

Un sistema para la segmentación en proposiciones de textos en español

Dina Wonsever*

Serrana Caviglia

Aiala Rosá

Javier Couto

Universidad de la República – Uruguay



Resumen: Se describe un sistema dedicado al reconocimiento de proposiciones y de las relaciones jerárquicas entre ellas, para textos en español. Dicho sistema fue concebido para brindar un servicio de base a tareas de extracción de información de texto y de análisis sintáctico. El sistema opera a partir de los resultados de un etiquetador probabilístico. Los conocimientos lingüísticos se expresan en un formalismo de reglas de reescritura, con reglas que permiten una especificación parcial de la parte de reescritura y tomar en cuenta el texto que rodea esta parte (contexto). La descripción se centra en la organización de las reglas en módulos que atacan distintos tipos de problemas lingüísticos, mostrando las sucesivas etapas del análisis completo de una oración ejemplo.

1 Introducción

En este artículo presentamos un sistema de segmentación en proposiciones y de cálculo de la estructura de una oración en términos de las proposiciones que contiene. Dicho sistema fue concebido para suministrar un servicio de base a otras tareas de análisis textual. En particular, diversos sistemas de filtrado de información por Exploración Contextual ([DES 97a, 97b]) necesitan un reconocimiento previo de las proposiciones para poder determinar ya sea el valor de una ocurrencia del imperfecto ([MAI 90]), o los argumentos de

* wonsever@fing.edu.uy

relaciones de causalidad ([GAR 98] u otros fenómenos lingüísticos. Considerando que una proposición tal como fue definida en el ámbito de este trabajo es una buena aproximación a la noción de espacio (segmento) textual donde coocurren los predicados y sus argumentos y modificadores, la detección de las proposiciones en una oración larga y compleja ofrece un espacio de búsqueda coherente a sistemas de extracción de información en general.

El conjunto de proposiciones recuperadas y sus relaciones estructurales en la oración constituyen, por otra parte, un punto de partida interesante para un analizador sintáctico. Este tendría como entrada unidades de menor tamaño que la oración entera, lo que es muy importante en textos reales que suelen tener oraciones muy largas (p.ej., textos periodísticos). En el proceso de construcción de las proposiciones del texto, así como en el cálculo de las relaciones estructurales, se realiza una toma de decisiones que toma en cuenta todo el material existente en la oración y resuelve potenciales conflictos. Se lograría de este modo reducir la explosión combinatoria generada por analizadores que mantienen todos los análisis posibles.

El sistema realizado es de tipo procesamiento simbólico, aunque utiliza los resultados de un etiquetador probabilístico. Se basa en la descripción, mediante reglas, de los elementos centrales para la determinación de las proposiciones (verbos, conjunciones, signos de puntuación) y su estructuración en la oración. El formalismo utilizado, llamado *Reglas Contextuales*, permite la descripción parcial de segmentos de texto considerando el contexto (texto que rodea el segmento). En este artículo se describen pautas generales en el diseño de las reglas y se muestra el funcionamiento de algunas reglas a partir del análisis completo de una oración ejemplo (sección 4). Previamente, en la sección 2 se presentan brevemente la noción de proposición utilizada en este trabajo y el tipo de resultados que se aspira obtener. En la sección 3 se describe el formalismo de reglas contextuales, con el que se representaron los conocimientos lingüísticos en el sistema.

Los resultados de una validación cuantitativa que se realizó sobre un conjunto de textos se presentan en la sección 5 y una discusión final sobre los errores, puntos de especial dificultad y refinamientos posibles en la sección 6.

2 Noción de proposición y enunciado

La noción de proposición que se utiliza en este trabajo es sintáctica y “concreta”: se considera proposición a un segmento de texto que contiene un verbo conjugado, sus argumentos y modificadores. Si bien la idea de proposición está vinculada a una noción

más amplia de predicación, es claro que la predicación no se realiza solamente por medio de expresiones que giran en torno a un verbo conjugado, así como no siempre son predicativas este tipo de expresiones. Es por esto que decimos que utilizamos una noción sintáctica de proposición. El término “concreta” refiere a que se aproxima la noción gramatical abstracta de proposición por una realización material en términos de segmentos de texto. Esto tiene como consecuencia que en muchos casos todos los argumentos y/o modificadores no se encuentren efectivamente realizados en el segmento de texto asociado a una proposición; esto ocurre por ejemplo en casos de coordinadas, donde material asociado a las dos proposiciones que coordinan puede aparecer en la primera o en la segunda.

En (1) se muestra el resultado del análisis que realiza nuestro sistema sobre el siguiente texto de entrada:

Cuando se editó hace muy poco un disco compacto de Rubén Rada en Buenos Aires, con materiales grabados en 1987, no se consultó a Rada ni a sus colaboradores ni se les pagó por la edición, que por otra parte fue muy grande por lo que se tuvo que vender a precio muy bajo, compitiendo en el mercado con su disco recién editado en Estados Unidos.

- (1) [enunciado en2/
 [prop gc21/
 [prop gf22/
 [prop ge02/ Cuando se editó hace muy poco
 un disco compacto de Ruben Rada en Bue-
 nos Aires, con materiales grabados en 1987,
 /prop ge02],
 no se consultó a Rada ni a sus colaboradores
 /prop gf22],
 ni
 [prop gf21/ se les pagó por la edición,
 [prop gf16/ que por otra parte fue muy grande
 [prop ge02/ por lo que se tuvo que
 vender a precio muy bajo, compitiendo en el mercado con su disco recién editado en Estados Unidos
 /prop ge02]
 /prop gf16]
 /prop gf21]
 /prop gc21]
 /enunciado en2]

El texto en **negrita** corresponde a las marcas de proposición generadas por el sistema. Hay un par de marcas (apertura y cierre) por cada proposición (etiqueta *prop*). En las marcas que se muestran en el ejemplo aparece además el código de la regla utilizada para generar la proposición. Comentando sobre el ejemplo, podemos decir que se reconocieron 6 proposiciones en la oración anterior, detalladas como sigue:

Prop1 – cuando se editó hace muy poco un disco compacto de Ruben Rada en Buenos Aires, con materiales grabados en 1987	código ge02
Prop2 – <prop1> no se consultó a Rada ni a sus colaboradores	código gf22
Prop3 – por lo que que se tuvo que vender a precio muy bajo, compitiendo en el mercado con su disco recién editado en Estados Unidos	código ge02
Prop4 – que por otra parte fue muy grande	código gf16
< prop3	
Prop5 – se les pagó por la edición, <prop4>	código gf21
Prop6 – <prop2>, ni <prop5>	

Las indentaciones en la despliegue de la oración analizada señalan de algún modo relaciones de subordinación o dependencias en el interior de la oración. Notar que *Prop1* podría verse como un modificador de todo el material restante (o como la primera parte de una construcción bipolar), este tipo de análisis no se ha realizado aún.

Las marcas *enunciado* señalan comienzo y fin de oración de texto. Consideramos que la puntuación es una indicación explícita de la estructuración en unidades en textos escritos [NUN90]. La oración de texto queda determinada por el par de delimitadores (*mayúscula inicial*, *punto de fin de oración*) y puede contener en su interior categorías textuales indicadas por los signos ‘:’ y ‘;’, considerados a veces como límite de oración en sistemas de procesamiento de lenguaje natural.

3 El formalismo de reglas contextuales

La descripción de las proposiciones se realiza mediante un formalismo diseñado a esos efectos y cuyas expresiones básicas hemos denominado *Reglas Contextuales*. Operando sobre un conjunto predefinido de etiquetas, las Reglas Contextuales definen operaciones de reescritura de secuencias de etiquetas y zonas que ocurren en determinado contexto.

Mostramos los elementos sintácticos y el funcionamiento de una regla contextual analizando uno de los ejemplos asociados a las expresiones “iniciadoras” de proposición. Para una descripción completa del formalismo véase [WON01].

- (2) q3b: iniProp(0) → tener *(S,4) \ que / uvInf; S={uniVerb, que, iniProp, punt}
 q3b – código de la regla
 iniProp – etiqueta que indica iniciador de proposición
 tener – etiqueta que indica un verbo con lema *tener*
 que – etiqueta que indica una ocurrencia de la palabra *que*
 uvInf – etiqueta que indica un verbo en infinitivo
 uniVerb – etiqueta que indica una Unidad Verbal (verbo + eventuales clíticos)
 punt – etiqueta que indica un signo de puntuación

El símbolo que precede a la flecha es la etiqueta que se agrega en caso de que se satisfaga la regla. En el ejemplo, iniProp. El argumento 0 refiere a un subtipo de iniProp; indica en el ejemplo que esta ocurrencia de *que* no corresponde de hecho a un iniciador de proposición.

La condición de satisfacción de la regla aparece a la derecha de la flecha. Podemos distinguir 3 partes, delimitadas por ‘\’ y ‘/’. Estas son: el contexto izquierdo (tener *(S,4)), el cuerpo (que) y el contexto derecho (uvInf). El cuerpo es la parte que el símbolo del lado izquierdo reescribe. Los contextos izquierdo y derecho expresan condiciones adicionales para que esa reescritura pueda tener lugar.

Por medio de la expresión *(S,4), siendo S={uniVerb, que, iniProp, punt} (llamamos *zona* a expresiones de este tipo) especificamos una porción de texto de largo a lo sumo 4 donde no debe ocurrir ninguna de las categorías definidas en el conjunto S.

Resumiendo: la regla q3b tiene como objetivo excluir del conjunto de indicadores de inicio de proposición a ocurrencias de *que* que están dentro de perífrasis verbales de la forma *tener + que + infinitivo*. Esta perífrasis es igualmente reconocida si se inserta algún elemento entre la forma del verbo *tener* y *que* (ej. *Tengo necesariamente que hacerlo*).

Se ha desarrollado un sistema que incluye un intérprete para las reglas contextuales. Este intérprete opera sobre grupos de reglas (llamados *módulos*) y puede hacerlo en distintas modalidades que se especifican en su invocación. El intérprete está incluido en un sistema que maneja una agenda de tareas (especificadas en su mayor parte como conjuntos de reglas contextuales) cuyo resulta-

do final es el reconocimiento de las proposiciones. Está implementado en Sicstus Prolog y existe también una versión experimental que utiliza el módulo CHR (*Constraint Handling Rules*) para Programación Lógica con Restricciones [GAR02].

4 Módulos de Reglas

En el contexto de nuestro sistema, un *módulo* es un conjunto de reglas contextuales interrelacionadas que se ejecutan simultáneamente.

La aplicación sobre el texto de los módulos de reglas se realiza “en cascada”. Cada paso toma como entrada la salida de la etapa anterior.

El diseño de módulos se hizo bajo el principio de tratar un problema por vez. Conceptualmente podemos distinguir entre módulos que toman decisiones locales (195 reglas) y el módulo final que toma decisiones globales (89 reglas). Las decisiones locales corresponden a la desambiguación contextual de elementos polisémicos (conjunciones fundamentalmente), a la integración de verbos con elementos muy próximos (clíticos, negación) y al reconocimiento (parcial) de series e incisos. También incluimos dentro de los módulos locales los que arman el núcleo de las proposiciones.

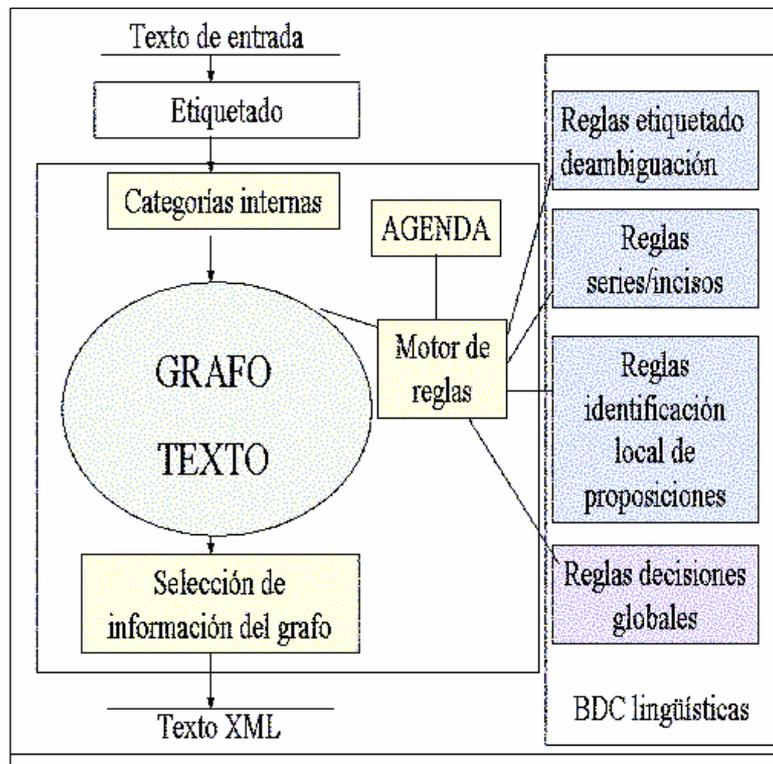
Las decisiones globales, por su parte, realizan el cálculo definitivo de las proposiciones y de las relaciones estructurales entre las mismas.

En el cuadro de la página siguiente se muestran las componentes del sistema y los distintos módulos de reglas (bases de conocimiento) utilizadas.

El sistema desarrollado toma como entrada textos preprocesados por un etiquetador / desambiguador probabilística.¹

Las categorías asignadas por el etiquetador se traducen a categorías internas del sistema; se pretende de este modo conservar una relativa independencia respecto del etiquetador que se utilice. En la mayor parte de los casos la traducción genera categorías iniciales. En otros casos, se asignan directamente categorías funcionales. Esto es posible para las categorías funcionales que no dependen del contexto y por lo tanto no requieren de la aplicación de reglas contextuales.

¹ Se utilizó Xelda (Xerox Linguistic Development Architecture), desarrollado por XRCE, Xerox.



Las etiquetas son objetos estructurados: una etiqueta es un conjunto de pares atributo-valor. Las reglas o procesos que utilizan las etiquetas no necesitan especificar los valores de todos los atributos, basta con que seleccionen algún/os atributo/s. No todas las etiquetas tienen el mismo conjunto de atributos: podemos hablar de subtipos.

Las etiquetas iniciales están asociadas a descripciones relativamente estandarizadas de categorías morfosintácticas y sus rasgos típicos. Las etiquetas funcionales están vinculadas a la operatoria del sistema. En este sentido, pueden agrupar elementos que para un análisis tradicional pertenecen a categorías gramaticales distintas.

En lo que sigue describiremos brevemente los distintos módulos de reglas e ilustraremos paso a paso su funcionamiento al aplicarlos sobre una oración ejemplo. La salida de cada módulo representa el análisis estructural necesario para la aplicación de las reglas del módulo siguiente.

De esta manera, la salida de cada módulo permitirá apreciar los sucesivos análisis estructurales por los que debe pasar un texto antes de alcanzar el análisis estructural final representado en (1).

Para que la visualización de las salidas de los sucesivos módulos del sistema resulte más clara, se muestran solamente las etiquetas asignadas por el módulo que se está presentando y se ocultan las asignadas por módulos anteriores. A su vez, para cada etiqueta no se despliegan todos los atributos existentes sino que se selecciona el subconjunto de atributos y valores relevantes a los efectos de la exposición.

4.1 Reglas locales

Módulo 1: unidades verbales

La marca de verbo del procesador morfosintáctico es traducida a la etiqueta interna *uniVerb*. Las reglas contextuales de este módulo extenderán la etiqueta *uniVerb* de manera de abarcar los clíticos, el adverbio de negación y el sujeto pronominal precedentes. Esto es, aquellos elementos que indudablemente van a pertenecer a la proposición que ese verbo conjugado nuclea. De esta manera se busca asegurar el límite izquierdo de las proposiciones.

Salida del módulo 1

- (3) Cuando [uniVerb v3/ **se** [uniVerb lex F=fin/ **editó** /uniVerb lex] /uniVerb v3] [uniVerb lex F=fin/ **hace** /uniVerb lex] muy poco un disco compacto de Ruben Rada en Buenos Aires, con materiales grabados en 1987, [uniVerb v1/ **no** [uniVerb v3/ **se** [uniVerb lex F=fin/ **consultó**, /uniVerb lex] /uniVerb v3] /uniVerb v1] a Rada ni a sus colaboradores, ni [uniVerb v3/ **se** [uniVerb v3/ **les** [uniVerb lex F=fin/ **pagó** /uniVerb lex] /uniVerb v3] /uniVerb v3] por la edición, que por otra parte [uniVerb lex F=fin/ **fue** /uniVerb lex] muy grande por lo que [uniVerb v3/ **se** [uniVerb lex F=fin / **tuvo** /uniVerb lex] /uniVerb v3] que [uniVerb lex F=noFin/ **vender** /uniVerb lex] a precio muy bajo, [uniVerb lex F=noFin/ **compitiendo** /uniVerb lex] en el mercado con su disco recién [uniVerb lex F=noFin/ **editado** /uniVerb lex] en Estados Unidos.

La salida del módulo 1 nos permite observar que, tal como advertimos anteriormente, bajo las etiquetas de este sistema quedan agrupadas distintas categorías gramaticales, por ejemplo, [uniVerb v1/ **no** [uniVerb v3/ **se** [uniVerb lex F=fin/ **consultó**, /uniVerb lex] /uniVerb v3] /uniVerb v1].

Ahora bien, dado que el sistema considera proposición sólo a aquellas predicaciones nucleadas por un verbo conjugado, las reglas se aplican sólo sobre un subtipo de los segmentos etiquetados como *uniVerb lex*, a saber, aquellos cuyo atributo *Forma* tiene el valor *finito*, dejando de lado las formas no finitas del verbo. En negrita resaltamos los verbos que dan cuenta de tal subtipo y constituyen núcleo de proposición. Las etiquetas *uniVerb* con código de regla extienden estos núcleos, por ejemplo, la regla v3 agrega el clítico formando *se consultó* y la regla v1 agrega la negación formando *no se consultó*.

En este módulo quedan asignados seis núcleos de proposición. También quedan propuestos seis inicios de proposición, ambos hechos se destacan con las negritas. Esta primera aproximación a la estructura del texto va a ir siendo corregida por la aplicación de los siguientes módulos de reglas.

Módulo 2: iniciadores de proposición

La etiqueta *iniProp* agrupa a toda unidad léxica capaz de introducir una proposición subordinada. Esta función recae básicamente en el conocido grupo de los pronombres y adverbios relativos e interrogativos, las conjunciones y adverbios subordinantes y las locuciones conjuntivas.

Al listar estas partículas vemos que muchas de ellas actúan siempre como elementos iniciadores de proposición (*porque, ya que, quien, donde* etc.) pero otras son ambiguas ya que no siempre están usadas en su función subordinadora (*que, como, ni bien, etc.*) Por ejemplo, la forma *que* es subordinadora en sus usos mayoritarios pero cuando está usada como introductora de la coda comparativa (*más alto que Juan*) y cuando integra algunas perífrasis verbales (*haber/tener+ que+ infinitivo*) no lo es. La desambiguación de estas unidades es el cometido específico de las reglas de este módulo. En efecto, es el conjunto de reglas de este módulo el que define los contextos que permiten decidir si estas unidades están actuando o no como subordinadores [CAV 03].

El grado de certeza con que un elemento portador de la etiqueta *iniProp* será efectivamente un iniciador de proposición se especifica a través del valor del atributo *Peso*. Por lo tanto, si una forma queda definida como *iniProp P=1*, subordinará al verbo conjugado que le sigue; si queda definida como *iniProp P=2* lo subordinará solo si entre ella y el verbo no se interpone una coma; si el valor que se le atribuye es 0, entonces no va a establecer ninguna relación de subordinación.

El módulo 2 contiene también reglas que extienden la etiqueta *iniProp* de manera de incorporar los artículos definidos y las preposiciones que aparezcan a la izquierda.

En (4) mostramos exclusivamente las etiquetas generadas por el módulo 2 ocultando las generadas por el módulo 1 para claridad de la exposición.

- (4) [*iniProp* P=1/ **cuando** /*iniProp*] se editó hace muy poco un disco compacto de Ruben Rada en Buenos Aires, con materiales grabados en 1987, no se consultó a Rada ni a sus colaboradores, ni se les pagó por la edición, [*iniProp* P=1/ **que** /*iniProp*] por otra parte fue muy grande [*iniProp* P=1/ **por lo que** /*iniProp*] se tuvo [*iniProp* P=0/ **que** /*iniProp*] vender a precio muy bajo, compitiendo en el mercado con su disco recién editado en Estados Unidos.

Este módulo asignó cuatro etiquetas *iniProp* (obsérvense las palabras en negrita), tres de ellas con *Peso*=1 y una con *Peso*=0, por lo que debemos esperar tres oraciones subordinadas.

Módulo 3: incisos

El reconocimiento de los incisos es necesario por distintas razones. En los casos de los incisos con separación más fuerte (entre paréntesis, entre guiones) es necesario para impedir toda interacción entre elementos interiores y exteriores al mismo. En el caso de los incisos indicados por coma, hay al menos 2 objetivos: el reconocimiento de los incisos de discurso forma parte de los objetivos del sistema, en la medida en que al contener verbos conjugados son considerados proposiciones (estos incisos aparecen bajo la categoría principal *prop*). El reconocimiento de los otros incisos indicados por coma permite categorizar algunas de las comas que los rodean y dejar de considerarlas como posibles límites de proposiciones.

Los incisos de discurso se distinguen por contener un verbo conjugado perteneciente al grupo de verbos de discurso o creencia. Para cada *uniVerb*, la pertenencia a este grupo está indicada por el valor *sí* en el atributo *Discurso* de la etiqueta. Como muestra el ejemplo *1inc*, el inciso de discurso queda delimitado por las comas. En otros casos se encuentran al final de la oración y quedan delimitados por el punto.

Algunas reglas de este módulo, permiten reconocer ciertas conformaciones contextuales donde la forma *hace* (*hace dos años*, *hace tiempo*) etiquetada como *uniVerb* en su momento, no está ejerciendo como núcleo de proposición. Es así que se la sobreetiqueta como inciso y se evitan segmentaciones equivocadas.

- (5) cuando se editó [**inciso it3/** hace muy poco /**inciso it3**] un disco compacto de Ruben Rada en Buenos Aires, con materiales grabados en 1987, no se consultó a Rada ni a sus colaboradores, ni se les pagó por la edición, que por otra parte fue muy grande por lo que se tuvo que a precio muy bajo, compitiendo en el mercado con su disco recién editado en Estados Unidos.

En el ejemplo, la etiqueta que engloba el inciso, oculta la ocurrencia del verbo *hace* para las etapas posteriores, por lo que, ahora, habrá 5 posibles núcleos de proposición.

Módulo 4: series

El cálculo de series es un subproducto del sistema. A los efectos del cálculo de proposiciones, el efecto importante es dejar “cubiertos” bajo el arco de serie algunos signos de puntuación y/o conjunciones que podrían ser candidatos a ejercer como separadores de proposiciones.

Es evidente que este módulo cumple también con una finalidad de desambiguación, en la medida en que permite identificar y volver invisibles aquellas comas y conjunciones que no juegan ningún rol en los pasos subsiguientes: la delimitación de proposiciones y su estructuración jerárquica.

- (6) cuando se editó hace muy poco un disco compacto de Ruben Rada en Buenos Aires, con materiales grabados en 1987, no se consultó [**serie nipv1/** a Rada ni a sus colaboradores /**serie nipv1**], ni se les pagó por la edición, que por otra parte fue muy grande por lo que se tuvo que vender a precio muy bajo, compitiendo en el mercado con su disco recién editado en Estados Unidos.

En el ejemplo (6), de 2 ocurrencias de la conjunción *ni*, solamente una será considerada como posible separador de proposiciones en etapas posteriores del proceso de segmentación.

Módulo 5: proposiciones locales y segmentos

En esta etapa se construyen partes “seguras” de proposiciones, uniendo los indicadores de inicio de proposición con verbos (proposiciones insertadas) o extendiendo hasta determinadas barreras (signos de puntuación, conjunciones de coordinación) los segmentos que contienen verbos (proposiciones independientes). Quedan determinados asimismo segmentos (entre signos de puntuación y conjunciones) no proposicionales.

- (7) [prop prl3/ cuando se editó hace muy poco un disco compacto de Rubén Rada en Buenos Aires /prop prl3],
[seg s6/ con materiales grabados en 1987 /seg s6],
[prop pli4/ no se consultó a Rada ni a sus colaboradores /prop pli4],
ni
[prop pli4/ se les pagó por la edición /prop pli4],
[prop prl5/ que por otra parte fue muy grande /prop prl5]
[prop prl3/ por lo que se tuvo que vender a precio muy bajo /prop prl3],
[seg s2/ compitiendo en el mercado con su disco recién editado en Estados Unidos /seg s2].

4.2 Reglas globales: módulo 6

Las reglas globales constituyen el último módulo (en secuencia de ejecución) del reconocimiento de proposiciones. Sobre los segmentos, proposicionales o no, reconocidos en etapas anteriores, se realiza el dimensionamiento definitivo de las proposiciones, que pueden extenderse en esta etapa. Esta extensión se da ya sea por incorporación de segmentos no proposicionales (reglas de extensión) como por incorporación de proposiciones en otras (reglas de inserción), como por reglas que combinan ambos tipos de operaciones. Además de extenderse/insertarse, las proposiciones pueden coordinarse y se proveen reglas de coordinación a estos efectos. Finalmente, se marcan las unidades textuales de enunciación (reglas enunciado). La salida de este proceso tiene como resultado el análisis estructural representado en (1).

En otras palabras, esta sección revela el proceso derivacional de (1).

5 Evaluación de resultados

Para el análisis cuantitativo de resultados se utilizaron las medidas standard de precisión y recuperación². El par precisión / recuperación presenta alguna ventaja respecto a otras medidas (por ejemplo, cruzamiento de corchetes [CAR98]) para validación contra corpus anotados: básicamente, se miden tanto los constituyentes equivocados como los omitidos. La validación se realizó a partir del análisis de un conjunto de artículos periodísticos no incluidos en el

² precisión = total props. correctas recuperadas / total props. recuperadas; recuperación = total props. correctas recuperadas / total props. en los textos

corpus utilizado para la inferencia de reglas. Estos textos contienen 176 oraciones, de largo promedio 40 palabras por oración. Se obtuvieron tasas de precisión y de recuperación del 91%. Restringiendo los cálculos a oraciones consideradas “complejas” (más de 3 proposiciones), estos resultados empeoran y la proporción de proposiciones correctamente recuperadas sobre el total de proposiciones desciende al 87%.

El análisis de los errores permite discriminar entre errores externos originados por el etiquetador y errores o insuficiencias del propio sistema. Dentro de los últimos podemos discriminar asimismo entre errores fácilmente corregibles (p.ej., la no inclusión de alguna locución conjuntiva y eventuales reglas de desambiguación asociadas) y problemas derivados de la no consideración o simplificación excesiva de algún fenómeno lingüístico. Como ejemplos para la segunda opción podemos mencionar la necesidad de realizar un tratamiento más depurado de algunas categorías indicadas por puntuación, como los incisos y las series.

Bibliografía

- [BOS99] Bosque, I. y V. Demonte (dirs.) (1999): *Gramática Descriptiva de la Lengua Española*, Espasa Calpe, Madrid.
- [CAR98] Carrol, J., T. Briscoe, A. Sanfilippo, (1998) *Parser Evaluation: a Survey and a new proposal*. Proceedings First Conference on Linguistic Resources, Granada. pp. 447-455.
- [CAV03] Caviglia, S., Couto, J., Rosá, A., Wonsever, D. (2003) “Reconocimiento de indicadores de inicio de proposición en español”, *Actas del VIII Simposio de Lingüística Aplicada*, Santiago de Cuba.
- [DES97a] Desclés, J.-P., E. Cartier, A. Jackiewicz, J.-L. Minel (1997): “Textual Processing and Contextual Exploration Method.”, *Context* 97, Universidade Federal do Rio de Janeiro, Brésil. pp. 189-197.
- [DES97b] Desclés, J.-P. (1997): “Systèmes d’exploration contextuelle”, *Co-texte et calcul du sens*. Claude Guimier (éd.). Presses Universitaires de Caen, pp. 215-232.
- [GAR02] Garat, D. y D. Wonsever (2002): “A Constraint Parser for Contextual Rules”, *Actas de SCCC*, Chile.
- [GRA01] Grassi, M., M. Malcuori, J. Couto, J. J. Prada, D. Wonsever (2001): “Corpus informatizado: textos del español del Uruguay (CORIN)”, *Actas SLPLT-2*, Jaén.
- [HOB97] Hobbs, J., P. D. Appelt, J. Bear, D. Israel, M. Kameyama, M. Stickel, M. Tyson (1997): “FASTUS: A Cascaded Finite-State Transducer for Extracting Information”, *Natural-Language Texts in Finite-State Language Processing*. Roche, E., Y. Schabes (eds.). MIT Press.
- [PAV99] Pavón Lucero, M. V. (1999) “Clases de partículas: preposición, conjunción y adverbio” en Bosque, I. y V. Demonte (1999) cap. 9 pp. 565-654.
- [WON01] Wonsever, D., J.-L. Minel (2001): “Contextual Rules for Text Analysis”, *Lecture Notes in Computer Science 2004*, February.