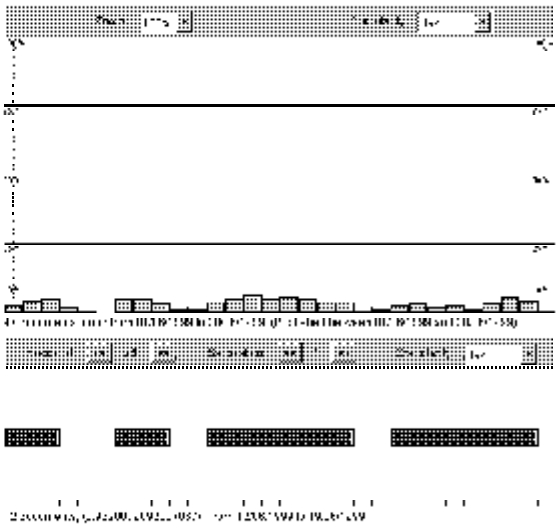

Figura 1: Secuencias de crónicas sobre Méjico

El sistema primero debe calcular el conjunto de documentos de la colección, $D^{q,\beta}$ que cumpla que $\forall d^i \in D^{q,\beta}$, $rel(d^i, q) > \beta$, donde β es un umbral que debe establecerse empíricamente. La secuencia de crónicas D_{Ch} se define entonces como el conjunto de particiones de $D^{q,\beta}$ producido por la siguiente condición:

$\forall d^j \in D_{Ch}$, si $\exists d^i \in D^{q,\beta}$, $i \neq j$, tal que:

$abstract(et^i, g) \cap abstract(et^j, g) \neq \emptyset$ entonces $d^i \in D_{Ch}$

Es decir, todos los documentos con intersección temporal serán agrupados en una misma crónica.

Una vez obtenidos los documentos de las crónicas, para cada una de ellas calcularemos el periodo de suceso y su relevancia según se ha definido en la sección anterior.

La principal ventaja de utilizar un sistema de recuperación de la información para obtener crónicas es su gran eficiencia y su simplicidad. De hecho, los interfaces diseñados para la exploración temporal de sucesos funcionan en tiempo real (ver figura 1), permitiendo al usuario interactuar con los mismos mediante la redefinición de los parámetros de las crónicas (q, g y β). Por contra, la construcción de las crónicas se hace exclusivamente en base al tiempo de suceso de los documentos, ignorando el contenido de los mismos. Esto puede provocar la fusión de sucesos distintos en una misma crónica, cuando éstos solapan de alguna forma en el tiempo. El método que se describe a continuación trata de resolver este problema.

4.2. Obtención de crónicas con un sistema de detección de tópicos

Otro método para la obtención de un conjunto de crónicas es la utilización de un sistema de detección de tópicos. Para la tarea de detección de sucesos hemos adoptado el algoritmo de *Single-Pass*, el cual es ampliamente utilizado en esto sistemas por su simplicidad y eficiencia [Yan98].

En el algoritmo 2 se muestran las modificaciones introducidas al algoritmo *Single-Pass* para que tenga en cuenta el periodo de suceso de los documentos. Brevemente, el algoritmo *Single-Pass* ubica cada documento entrante con el grupo más semejante, siempre que la semejanza sea mayor que un determinado umbral. En este caso, el grupo se actualiza con el nuevo documento. Si no existe ese grupo entonces se crea uno con este nuevo documento.

Para el cálculo de la relevancia de cada documento en cada grupo, hemos definido una semejanza que tenga en cuenta tanto la semejanza entre los términos de los documentos como la distancia temporal entre sus periodos de suceso. Esta medida se define en la sección 4.2.1 y se utiliza en la función similarity del algoritmo de*Single-Pass*. Ello da lugar a que se requieran dos umbrales, uno para la semejanza de términos β_t, y otro para la distancia temporal β_{et} a la granularidad de g.

Algorithm 2 Algoritmo de detección de tópicos inmediata

Require: $Corpus, \beta_t, \beta_{et}$
Ensure: $Clusters$
1: **for all** $article \in Corpus$ **do**
2: $max = \beta_t; c_i = FALSE$
3: **for all** $class \in Clusters$ **do**
4: $s = similarity(article, class, \beta_{et})$
5: **if** $s > max$ **then**
6: $max = s; c_i = class$
7: **end if**
8: **end for**
9: **if** c_i **then**
10: $re\_compute(c_i, article)$
11: **else**
12: $new\_cluster(clusters, article)$
13: **end if**
14: **end for**

A cada uno de los grupos generados con este algoritmo de detección le podemos asociar un periodo de suceso y una relevancia calculada del mismo modo que para las crónicas. De este modo, si seleccionamos todos los grupos que son relevantes a una consulta q, tendremos una secuencia de crónicas $CH^g_{q,\beta}$. Ahora β tiene una componente para la semejanza temporal, que se evalúa al nivel de la granularidad g, y otra para la semejanza entre términos.

4.2.1. Medida de semejanza temporal

Tanto en los sistemas de recuperación de información(RI) como en los sistemas de detección de tópicos se requiere la definición de una medida de semejanza entre los documentos. En los sistemas de RI se suele aplicar para ordenar los documentos resultado de la consulta según el grado de semejanza entre las consultas y los documentos. En los sistemas de detección de tópicos la medida de semejanza se aplica para agrupar entre sí aquellos documentos que hablan sobre el mismo suceso.

Tradicionalmente estos sistemas utilizan como medida de semejanza la función del coseno, a partir del vector de términos de los documentos:

$$S_{term}(d^i, d^j) = \frac{\sum_{k=1}^{n} w_k^i \cdot w_k^j}{\sqrt{\sum_{k=1}^{n} w_k^{i2}} \cdot \sqrt{\sum_{k=1}^{n} w_k^{j2}}} \quad (2)$$

En nuestra aproximación, debemos considerar también la similitud entre los periodos de suceso. Para ello, proponemos la siguiente media de semejanza entre dos documentos d^i y d^j:

$$S(d^i, d^j) = \begin{cases} S_{term}(d^i, d^j) & : if\ d(et^i, et^j) < \beta_{et} \\ 0 & : otherwise. \end{cases} \quad (3)$$

donde β_{et}, es el umbral temporal que determina cuando el periodo de suceso entre dos documentos está lo suficientemente próximo para pertenecer al mismo suceso. Para medir la distancia entre dos periodos de suceso utilizaremos la distancia de Minkowski [Ich94], que se define como:

$$d(f_1, f_2) = |f_1 \oplus f_2| - |f_1 \oslash f_2| + \rho \cdot (3 \cdot |f_1 \oslash f_2| - |f_2| - |f_1|) \quad (4)$$

donde $f_1 \oplus f_2$ representa la unión de los intervalos, $f_1 \oslash f_2$ representa la intersección de ambos y $|f_1|$ es la longitud del intervalo f_1.

4.3. Evaluación de los sistemas de generación de crónicas

En esta sección presentamos los resultados experimentales para los diferentes métodos propuestos de obtener las crónicas de sucesos, tanto para la recuperación de información como para la detección de tópicos. El corpus de evaluación comprende un conjunto de artículos periodísticos de Junio de 1999 publicados en "El País Digital". En esta colección de 553 artículos hemos identificado manualmente 20 tópicos, los cuales contienen 61 sucesos. Merece la pena destacar que los tópicos seleccionados abarcan los sucesos ocurridos al final de la 'Guerra de Kosovo', los cuales presentan un alto solapamiento temporal.

En la evaluación hemos diferenciado entre las crónicas que relatan sucesos y las que relatan tópicos. Entendemos que un suceso es algo que ocurre en un periodo específico de tiempo y en un lugar determinado, mientras que un tópico es un conjunto de sucesos directamente relacionados con un suceso especialmente relevante. Por ejemplo, en nuestro corpus de evaluación, al final de la guerra de Kosovo, se define la secuencia de tópicos 'Acuerdo de Paz','Activación del plan' y 'Conflicto con los refugiados'. El primer tópico se compone a su vez de dos sucesos: 'Negociaciones de Paz' y 'Firma del Acuerdo'.

La efectividad de las distintas aproximaciones se mide utilizando la medida F1 [Bae99] entre cada tópico i etiquetado manualmente y cada grupo j generado por el sistema. El F1 global se obtiene promediando según el tamaño de los grupos [Pon02]:

$$F1 = \frac{1}{N_{docs}} \sum_{i=1}^{N_{level}} n_i\ F1(i, argmax_j\{F1(i,j)\}) \quad (5)$$

donde N_{docs} es el número total de documentos de la colección y N_{level} es el número total de grupos en el nivel evaluado (*sucesos* o *tópicos*), n_i el número de documentos del grupo i, y el $F1(i, argmax_j\{F1(i,j)\})$ es el F1 máximo obtenido para un grupo manual i.

La tabla del cuadro 3 muestra los resultados de las aproximaciones propuestas para los valores óptimos de sus parámetros. También incluímos los mejores resultados para el algoritmo de *Single-Pass* sin utilizar los periodos de suceso, y el sistema de crónicas propuesto originariamente en [Ara01]. Como puede verse, la aproximación a las *crónicas* con un sistema de RI mejora si se añade un umbral de relevancia al nivel de sucesos, pero no al nivel de tópicos. Nosotros atribuímos este comportamiento a las consultas realizadas en el sistema de RI, que expresa de un modo muy específico los tópicos requeridos.

Claramente, los mejores resultados de ambos niveles se han obtenido con el algoritmo de *Single-Pass* utilizando el periodo de suceso. Además, los resultados de esta aproximación son tan buenos como los obtenidos con otros sistemas de detección de sucesos más sofisticados de la literatura. Sin embargo, es necesario

evaluar el sistema con una colección estándar como las utilizadas en las conferencias del MUC para la tarea de detección de tópicos [TDT98].

Método	Sucesos	Tópicos
Crónicas RI [Ara01]	0.404	0.647
Crónicas RI (β)	0.534	0.643
Single-Pass	0.539	0.675
Single-Pass (et)	0.677	0.692

Cuadro 4: Comparativa se los sistemas con la medida F1.

Si bien sucesos y tópicos podrían especificarse mediante una combinación de palabras clave en un sistema de recuperación, éstas no son suficientemente expresivas para obtener la información que se requiere, sobre todo a nivel de sucesos, por lo cual funciona mejor el sistema de detección de sucesos que el sistema de recuperación de información.

5. Descubrimiento de patrones temporales entre sucesos

En esta sección vamos a analizar cómo pueden utilizarse las crónicas de sucesos descritas en las secciones anteriores, para descubrir patrones temporales entre sucesos. Entenderemos como *patrón temporal* a un conjunto de temas (sucesos o tópicos) que están relacionados entre sí mediante relaciones temporales, especialmente aquellas que expresan la simultaneidad de tópicos, y las relaciones de causa-efecto.

Para formalizar el concepto de patrón temporal introducimos la definición de *estructura de secuencias de crónicas*, denotada CHS, y que consiste en un grafo acíclico $(\mathcal{CH}, TC)$, donde $\mathcal{CH}$ es un conjunto de secuencias de crónicas $CH^{g_i}_{q_i,\theta_i}$, cada una de ellas obtenidas para un tema q_i a una granularidad g_i, y TC contiene los arcos del grafo que representa las relaciones temporales entre dichas secuencias. Estos arcos tienen la forma $CH_i\ R_{i,j}\ CH_j$ con $i \neq j$.

Para las relaciones temporales, se han considerado tres tipos de relaciones sobre intervalos: $before(x,g)$, $overlaps(x,g)$ y $during(x,g)$. Estas relaciones aportan la semántica suficiente para expresar patrones temporales sobre sucesos. Cada relación contiene dos parámetros: un conjunto de distancias características para la relación (x), y la granularidad asociada a dicha distancia (g). En este trabajo, el conjunto de distancias características se representa mediante un intervalo, el cual indica todas las distancias que deben analizarse en el proceso de descubrimiento de patrones. Precisamente, el algoritmo de descubrimiento tratará de seleccionar las distancias características más frecuentes de cada relación. En el cuadro 5 se presenta la semántica de estas relaciones temporales, donde los extremos inferior y superior de un intervalo I se representan como I^- y I^+ respectivamente, y las operaciones de desplazamiento temporal como dec y inc.

Relación	Semántica
$c_1\ before(x,g)\ c_2$	$dec(abs(c_2,g)^-, x^+) \leq abs(c_1,g)^+ \wedge$ $abs(c_1,g)^+ \leq dec(abs(c_2,g)^-, x^-)$
$c_1\ overlaps(x,g)\ c_2$	$inc(abs(c_2,g)^-, x^-) \leq abs(c_1,g)^+ \wedge$ $abs(c_1,g)^+ \leq inc(abs(c_2,g)^-, x^+) \wedge$ $abs(c_1,g)^- \leq abs(c_2,g)^-$ $abs(c_1,g)^+ \leq abs(c_2,g)^+$
$c_1\ during(x,g)\ c_2$	$inc(abs(c_2,g)^-, x^-) \leq abs(c_1,g)^-$ $abs(c_1,g)^- \leq inc(abs(c_2,g)^-, x^+)$ $abs(c_1,g)^+ \leq abs(c_2,g)^+$

Cuadro 5: Relaciones Temporales

A partir de las definiciones anteriores, definimos un *problema de descubrimiento temporal* como una estructura de crónicas CHS donde las distancias características de las relaciones temporales deben tener un soporte superior a un determinado umbral θ. El soporte de una distancia característica será el porcentaje de crónicas que cumplan la relación temporal con esa distancia característica, y se representará con un número real entre 0 y 1.

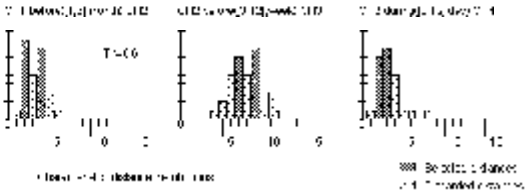

Figura 2: Ejemplo de problema de descubrimiento temporal

A modo de ejemplo, en el campo de la Economía, un posible patrón temporal podría involucrar los siguientes sucesos: los incrementos de exportación de la UE a EE.UU. (q_1), el cambio de divisas euro/dolar (q_2), los decrementos de los intereses en la UE (q_3), y los incrementos en los índices de los mercados europeos (q_4). Las relaciones temporales que resultan intere-

santes en este contexto son las siguientes: que q_1 ocurra entre 1 y 3 meses antes que q_2, que q_2 ocurra entre 3 y 12 semanas antes que q_3, y que q_2 sea ocurra simultáneamente durante 0 y 12 días con respecto a q_4. Este patrón se expresaría con la siguiente estructura de secuencias de crónicas:

$CHS =$
$(\{CH^m_{q_1}, CH^d_{q_2}, CH^w_{q_3}, CH^d_{q_4}\},$
$\{CH^m_{q_1} before([1,3],m) CH^d_{q_2},$
$CH^d_{q_2} before([3,12],w) CH^w_{q_3},$
$CH^d_{q_2} during([0,12],d) CH^d_{q_4}\}).$

La solución propuesta para un problema de descubrimiento temporal consiste en interpretar la estructura de secuencias de crónicas como una red de restricciones temporales, sobre la cual se pueden aplicar los conceptos y técnicas de CSP (Constraint Solving Problem).

La estrategia del *Algoritmo de Descubrimiento* [Ber98] se muestra en el Algoritmo 6, y consiste en determinar la red mínima de una estructura de secuencias de crónicas. A partir de la red mínima, se calcula para cada relación temporal $(CH_i\ R_{ij}\ CH_j)$ la frecuencia de cada distancia característica definida para R_{ij}. Esta frecuencia será la cantidad de crónicas en CH_i que cumplen la relación R_{ij}, en relación al número total de crónicas en CH_i.

En este caso, como la red temporal no contiene relaciones disjuntas, el problema puede resolverse en tiempo polinomial [Dec91], y para ello basta forzar la consistencia entre arcos (arco-consistencia) [Bes94].

Dada una estructura de secuencias de crónicas, el arco $R_{i,j}$ que conecta los nodos CH_i, CH_j, diremos que es *arco-consistente* si para cada periodo de suceso I_i de la secuencia de crónicas representada por CH_i existe otro periodo I_j de otra secuencia CH_j tal que la se cumple que $(I_i\ R_{ij}\ I_j)$. El algoritmo 4 muestra como conseguir la arco-consistencia.

Una vez forzada la arco-consistencia, el grafo original y la red mínima deben compararse para determinar la distribución de frecuencias de cada relación temporal en términos de la duración que la caracteriza. Para ello, se utiliza el Algoritmo 6, el cual toma como entrada el problema de descubrimiento (CHS, θ) y devuelve para cada relación una lista con las distancias que se producen con una frecuencia mayor que θ. En la figura 2 se muestra gráficamente un ejemplo de solución para el problema descrito anteriormente.

Con la información proporcionada por este algoritmo, los usuarios pueden ajustar la estructura de las secuencias de crónicas con las distancias más frecuentes halladas. Para ello primero cada arco debe reducirse a las distancias características seleccionadas. Si alguno de los arcos no tiene valores seleccionados entonces el problema de descubrimiento no tiene solución. En otro caso, las secuencias de crónicas deben reducirse a las crónicas que satisfagan los arcos reducidos. Este paso puede obtenerse forzando de nuevo la consistencia de los arcos modificados. El resultado final es una estructura que contiene los documentos relevantes que se refieren a los patrones temporales más frecuentes entre los sucesos relatados.

Algorithm 3 Algorithm Revise (input/output $Ch1$, input $Ch2$, R)

```
Require: Ch1,Ch3, R
Ensure: CH1 {Ch1, Ch2 are chronicle sequences and R is
    the temporal relationship between them.
    I1 and I2 are two time intervals}
1: for all I1 in the sequence of Ch1 do
2:   for all I2 in the sequence of Ch2 do
3:     if not (I1 R I2) then
4:       remove the chronicle of I1 from Ch1
5:     end if
6:   end for
7: end for
```

Algorithm 4 Algorithm Arc_Consistent

```
Require: CHS
Ensure: CHS { CHS is a chronicle structure, Q is a queue
    of chronicle sequences, Ch1 and Ch2 are chronicle se-
    quences}
1: enqueue all the chronicle classes of CHS into Q follo-
    wing a topological order of CHS
2: while Q is not empty do
3:   pop a chronicle class Ch1 from Q
4:   for all arc R from Ch1 to Ch2 do
5:     revise(Ch1, Ch2, R)
6:   end for
7: end while
```

6. Conclusiones

En este artículo se han presentado distintas técnicas para representar la temporalidad del los documentos. Primero, hemos propuesto un método para extraer las entidades temporales de los documentos de texto. Posteriormente hemos presentado un algoritmo para obtener el periodo de suceso de los documentos teniendo

Algorithm 5 Algorithm Count_Coo

Require: $Ch1, Ch2$
Ensure: LC {**input:** $Ch1, Ch3$ are two chronicle classes, R is a temporal relationship. **output:** LC is an indexed list of counters}
1: Let X be the characteristic distance interval of R and g its associated granularity
2: Let LC be a list with as many counters as elements in the range of X
3: Counters in LC are initialized to 0
4: **for all** interval $I1 \in$ the sequence of $Ch1$ **do**
5: Visited =
6: **for all** interval $I3 \in Ch3$ **do**
7: **if** $(I1\ R\ I3)$ **then**
8: Let D be the R's characteristic distance between $I1$ and $I3$
9: **if** $D \notin Visited$ **then**
10: $LC[D]++$
11: Visited = Visited $\cup$ D
12: **end if**
13: **end if**
14: **end for**
15: **end for**

Algorithm 6 Algorithm Frequencies

Require: CHS, θ
Ensure: Fr {CHS is a chronicle structure, θ is the minimum support for the characteristic distances; Fr is a matrix of frequencies associated to CHS} { Vfr is a list of counters}
1: $CHS' = CHS$;
2: ArcConsistent(CHS')
3: **for all** arc $R_{i,j} \in CHS'$ **do**
4: Let Ch'_i and Ch'_j be the two chronicle classes connected by $R_{i,j}$ in CHS'
5: CountCoo($Ch'_i, Ch'_j, R_{i,j}, Vrf$)
6: **for all** index $D \in Vrf$ **do**
7: **if** $Vrf[D]/length(Ch_i) > \theta$ **then**
8: $Fr[i,j,D] = Vrf[D]/length(Ch_i)$
9: **end if**
10: **end for**
11: **end for**

en cuenta las entidades temporales previamente extraídas. Este periodo de suceso constituye el intervalo de tiempo más relevante para ubicar los contenidos del documento.

Posteriormente, hemos descrito cómo las características temporales de los documentos pueden utilizarse en los sistemas de recuperación de la información y los de detección de tópicos, gracias a la introducción del concepto de *crónica*, y de nuevas medidas de semejanza entre documentos. Los resultados experimentales demuestran que la inclusión del periodo de suceso mejora la efectividad de este tipo de sistemas, siendo comparables con los obtenidos en sistemas de detección de tópicos más sofisticados.

Finalmente, las herramientas propuestas permiten la exploración temporal de la información contenida en una gran colección de documentos. Por otro lado, el concepto de *crónica* permite el descubrimiento de patrones temporales frecuentes entre los sucesos relatados en dichos documentos, utilizando para ello técnicas de resolución de restricciones (CSP). En resumen, este conjunto de técnicas supone un notable avance frente a los sistemas actuales de búsqueda de información, que están generalmente limitados a la especificación de una serie de palabras clave.

Referencias

[Ara01] M. J. Aramburu y R. Berlanga. "A Temporal Object-Oriented Model for Digital Librares of Documents". Concurrency: Practice and Experience, Vol. 13 (11), 2001.

[Ber01] R. Berlanga, J. M. Pérez, M. J. Aramburu y D. Llidó. "Techniques and Tools for the Temporal Analysis of Retrieved Information". Database and Expert System Applications (Lecture Notes in Computer Science 2113), Ed. Springer-Verlag, pp. 72-81, Munich, 2001.

[Bet00] C. Bettini, S. Jajodia and X. S. Wang. "Time Granularities in Databases, Data Mining, and Temporal Reasoning", Springer-Verlag, 2000.

[Chi97] N. Chinchor. "MUC-7 Named Entity Task Definition". September 1997. http://www.muc.saic.com/proceedings/ne_task.html

[Lli01] D. M. Llidó, R. Berlanga, M. J. Aramburu. "Extracting temporal references to automatically assing document event-time periods". Database and Expert System Applications (Lecture Notes in Computer Science 2113), Ed. Springer-Verlag, pp. 72-81, Munich, 2001.

[Pap99] R. Papka. "On-line new event detection, clustering and tracking". PhD Disertation, Departament of Computer Science. University of Massachusetts, 1999.

[Pon02] A. Pons, R. Berlanga y J. Ruiz-Shulcloper. "Detecting Events and Topics by Using Temporal References". Lecture Notes in Artificial Intelligence, Vol. 2527, Ed. Springer-Verlag, pp. 11-20, 2002.

[Ich94] M. Ichino and H. Yagushi. "Generalized Minkowski Metrics for Mixed Feature-Type Data Analysis". IEEE Transactions on Systems, Man and Cybernetics, Vol. 24 (4), 1994.

[TDT98] National Institute of Standards and Technology. The Topic Detection and Tracking Phase 2 (TDT2) evaluation plan. version 3.7, 1998.

[Yan98] Y. Yang, J. Carbonell, R. Brown, T. Pierce, B. T. Archibald, X. Liu. "Learning approaches for Detecting and Tracking News Events". In Proc. of ACM/SIGIR, 1998.

[Bae99] R. Baeza-Yates and B. Ribeiro-Neto. "Modern Information Retrieval". Ed. ACM-Press, 1999.

[Bes94] C. Bessière. "Arc-consistency and Arc-consistency Again" Artificial Intelligence Vol. 65(1), pp 179-190, 1994.

[Dec91] R. Dechter,I. Meiri and J. Pearl. "Temporal Constraint Networks" Artificial Intelligence Vol. 49(1), pp 61-95, 1991.

[Ber98] R. Berlanga, M.J. Aramburu and F. Barber "Discovering Temporal Relationships in Database of Newspaper".(IEA-98-AIE). Published in "Tasks and Methods in Applied Artificial Intelligence", LNAI 1416, pp 36-45, Springer Verlag, 1998.