# Generación Automática de Meta-Datos y Modelo ER para Fuentes de Información No-Estructuradas

Javier Gramajo and David Riaño

Universidad San Carlos de Guatemala
Universidad Rovira y Virgili (URV) Tarragona, España
{jgramajo, drianyo}@etse.urv.es

**Resumen** El constante incremento de datos en Internet requiere de sofisticadas herramientas para recuperar información. En este artículo introducimos el sistema GINY el cual nos permite organizar información obtenida de Internet, GINY genera una estructura de datos basándose en un Modelo Entidad-Relación o en un formato de Descripción de Contenidos RDF, dicha estructura se obtiene utilizando algoritmos de aprendizaje inductivo (clustering).

## 1 Introducción

Internet ha experimentado un rápido crecimiento desde su concepción en 1969. Con la introducción de los motores de búsqueda[1] podemos realizar búsquedas que nos permiten acceder a toda la información que dicho motor tenga indexada. Algunos de los más potentes buscadores anuncian que son capaces de manejar el tamaño de la Web, esto significa que deberán indexar todo su contenido, el cual puede equipararse a una enciclopedia que contiene 15-billones de palabras.

Podemos decir que la Web es una fuente de información distribuida, dinámica y de rápido crecimiento, estas características representan dificultades para los sistemas tradicionales de recuperación de información ya que dichos sistemas han sido diseñados para entornos distintos Típicamente se han utilizado para la indexación y recolección directa de documentos. Por la naturaleza de la Web se plantean dos interrogantes relacionados con los motores de búsqueda: ¿puede una arquitectura centralizada ocuparse de un número amplio de documentos? y ¿pueden los buscadores actualizar regularmente sus bases de datos cuando se detectan modificaciones en la red?.

Las respuestas a estas preguntas están relacionadas con la mejora en la metodología de búsqueda en la Web. En las últimas décadas, los sistemas de acceso a Internet persiguen la búsqueda, obtención y representación de información distribuida, con la finalidad de satisfacer las necesidades de un usuario. Por ello se cuenta con sistemas que adoptan distintas estrategias para cada una de las características relacionadas con una búsqueda(definición de la búsqueda,

[1] Motores de Búsqueda: Alta Vista (www.altavista.com) o Excite (www.excite.com)

información recuperada y definición del resultado). A continuación se mencionan dichas características y estrategias adoptadas, respectivamente.

- *Definición de la búsqueda* : Listas de términos, Bases de Datos, Bases de Conocimiento u Ontologías.
- *Información recuperada* : Datos o referencias a las páginas de origen de la información.
- *Descripción del resultado* : Listas de términos, Bases de Datos, Bases de Conocimiento u Ontologías.

En este trabajo se presenta una metodología en la que se realizan procesos que nos permiten transformar el conjunto de datos recuperados tras una búsqueda en Internet, en un modelo conceptual de datos Entidad-Relación (ER) o en una Descripción de Contenidos (RDF). El proceso de transformación al que nos referimos lo realizamos aplicando estrategias de aprendizaje inductivo.

En la sección 2 se hace la descripción del sistema GINY [1], en la sección 3 se explica el proceso realizado para la obtención del modelo conceptual de datos, en la sección 4 se presentan las pruebas realizadas y los resultados obtenidos, para finalizar con las conclusiones en la sección 5.

## 2 Descripción del Sistema GINY

GINY es una herramienta que nos permite estructurar contenidos de manera automática para lo cual se han elegido estrategias de aprendizaje inductivo. En concreto hemos elegido un proceso de Clustering [2] para encontrar la estructura que almacene los datos de un dominio descrito por una palabra clave y un conjunto de propiedades.

En la figura 1 se presenta el modelo del sistema. Los componentes de GINY son los siguientes:

- Módulo de definición de la búsqueda.
- Motor de búsqueda.
- Módulo de generación de la estructura de datos.
- Componente encargado del almacenamiento y manipulación de los datos.

GINY se organiza como un front-end, en el que un usuario realiza consultas y obtiene respuestas sobre una base de datos centralizada. Por otro lado tenemos el back-end en el que se tienen tanto un módulo de gestión de conocimiento para definir el dominio de búsqueda, un motor de búsqueda y la aplicación que se encarga de estructurar la información, generando para ello los scripts de construcción de una base de datos o de Descripción de Contenidos.

Para comprender el funcionamiento de estos componentes se hace un recorrido por el modelo de la figura 1: Un usuario que hace una consulta, utiliza una ontología de definición de la consulta, la consulta es enviada al motor de

búsqueda y se recibe una respuesta en forma de tabla de valores conteniendo los datos recuperados de Internet que están relacionados con el dominio de búsqueda que representa la ontología inicial. Esa tabla es entregada al módulo de generación de la estructura de datos que genera el modelo conceptual donde se almacenará toda la información recibida. El modelo conceptual puede transformarse en un script DDL de definción de una Base de Datos relacional o en un esquema RDF de Definición de contenidos. Los datos son volcados sobre la estructura generada para luego dar respuesta a las consultas del usuario.

**Figura 1.** Modelo del Sistema GINY.

### 2.1 Definición de la Búsqueda

Se tienen dos tipos de usuarios: el usuario que define los datos que el sistema busca y organiza y el usuario que utiliza los datos ya organizados. El módulo de definición de la búsqueda será utilizado únicamente por el primer tipo de usuario.

La definición se realiza utilizando una Ontología que caracteriza el dominio con el cual trabajaremos. El usuario que haga la definición deberá conocer y estar relacionado con el dominio. Para este trabajo se presenta una simplificación del modelo general en la que la búsqueda queda definida por una palabra clave y una lista de propiedades.

### 2.2 Motor de Búsqueda

El motor de búsqueda recibirá una definición de dominio que utilizará para buscar documentos en la Web que contengan información que coincida con la definición. Así, los documentos que encontremos en Internet deberán ser analizados en contenido para poder extraer la información definida y requerida por la Ontología.

### 2.3 Analizador de Lenguaje Natural y Generador de la Estructura

El Analizador de Lenguaje Natural recibe documentos en HTML [3], los cuales satisfacen los requerimientos de información hechos en la definición de la búsqueda. Estos documentos son analizados para encontrar las características definidas. El resultado del análisis es una matriz que contiene los datos de las propiedades definidas. Dicha matriz es analizada por medio de algoritmos de clustering, con los cuales se obtendrá las particiones de datos (entidades) y sus vínculos (relaciones), es decir la información será organizada y almacenada de acuerdo a un modelo conceptual de datos generado a partir de los mismos datos.

### 2.4 Almacenamiento y Manipulación de los Datos

Una vez obtenida la estructura se realiza el almacenamiento y la manipulación de los datos. Para la definición se emplea el Data Definition Language, DDL. Para la utilización y los procesos de actualización de los datos el Data Management Language, DML. Tanto DDL como DML, son lenguajes ejecutados por el SQL-DBMS de cualquier sistema de base de datos relacional.

## 3 Descripción del Proceso

La aportación que hace GINY es la de transformar una matriz de datos con información de nuestro interés en un modelo conceptual de datos.

Para la estructuración de contenidos se utilizan algoritmos de clustering como estrategia, una vez obtenida la estructura de los datos (clusters) se pueden obtener dos scripts, el primero con comandos SQL [4] para construir un modelo Entidad-Relación (ER) [5], el segundo script es para el sistema de descripción de contenidos (RDF) [6]. RDF es una DTD (definición del tipo de documento) de XML[7], una aplicación de meta datos que utiliza XML a fin de proporcionar un marco estándar para la interoperabilidad en la descripción de contenidos Web.

### 3.1 Extracción automática de la Estructura de Datos

Los datos obtenidos de Internet son organizados en una matriz de datos en donde cada columna tiene asociado un nombre y las filas de la matriz son los

valores asociados a los nombres. Desde la perspectiva de una base de datos se puede decir que la matriz es vista como una tabla, las filas son los registros y las columnas los campos de la tabla. La extracción de la estructura de datos es un procedimiento en el cual la matriz inicial es utilizada para construir un modelo Entidad-Relación, que se emplea para implementar una base de datos relacional.

**Figura 2.** Pasos para la generación del modelo conceptual.

En la figura 2 se describe el proceso, en el cual se siguen tres pasos:

1. Captura de los datos en una matriz.
2. Detección de las tablas y sus relaciones.
3. Generación del script que nos permitirá implementar la estructura de datos ya, sea como una base de datos relacional o como una Descripción de Contenidos RDF, para el manejo posterior de los datos por parte del usuario final.

### 3.2 Detección de Tablas y Relaciones

La *detección de tablas y relaciones* es un proceso que involucra la generación de una matriz de distancias entre las propiedades representadas por las columnas de la matriz, la determinación de los clusters que configuran las entidades del

modelo conceptual de datos, la reducción de las repeticiones que determina la cardinalidad de las relaciones entre entidades y la generación de las relaciones.

La matriz de distancias se construye a partir de la función de correlación de Pearson aplicada a los datos de entrada. Los valores de la matriz son tomados en su valor absoluto por ser de interés el grado de la relación entre propiedades (magnitud) y no el carácter de la relación (signo).

Una vez que se tiene la matriz de magnitudes se procede a determinar los clusters. Para ello se utiliza el Algoritmo Johnson [2]. La determinación de los clusters se hace partiendo de un valor pivote que relaciona dos columnas. Dicho proceso puede realizarse siguiendo dos estrategias: *Single-link* y *Complete-link*. La diferencia entre una u otra estrategia radica en la selección del valor mínimo o máximo en el momento de determinar el pivote en cada iteración del algoritmo. El número de clusters estará determinado por el valor de entrada *NúmeroDeClusters*, el cual puede oscilar entre 1 y el número máximo de columnas de la matriz de datos de entrada.

Para el siguiente paso, la *reducción de las tablas* generadas, se compara cada par de filas entre si. Para esto utilizamos la función de mínimos cuadrados, con la cual determinamos si las filas de una tabla son similares. Esta función nos devuelve un valor el cual comparamos con un parámetro de entrada, llamado *ValorDeConfianza* que está comprendido en un rango del 0% al 100%, y que nos permite determinar el nivel de confianza. Si el valor de la función de mínimos cuadrados es menor al parámetro, se elimina una de las dos filas comparadas en la tabla. Este proceso se aplica a todos los clusters generados, con lo cual se determina un nivel de reducción en cada cluster, al cual llamaremos *NivelDeReducción*.

Como último paso se procede a *determinar las relaciones* que existen entre cada una de las clases generadas para lo cual se utiliza el algoritmo que se presenta en la figura 3. La determinación de las relaciones entre clusters está condicionada a si el valor de la distancia mínima entre dos clusters es mayor o igual a *MinValor* y si el valor de la distancia de la columna que se está tratando tiene una reducción del 0%, o no. La evaluación de esta condición determina si se procede a relacionar las clases $i$ y $j$ con una relación M:N (ambas tablas tienen una reducción de elementos), 1:N (sólo una de las tablas tiene una reducción de filas), o 1:1 (ninguna de las tablas tiene reducción).

### 3.3 Generación de Scripts

La generación de los scripts se hace a partir del modelo conceptual Entidad-Relación el cual está compuesto de entidades, atributos y relaciones, estos elementos se transforman en tablas, columnas y llaves foraneas para el modelo de datos y en clases, propiedades, y descripciones para el formato de Descripción de Contenidos RDF. Así una entidad $E$ con atributos *a1, a2,..an* se transforma en la tabla $T$ o en la clase $C$ que se muestra en la figura 2

```
for i =1..NumeroDeClusters - 1 {
  Crear tabla i con atributos del cluster[i];
  if (nivel_reduccion_tabla[i] >  0%)
  then {
    for j = i+1 .. NumeroDeClusters {
      if (Abs(distancias[i,j]) >= Abs (MinValor))
          and (nivel_reduccion_tabla[j] == 0%)
      then
        Agregar referencia de cluster[j]
           como llave foranea en tabla i;
    }
  }
}
```

**Figura 3.** Algoritmo de generación automática de relaciones en un modelo conceptual de datos.

## 4 Pruebas y Resultados

La extracción automática de la estructura ha sido probada con dos dominios provenientes del repositorio de la Universidad de Irvine (UCI), *Bupa* consta de 7 atributos y 345 elementos mientras *Page-segments* consta de 18 atributos y 210 elementos.

Estos dominios son representados por matrices a las que se ha aplicado el proceso para descubrir el modelo conceptual de datos. Todos los dominios con las propiedades (columnas) son de tipo numérico y no contienen valores *missing*, con lo cual se han eliminado factores que no son prioritarios en la investigación realizada hasta el momento. A continuación describimos dos de los cinco dominios probados:

*Page-segments* está relacionado con el análisis de bloques en documentos y su contenido, la separación de texto y gráficos es un paso que se sigue en el proceso de análisis, por lo que se definen variables que puedan determinar las dos áreas.

*Bupa* es un dominio relacionado con desórdenes del hígado. Las variables determinan factores en la sangre asociados con el consumo excesivo de alcohol en individuos de sexo masculino.

Los parámetros de configuración utilizados con cada matriz se han determinado de la siguiente forma: para cada matriz de datos se generan tres clases (*NúmeroDeClases = 3*) y se define un valor de confianza del 2 % ( *ValalorDeConfianza = 0.02*), hemos seleccionado como estrategia de clustering *Complete-link* para los dominios probados.

Los resultados obtenidos en cada uno de los casos se muestran en las figuras 4 y 5. Para la matriz de datos *Page-segments* el modelo Entidad-Relación se transforma en el diseño de base de datos de la figura 4, donde se aprecia la

unión de las columnas en las tablas *NAME0*, *NAME1*, *NAME2*. A continuación se describen cada una de las tablas y las columnas que las componen:

**NAME0** Altura de un bloque, Porcentaje de pixeles negros dentro del bloque (blackpix / area), Porcentaje de pixeles negros después del uso del algoritmo (RLSA) (blackand/area), Número total de pixeles negros en el bitmap del bloque después del RLSA, y Número de transiciones entre blanco y negro en el bitmap original del bloque.

**NAME1** Longitud del bloque, Excentricidad del bloque (lenght/height), y Número total de pixeles negros en el bitmap original del bloque.

**NAME2** Área del bloque (height * lenght), y Número medio de transiciones blanco-negras (blackpix / wbtrans).

**Figura 4.** Script DDL generado para el Modelo Entidad-Relación, ER.

Los modelos Entidad-Relación conceptuales que genera GINY también pueden ser expresados como una Descripción de Contenidos RDF tal como se muestra en la figura 5 para los datos de *Bupa*. A continuación se describen las tres tablas generadas:

**NAME0** Volumen medio corpuscular, Fosfato alcalino, Alamine aminotransferase, y Número de medias-pintas equivalentes a bebidas alcohólicas tomadas por día.

**NAME3** Aspartate aminotransferase.

```
<rdf:RDF xml:lang="en"> <rdfs:Class rdf:ID= " name_0">
<rdfs:comment>Data Table<dfs:comment> <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/classes#Entity"/>
</rdf:Class>

 <rdf:Property ID="nameKey_0">  <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#Integer"/>
</rdf:Property>

 <rdf:Property ID="mcv"> <rdfs:comment>mcv</rdfs:comment>
<rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#real"/>
</rdfs:domain rdf:resource="#name_0"/> </rdf:Property>

 <rdf:Property ID="alkphos"> <rdfs:comment>alkphos</rdfs:comment>
<rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#real"/>
</rdfs:domain rdf:resource="#name_0"/>

</rdf:Property> <rdf:Property ID="sgpt">
<rdfs:comment>sgpt</rdfs:comment> <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#real"/>
</rdfs:domain rdf:resource="#name_0"/> </rdf:Property>

 <rdf:Property ID="drinks">
<rdfs:comment>drinks</rdfs:comment> <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#real"/>
</rdfs:domain rdf:resource="#name_0"/> </rdf:Property>

 <rdf:Property ID="selector">
<rdfs:comment>selector</rdfs:comment> <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#real"/>
</rdfs:domain rdf:resource="#name_0"/></rdf:Property>  </rdf:RDF>

<rdf:RDF xml:lang="en"> <rdfs:Class rdf:ID= " name_3">
<rdfs:comment>Data Table<dfs:comment> <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/classes#Entity"/>
</rdf:Class>  <rdf:Property ID="nameKey_3"> <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#Integer"/>
</rdf:Property>

 <rdf:Property ID="sgot"> <rdfs:comment>sgot</rdfs:comment>
<rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/database#real"/>
</rdfs:domain rdf:resource="#name_3"/> </rdf:Property>  </rdf:RDF>
<rdf:RDF xml:lang="en"> <rdfs:Class rdf:ID= " name_4">
<rdfs:comment>Data Table<dfs:comment> <rdfs:range
rdf:resource="http://www.lsi.upc.es/jgramajo/classes#Entity"/>
</rdf:Class>
```

**Figura 5.** Script generado Descripción de Contenidos, RDF.

**NAME4** Gamma-glutamyl transpeptidase.

En las tablas *NAME3* y *NAME4* quedan separados los datos *sgot* y *gammagt* que son interpretadas con respecto a la tabla *NAME0* como una relación Maestro-Detalle.

## 5 Conclusiones

La información no-estructurada disponible en Internet hace necesario el desarrollo de herramientas como GINY, con la que podemos estructurar información contenida en dominios distribuidos.

El problema de inferir un modelo conceptual de datos se ha resuelto con la definición de un algoritmo de aprendizaje, el cual nos permite generar una estructura de datos a partir de una matriz de datos.

Hasta el momento se pueden inferir modelos conceptuales para dominios con datos de tipo numérico, los resultados se pueden ampliar a datos de tipo alfanumérico.

El proceso de generación de modelos conceptuales se realiza automáticamente con el empleo de técnicas de análisis de datos de Inteligencia Artificial. Se ha propuesto un proceso en el que el modelo conceptual es el resultado de la unión de propiedades que están altamente correlacionadas. El proceso se completa con la obtención de las relaciones el cual se basa en la eliminación de elementos redundantes.

La generación de un script a partir del modelo conceptual Entidad-Relación se hace tanto para obtener diseños de tablas relacionales como para generar Descripción de Contenidos RDF.

## Agradecimientos

Esta investigación ha sido parcialmente financiada por el proyecto h-Techsight, de la Comisión Europea IST-2001-33174.

## Referencias

1. David Riaño and Javier Gramajo. Automatic extraction of data structure. Technical report, Universitat Rovira i Virgili, October 2000.
2. Anil k. Jain and Richard C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall Inc., 1988.
3. *HTML, HyperText Markup Language*. http://www.w3.org/MarkUp/Activity.
4. Judith S. Bowman, Sandra L. Emerson, and Marcy Darnovsky. *The practical SQL Handbook. Using Strcutured Query Language*. Addison Wesley, 3 edition, 1998.
5. Peter P. Chen. The entity-relationship model - toward a unified view of the data. *Transactions on Database Systems*, 1(1):9–36, 1976.
6. *RDF, Resource Description Framework*. http://www.w3.org/TR/rdfs-schema.
7. *XML, The Extensible Markup Language*. http://www.w3.org/XML/.