Base de Datos para el Análisis Morfosintáctico de un Corpus con Anotación Lingüística

Database for morphological and syntactic analysis of an annotated linguistic corpus

Cristina Bender, Claudia Deco, Zulema Solana
Universidad Nacional de Rosario
bender@fceia.unr.edu.ar, deco @fceia.unr.edu.ar, zsolana@arnet.com.ar

Abstract

We present a fragment of the corpus INFO, which consists on words belonging to newspapers between 2003 and 2008. It was prepared by the INFOSUR Research Group of the National University of Rosario.
This work describes the process of tagging, the design of the database, which contains the corpus, the process of database loading and queries that can be made. A morphological analyzer performs the automatic tagging, and disambiguation is done with linguistic information and statistical techniques. The corpus is stored in a relational database, in which every word of text is represented in a row. The table attributes are, the word, the position of the word in text and the tags provided by the analyzer. This design allows extract sequences such as, articles + names, regular verbs ending in-“ir”, or one or more clitics followed by a verb. In addition, it is possible to count the number of occurrences of such constructions in order to obtain statistics of their use.

Keywords: Keyword1, Keyword2, Keyword3, Keyword4, Keyword5.

Resumen

Se presenta un fragmento del corpus INFO, en construcción, de textos pertenecientes a periódicos entre los años 2003 y 2008, preparado por el grupo INFOSUR de la Universidad Nacional de Rosario. En la primera etapa este corpus constará de 100.000 palabras.
En este trabajo, se describen las etapas de etiquetado, el diseño de la base de datos que contiene el corpus, el proceso de carga de la base de datos a partir de los textos etiquetados, y las posibles consultas que se pueden realizar.
El etiquetado se realiza de modo automático con un analizador morfológico, y la desambiguación se efectúa con información lingüística y se complementa con técnicas estadísticas. Para la implementación se opta por una base de datos relacional, en la que cada palabra del texto se representa en una fila cuyos atributos son la posición de la palabra en el texto y las etiquetas provistas por el analizador. Este diseño permite extraer listas de secuencias tales como: nombres antecedidos de artículos, verbos regulares terminados en -ir, o uno o más clíticos seguidos de verbo, entre otros. Además, es posible contar la cantidad de ocurrencias de este tipo de construcciones a fin de poder obtener estadísticas de su utilización.

Palabras claves: Palabra clave 1, Palabra clave 2, Palabra clave 3, Palabra clave 4, Palabra clave 5.

1. INTRODUCCION

En este trabajo se propone producir una base de datos que le permita al usuario consultar categorías gramaticales, contextos sintácticos, frecuencia de ocurrencias de construcciones, uso de la puntuación, tiempos y modos verbales, etc. Para la experimentación, se pobló esta base de datos con palabras de un fragmento del corpus INFO, en construcción, de textos pertenecientes a periódicos entre los años 2003 y 2008, preparado por el grupo de investigación INFOSUR de la Universidad Nacional de Rosario. En la primera etapa, este corpus consta de 100.000 palabras, obtenidas del texto completo de noticias de tipo general, no especializadas, en español.

Se describen las etapas de etiquetado, el diseño de la base de datos que contiene el corpus, el proceso de carga de la base de datos a partir de los textos etiquetados, y algunas posibles consultas que se pueden realizar. El etiquetado se realiza de modo automático con un analizador morfológico, y la desambiguación se efectúa con información lingüística y se complementa con técnicas estadísticas. Para la implementación se opta por una base de datos relacional, en la que cada palabra del texto se representa en una fila cuyos atributos son la posición de la palabra en el texto y las etiquetas provistas por el analizador. Este diseño permite extraer listas de secuencias tales como: nombres antecedidos de artículos, verbos regulares terminados en -ir, uno o más clíticos seguidos de verbo, etc. Además, es posible contar la cantidad de ocurrencias de cada tipo de construcciones a fin de poder obtener estadísticas de su utilización.

2. ETIQUETADO MORFOLÓGICO

Para la carga de la base de datos, en primer lugar se efectúa el análisis del corpus de textos, mediante una herramienta que lo segmenta y etiqueta morfológicamente. En este caso se recurre a SMORPH [1], que tokeniza y efectúa un primer análisis morfológico, sin resolver las ambigüedades. Este tipo de herramientas genera un archivo etiquetado, por ejemplo como el que se muestra en el cuadro 1.

Este archivo es recorrido por un programa que extrae y vuelca esta información a una base de datos, que luego pueda ser consultada y permita realizar los análisis lingüísticos de interés.

A continuación se aclaran las etiquetas utilizadas, que corresponden a rasgos y valores [2].

‘EMS’: etiqueta morfosintáctica.

En el cuadro 1 aparecen las etiquetas morfosintácticas ‘v’ (verbo), ‘nom’ (nombre), ‘adj’ (adjetivo), ‘adv’ (adverbio), ‘prep’ (preposición), ‘det’ (determinante).

En el *verbo*, los rasgos utilizados se pueden clasificar en dos grupos, por un lado, los relacionados con los valores morfológicos de las terminaciones verbales (modo, tiempo, persona, número), por otro lado, los relacionados con la caracterización del tipo de conjugación y con sus aspectos regulares o irregulares.

Primer grupo:

‘MODOV’: tipo de modo (‘ind’ indicativo, ‘subj’ subjuntivo, ‘infin’ infinitivo,

‘imper’ imperativo),

‘TPO’ tipo de tiempo (‘pres’ presente),

‘PERS’ persona (‘1a’ primera, ‘2a’ segunda, ‘3a’ tercera),

‘NUM’ número (‘sg’ singular, ‘pl’ plural).

Segundo grupo:

‘TC’ : tipo de conjugación (‘c1’ primera, ‘c2’ segunda, ‘c3’ tercera), .

‘TR’: rasgo que indica si el verbo es o no regular (‘r’ regular, ‘irr’ irregular).

‘TIRR’: tipo de irregularidad (‘hiper’ hiperirregular).

En el *determinante*, se dan dos tipos: ‘art’ para los definidos e ‘indf’ para los indefinidos.

'Desde’. ['desde', 'EMS','prep'].

'ya,’. ['ya’, 'EMS', ‘adv’].

'hacer’. ['hacer', 'EMS','v', 'TC', 'c2', 'MODOV','infin', 'TR','ir', 'TIRR','hiper'].

'caer’. ['caer', 'EMS','v', 'TC','c2', 'MODOV','infin', 'TR','ir', 'TIRR','hiper'].

'una’. ['una', ‘EMS’,’det’, 'TDET','indf', 'GEN','fem', 'NUM','sg'].

['unir', 'EMS','v', 'TC', 'c3', 'MODOV','subj', 'TPO','pres', 'PERS','1a', 'NUM','sg'].

['unir', 'EMS','v', 'TC', 'c3', 'MODOV','subj', 'TPO','pres', 'PERS','3a', 'NUM','sg'].

['unir', 'EMS','v', 'TC', 'c3', 'MODOV','imper', 'TPO','pres', 'PERS','3a', 'NUM','sg'].

'privatización’.

['privatización', 'EMS','nom', 'GEN','fem', 'NUM','sg'].

'no’. ['no', ‘EMS’ ‘adv’].

'es’. ['ser', 'EMS','v', 'TC','c2', 'MODOV','ind','TPO','pres', 'PERS','3a', 'NUM','sg''TR','ir', 'TIRR','hiper'].

'un’. ['un', ‘EMS’,’det’, 'TDET','indf', 'GEN','masc', 'NUM','sg'].

'dato’. ['dato', 'EMS','nom', 'GEN','masc', 'NUM','sg'].

['datar', 'EMS','v', 'TC', 'c1', 'MODOV','indic', 'TPO', 'pres', 'PERS','1a', 'NUM','sg'].

'menor’. ['menor', 'EMS','adj', 'GEN','_', 'NUM','sg'].

'.’. ['.', 'EMS','ponc'].

Cuadro 1: Ejemplo de salida del analizador morfológico

3. DISEÑO DE LA BASE DE DATOS RELACIONAL

En la teoría de bases de datos existen diversos modelos que pueden utilizarse para representar, almacenar y recuperar la información. La propuesta de este trabajo es utilizar el modelo relacional de bases de datos, aprovechando la versatilidad que brinda este modelo en las implementaciones de las consultas a través del lenguaje de consulta estructurado SQL (Structured Query Language) [3]. En este modelo, la información se representa en forma de tablas, donde cada fila corresponde a un elemento dado (en nuestro caso una acepción de una palabra), y cada columna a un atributo descriptivo de la misma [4].

Como se observa en el cuadro 1, una palabra extraída del texto puede tener una o más etiquetas, como en el caso de "dato" que puede ser un nombre o un verbo conjugado. Por lo tanto en la base de datos se almacenan cada palabra en una o más filas, dependiendo de la cantidad de etiquetas morfosintácticas que posea. Es decir, para la palabra "dato" se tienen dos filas en la tabla**.**

Otro elemento a tener en cuenta en el diseño de la base de datos es el almacenamiento de la ubicación de cada término dentro del texto a analizar. Para esto se guarda información posicional que consiste en: número o identificación del texto (noticia) en la que se encuentra la palabra, número de oración en la que está la palabra y posición de la palabra dentro de la oración. La información posicional permite realizar consultas por adyacencia, tal como encontrar construcciones del tipo "nombre" antecedido de "artículo".

De esta forma, se propone el siguiente diseño de la base de datos *Corpus*:

Corpus(Palabra, NroTexto, NroOración, PosiciónEnOración,
EMS, TC, MODOV, TR, TIRR, TPO, PERS, NUM, GEN, ...)

cuyos atributos son:

Palabra: contiene la palabra como aparece en el texto.

NroTexto: contiene el número o identificación del texto donde se encuentra la palabra.

NroOración: corresponde al número de oración dentro del texto donde se encuentra la palabra.

PosiciónEnOración: es un número que representa la posición de la palabra dentro de la oración.

EMS, TC, MODOV, TR, TIRR, TPO, PERS, NUM, GEN, ...: contienen los valores de las etiquetas correspondientes.

Por ejemplo, para el texto del cuadro 1, la instancia correspondiente de la base de datos Corpus, es la presentada en el cuadro 2.

Palabra	NroTexto	NroOración	PosiciónEn Oración	EMS	TC	MODOV	TR	TIRR	TPO	PERS	GEN	NUM	TDET	Origen
desde	1	1	1	prep										
ya	1	1	2	adv										
hacer	1	1	3	v	c2	infin	ir	hiper						
caer	1	1	4	v	c2	infin	ir	hiper						
una	1	1	5	det							fem	sg	indf	
una	1	1	5	v	c3	subj			pres	1a		sg		unir
una	1	1	5	v	c3	subj			pres	3a		sg		unir
una	1	1	5	v	c3	imper			pres	3a		sg		unir
privatización	1	1	6	nom							fem	sg		
no	1	1	7	adv										
es	1	1	8	v	c2	ind	ir	hiper	pres	3a		sg		
un	1	1	9	det							masc	sg	indf	
dato	1	1	10	nom							masc	sg		
dato	1	1	10	v	c1	ind			pres	1a		sg		datar
menor	1	1	11	adj							_	sg		
.	1	1	12	ponc										

Cuadro 2: Instancia de la base de datos Corpus

El algoritmo para la carga de la base de datos es:

Tomar una noticia del corpus de textos
Generar el archivo etiquetado mediante una herramienta de análisis morfológico.
Procesar este archivo etiquetado para volcarlo a la base de datos
Leer una línea del archivo de salida etiquetado.
Si esta línea comienza con una palabra
Entonces Insertar una fila en la tabla con la palabra, la información posicional y los valores de las etiquetas correspondientes
Si la línea no comienza con una palabra [1]
Entonces Insertar una fila en la tabla con la palabra de la fila anterior, la nueva información posicional y los nuevos valores de las etiquetas correspondientes
Continuar mientras haya líneas en el archivo
Continuar mientras haya noticias en el corpus de textos
Fin.

El diseño propuesto para la base de datos Corpus, permite resolver, por ejemplo, consultas del siguiente tipo:

[1] En el archivo etiquetado del cuadro 1, hay líneas que no comienzan con una palabra sino con espacios en blanco, como es el caso de las tres líneas siguientes a la de la palabra *una*. Esto ocurre cuando una palabra tiene más de un etiquetado.

- Encontrar listas de secuencias de nombres antecedidos de artículos.
- Encontrar listas de secuencias de nombres antecedidos de artículos, con uno o más adjetivos entre ellos.
- Encontrar listas de secuencias de nombres antecedidos de artículos, con uno o más adjetivos y adverbios entre ellos.
- Encontrar los verbos regulares terminados en -ir.
- Encontrar uno o más clíticos seguidos de verbo.
- Encontrar uno o más clíticos seguidos de verbo conjugado y verbo en infinitivo.
- Contar la cantidad de ocurrencias de cualquiera de estas construcciones.

En el modelo relacional, una consulta se expresa en el lenguaje SQL. Una sentencia de consulta SQL tiene la siguiente sintaxis:

SELECT atributos FROM tabla WHERE condición;

donde

atributos: es la lista de columnas que se desea ver en la respuesta.

tabla: es el nombre de la tabla que contiene los datos, en nuestro caso Corpus.

condición: es un predicado que contiene operadores lógicos Y, O y NO.

4. CASO DE USO

Consideremos el siguiente texto que corresponde a las tres primeras oraciones de una noticia publicada el 25 de noviembre de 2003 en el periódico PAGINA 12.

> *"Mucho más por lo que sugiere como dibujo de las grandes líneas económicas gubernamentales que por el hecho y los involucrados en sí mismos, el quite de la concesión del Correo al grupo Macri y, esencialmente, la reprivatización de la empresa, son la gran noticia de los últimos tiempos. La decisión estaba tomada desde antes de la segunda vuelta electoral porteña y si no se comunicó hasta ahora fue porque, con buen cálculo, el Gobierno estimó que empalmarla con la derrota del presidente de Boca a manos de Aníbal Ibarra hubiera sido de muy mal gusto. Y siendo que eso fue así llama la atención que el grupo Macri no se mostrara más activo en la defensa de su gestión, con lo cual queda avalada la sospecha de que no le interesaba seguirla. Desde ya, hacer caer una privatización no es un dato menor."*

Este texto es analizado con una herramienta que lo segmenta y etiqueta morfológicamente, obteniendo una salida similar a la mostrada en el cuadro 1. Esta salida es luego cargada a la base de datos relacional Corpus.

A continuación se presentan algunas consultas SQL de ejemplo.

Ejemplo 1:

Para encontrar los verbos regulares terminados en -ir, la consulta en SQL es la siguiente:

```
SELECT      Palabra
      FROM          Corpus
      WHERE         EMS = 'v'
                                  AND  TC = 'c3';
```

Obteniéndose el siguiente resultado para el caso de uso que se está analizando:

sugiere

una

Ejemplo 2:

Si se desea por ejemplo, encontrar la lista de secuencias de nombres antecedidos de artículos determinantes, la consulta en SQL es la siguiente:

```
SELECT   A.Palabra, N.Palabra
      FROM  Corpus  A, Corpus N
      WHERE         A.NroTexto = N.NroTexto
                                  AND A.NroOración = N.NroOración
                                  AND N.PosiciónOración - A.PosiciónOración = 1
                                  AND A.EMS = 'det'
                                  AND A.TDET = 'art'
                                  AND N.EMS = 'nom' ;
```

Esta sentencia muestra una lista de artículos determinantes seguidos por nombres. En la condición de búsqueda se pide que las dos palabras estén en el mismo texto, la misma oración, que la resta de sus posiciones dé 1 y que la primera sea un artículo definido y la segunda sea un nombre. Para el texto de ejemplo el resultado de esta consulta es:

la concesión

la reprivatización

la empresa

la decisión

el Gobierno

la derrota

la atención

el grupo

la defensa

la sospecha

Ejemplo 3:

Volviendo al Ejemplo 1, el resultado de la consulta es erróneo, porque en el texto bajo análisis la palabra *una* es un indefinido que acompaña al sustantivo *privatización*. Esto se debe a que pueden existir palabras a las que se les asigne más de una categoría en la etapa de análisis morfosintáctico, como es el caso de la palabra *una*. Esto produce que en la instancia de la base de datos aparezca más de una fila para estas palabras. Una posibilidad para resolver este problema, es agregar condiciones. Para nuestro ejemplo, deberían descartarse aquellas construcciones del tipo artículo indefinido + nombre.

Esto puede realizarse en SQL con una consulta como la siguiente:

```
SELECT Palabra
    FROM Corpus
    WHERE       EMS = 'v'
                              AND  TC = 'c3'
                              AND Palabra not in (SELECT A.Palabra

    FROM  Corpus  A, Corpus N

    WHERE A.NroTexto = N.NroTexto

        AND A.NroOración = N.NroOración

        AND N.PosiciónOración-A.PosiciónOración=1

        AND A.TDET= 'indf'

        AND N.EMS = 'nom');
```

Ejemplo 4:

Para encontrar construcciones del tipo artículo definido seguido de adjetivo y luego de nombre, la

consulta en SQL es la siguiente:

```
SELECT   ART.Palabra, ADJ.Palabra, N.Palabra
      FROM  Corpus  ART, Corpus ADJ, Corpus N
      WHERE       ART.NroTexto = ADJ.NroTexto AND ADJ.NroTexto = N.NroTexto
                                 AND ART.NroOración = ADJ.NroOración
                                 AND ADJ.NroOracion = N.NroOracion
                                 AND ART.PosiciónOración - N.PosiciónOración = 2
                                 AND  ART.PosiciónOración  -  ADJ.PosiciónOración =
1
                                 AND ART.TDET = ‘art’
                                 AND ADJ.EMS = ‘adj’
                                 AND N.EMS = ‘nom’;
```

Obteniéndose el siguiente resultado para el caso de uso que se está analizando:

las grandes líneas

la gran noticia

los últimos tiempos

la segunda vuelta

Estos ejemplos de consultas presentados, permiten observar la versatilidad del lenguaje de consulta estructurado SQL y del modelo relacional para encontrar solución a los distintos tipos de problemas.

5. CONCLUSIONES

En este trabajo se propuso la utilización de una base de datos relacional que permita efectuar análisis sobre construcciones morfosintácticas utilizadas en el idioma español. Para esto, se presentó su diseño, el algoritmo de carga y el uso del lenguaje de consulta SQL para recuperar información. Los casos de uso presentados se ejecutaron sobre una instancia de la base de datos Corpus generada a partir de un fragmento del corpus INFO de textos pertenecientes a periódicos entre los años 2003 y 2008, preparado por el grupo INFOSUR de la Universidad Nacional de Rosario.

Esta propuesta es independiente del analizador morfológico que se utilice. La versatilidad de una base de datos relacional ofrece la ventaja de que con una consulta escrita en SQL es posible recuperar los datos de la forma requerida en cada caso. Es decir, las bases de datos relacionales tienen la capacidad de adaptarse con facilidad y rapidez a diversas funciones. Esto brinda una amplia gama de posibilidades para que los lingüistas puedan analizar diversas construcciones del idioma español mediante la preparación de una consulta adecuada.

Referencias

[1] Aït-Mokhtar, Salah 1998. *L'analyse présyntaxique en une seule étape*. Tesis doctoral. Universidad Blaise-Pascal/GRIL, Clermont-Ferrand.

[2] Solana, Z. y equipo INFOSUR 2006. *Morfología del verbo español*, Centro de Estudios de Adquisición del Lenguaje, Facultad de Humanidades y Artes, UNR.

[3] Date, C.J.; H. Darwen 1997. *A guide to the SQL Standard*. Ed. Addison-Wesley.

[4] Silberschatz, A., H. F. Korth 2003. *Fundamentos de Bases de Datos*, 3 edición, Ed. McGraw-Hill.

[5] Bés, Gabriel; Zulema Solana; Celina Beltrán 2005. "Conocimiento de la lengua y técnicas estadísticas en el análisis lingüístico". En Víctor Castel (ed.) *Desarrollo, implementación uso de modelos para el procesamiento automático de textos*. Facultad de Filosofía y Letras, UNCUYO.