Procesamiento del Lenguaje Natural, nº39 (2007), pp. 285-286 recibido 06-06-2007; aceptado 22-06-2007

JBeaver: Un Analizador de Dependencias para el Español*

Jesús Herrera
Departamento de Lenguajes y Sistemas Informáticos
Universidad Nacional de Educación a Distancia
C/ Juan del Rosal, 16, E-28040 Madrid
jesus.herrera@lsi.uned.es

Pablo Gervás, Pedro J. Moriano, Alfonso Muñoz, Luis Romero
Departamento de Ingeniería del Software e Inteligencia Artificial
Universidad Complutense de Madrid
C/ Profesor José García Santesmases, s/n, E-28040 Madrid
pgervas@sip.ucm.es, {pedrojmoriano, alfonsomm, luis.romero.tejera}@gmail.com

Resumen: *JBeaver* es un analizador de dependencias para el español desarrollado utilizando una herramienta de aprendizaje automático (*Maltparser*). Este analizador se caracteriza por ser el único públicamente disponible para el español, ser autónomo, fácil de instalar y de utilizar (mediante interfaz gráfica o por comandos de consola) y de elevada precisión. Además, el sistema desarrollado sirve para entrenar de manera sencilla modelos de *Maltparser*, por lo que se configura en potencia como un analizador de dependencias para cualquier idioma.
Palabras clave: Análisis de dependencias, Maltparser, JBeaver

Abstract: *JBeaver* is a dependency parser built using the *Maltparser* machine-learning tool. It is publically available , easy to install and to use, and provides high precision. It also allows training *Maltparser* models for any language, so it can be used to train dependency parsers for any language.
Keywords: Dependency parsing, Maltparser, JBeaver

1. JBeaver

El objetivo final era un analizador de dependencias para el español, de libre distribución y que fuera fácil de instalar y manejar. Por otra parte, se debían acotar los esfuerzos dada la limitación de recursos del proyecto.

1.1. Decisiones de Diseño y Elección de Recursos

Bajo los requisitos del proyecto era inviable el desarrollo de la algorítmica propia del análisis de dependencias, por lo que se hubieron de buscar recursos que evitasen esta labor. Uno de ellos es *Maltparser* (Nivre et al., 2006), que finalmente fue el elegido por las características que ofrecía: era autónomo, fácil de integrar como subsistema y proporcionaba unos resultados notables en las lenguas para las que se había probado hasta el momento.

Tanto para el entrenamiento de *Maltparser* como para la ejecución como analizador del modelo aprendido es necesario proporcionar el etiquetado de categorías gramaticales de las palabras del texto. Como uno de los objetivos era que *JBeaver* pudiese recibir textos sin anotar, para facilitar al máximo su uso, la propia herramienta debería etiquetar los textos recibidos a la entrada con su categoría gramatical. Igualmente que en el caso del análisis de dependencias, tampoco era factible el desarrollo de algoritmos para el etiquetado de categorías gramaticales. Por ello, fue necesario buscar una herramienta disponible, autónoma, fiable y fácil de integrar en *JBeaver*; esta fue, finalmente, *TreeTagger* (Herrera et al., 2007) (Schmid, 1994).

Tanto el entrenamiento de *Maltparser* como la evaluación del producto final obtenido requieren de corpora convenientemente anotados. Este aspecto se vio resuelto con el uso del corpus *Cast3LB* (Navarro et al., 2003), que contiene textos en español anotados con sus análisis sintácticos de constituyentes. Para obtener los corpora adecuados para el entrenamiento de *Maltparser* y la evaluación de *JBeaver*, se desarrolló una herramienta para convertir los análisis de consti-

* Partially supported by the Spanish Ministry of Education and Science (TIN2006-14433-C02-01 project).

 © 2007 Sociedad Española para el Procesamiento del Lenguaje Natural

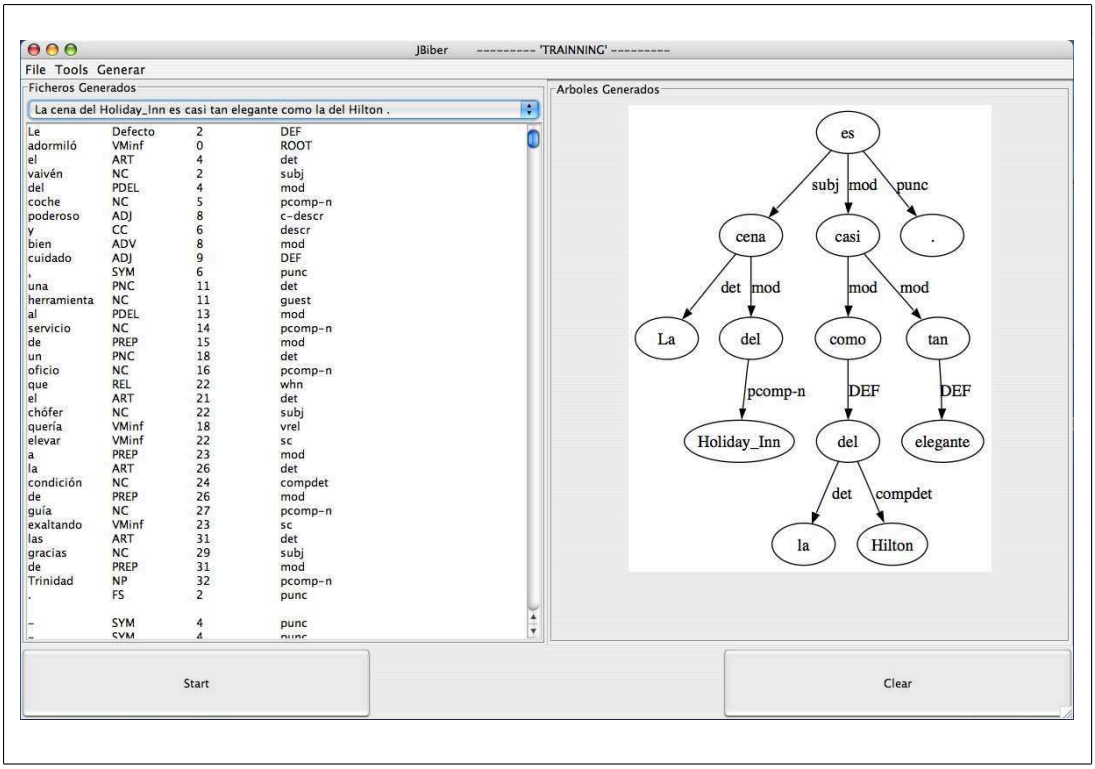

Figura 1: Interfaz gráfica de *JBeaver*

tuyentes del *Cast3LB* en análisis de dependencias (Herrera et al., 2007).

Otro aspecto definitorio de *JBeaver* es su interfaz gráfica de usuario (ver Figura 1). En ésta se muestran los análisis obtenidos en forma de grafos, para que los datos resulten visualmente cómodos de interpretar. No obstante, también se proporciona la salida en forma de fichero de texto, para que pueda ser fácilmente manipulado por otros programas. La representación de los grafos quedó delegada a *Graphviz*, como otro de los subsistemas que forman parte de *JBeaver*.

1.2. Pruebas

De las diversas pruebas a que fue sometido *JBeaver* durante la fase de desarrollo, son de destacar las relacionadas con el rendimiento del núcleo analizador, es decir, del modelo entrenado de MaltParser. Para ello se seleccionó una fracción del corpus *Cast3LB*, de 431 palabras, no usada previamente para el entrenamiento del modelo de *Maltparser* y se generó a partir de ella un corpus con análisis de dependencias, que se tomó como modelo de referencia. Se extrajeron los textos sin etiquetar de ese corpus y se sometieron al análisis de dependencias efectuado por el modelo aprendido. Posteriormente se comprobó la salida proporcionada por el analizador con el modelo de referencia, comprobándose que se habían encontrado correctamente el 91 % de las dependencias.

Bibliografía

J. Herrera, P. Gervás, P.J. Moriano, A. Muñoz, L. Romero. 2007. *Building Corpora for the Development of a Dependency Parser for Spanish Using Maltparser.* (SEPLN, this volume).

B. Navarro, M. Civit, M.A. Martí, R. Marcos, B. Fernández. 2003. *Syntactic, Semantic and Pragmatic Annotation in Cast3LB.* Proceedings of the Shallow Processing on Large Corpora (SproLaC), a Workshop on Corpus Linguistics, Lancaster, UK.

J. Nivre, J. Hall, J. Nilsson, G. Eryiğit and S. Marinov. 2006. *Labeled Pseudo–Projective Dependency Parsing with Support Vector Machines.* Proceedings of the CoNLL-X Shared Task on Multilingual Dependency Parsing, New York, USA.

H. Schmid. 1994. *Probabilistic Part-of-Speech Tagging Using Decission Trees.* Proceedings of the International Conference on New Methods in Language Processing, pages 44–49, Manchester, UK.