UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

EXTRACCIÓN AUTOMÁTICA DE TÉRMINOS EN CONTEXTOS DEFINITORIOS

T E S I S

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS
(COMPUTACIÓN)

P R E S E N T A :

LUIS ALBERTO BARRÓN CEDEÑO

DIRECTOR DE TESIS:
DR. GERARDO SIERRA MARTÍNEZ

CODIRECTOR DE TESIS:
DR. HUMBERTO CARRILLO CALVET

MÉXICO, D. F. 2007.

A Silvia,
que se mantenga viva la ilusión

Agradecimientos

Antes de comenzar quisiera señalar a algunos de los cómplices en esta etapa de mi vida.

El Dr. Gerardo Sierra me permitió continuar en el Instituto de Ingeniería. No sólo me otorgó este tema de tesis, sino que me mostró el mundo de la lingüística computacional.

El Dr. Patrick Drouin me dio una nueva visión de nuestra área y un modelo a seguir.

La Mtra. Iveth Carreño me ayudó a sobrevivir en un lugar extraño y, a pesar de las dificultades, continuar con esta investigación.

El Mtro. Elio Villaseñor siempre me dio ideas para continuar y me mostró que no estaba solo.

El Dr. Humberto Carrillo lograba encontrar momentos para escucharme a pesar de haber tenido un año tan duro.

Los participantes en el proyecto de contextos definitorios. La intrincada inmersión en el mundo de la lingüística habría sido aún más dura sin ellos.

El Dr. J. Alberto Escobar, quien en los momentos de mayor incertidumbre ha estado dispuesto a orientarme, ha compartido sus experiencias y jamás me ha dejado de llamar tocayo.

Teresita, Eduardo, Carlos, Alfonso, César, Antonio y el Grupo de Ingeniería Lingüística entero. Me han permitido adentrarme en esta interesante área en medio de muchas buenas vivencias.

El Instituto de Ingeniería, que desde hace más de cinco años me ha dado las condiciones necesarias para el desarrollo de mi carrera.

El Dr. Boris Escalante, Lulú, Diana y Amalia. Porque siempre que necesité algo del posgrado estuvieron ahí, incluso cuando se trataba sólo de platicar un rato.

La Universidad Nacional Autónoma de México, que desde que tenía 15 años me ha dado, sin pedir nada a cambio, educación de calidad.

Mi familia, pues siempre ha estado ahí, incluso cuando yo no lo notaba. Principalmente a mi mamá y a Silvia.

Los amigos que han estado conmigo todos estos años, sorteando dificultades y pasando buenos momentos. Kristian, Rafael, Paco, Víctor...

El profesor Jesús Jiménez M. me mostró las matemáticas y me dio el perfil que he buscado conservar.

Gracias a los profesores que se sentaron a leer mi trabajo y me orientaron, porque me ayudaron a generar un trabajo con calidad.

Agradezco al CONACYT y la DGEP por apoyarme económicamente durante todo este tiempo, permitiéndome dedicarme sólo a estudiar y a obtener este grado.

Índice de contenido

Índice de figuras

Índice de tablas

Introducción

Este trabajo se ocupa del tratamiento, en particular la extracción automática, de términos a partir del análisis automático de textos no estructurados.

Entre los proyectos que desarrolla actualmente el Grupo de Ingeniería Lingüística (GIL), que se desempeña en el ámbito de la lingüística computacional al interior del Instituto de Ingeniería de la UNAM, se encuentra el de *Extracción de conceptos en textos de especialidad a través del reconocimiento de patrones lingüísticos y metalingüísticos*[1]. El objetivo principal de este proyecto es analizar el comportamiento de los elementos que participan en la definición de un concepto (término, verbo definitorio y definición, entre otros), los cuales conforman un *contexto definitorio*, para desarrollar la tecnología aplicada necesaria para su reconocimiento y extracción de manera automática.

El punto de partida del proyecto es la generación del *Corpus de Contextos Definitorios*, que contiene un compendio de los contextos definitorios hallados en distintos corpus especializados. Sobre este corpus se plantean realizar no sólo los análisis necesarios, sino las pruebas de las herramientas generadas durante el proyecto. Al final, se espera que todos sus elementos queden delimitados.

Es aquí donde este trabajo de tesis cobra vida, pues en él se busca establecer las bases y la implementación de software necesarios para la extracción de los términos que aparecen al interior de un contexto definitorio, tarea que será la primer aplicación del extractor generado.

Con el desarrollo de este extractor se espera no sólo apoyar en las diversas investigaciones que se llevan a cabo al interior del GIL, sino además proporcionar una herramienta útil para diversos profesionales que incluyen a los términos entre sus herramientas de trabajo (terminólogos, traductores, editores, etc.).

El evitar la revisión manual de un texto completo para extraer su terminología puede resultar de gran utilidad para este tipo de especialistas que actúan del lado de la lingüística, entre muchas otras aplicaciones, en la generación de herramientas lexicográficas como diccionarios y glosarios, en el análisis diacrónico de las lenguas de especialidad (para encontrar el momento en que un término surge en una disciplina, por ejemplo) y para analizar las variaciones dialectales entre distintas regiones que hablan la misma lengua (pues en ocasiones existen distintas formas de llamar a un mismo elemento en un área de especialidad en distintas regiones, como es el caso de los términos *red neuronal* y *red de neuronas*, los cuales ocurren respectivamente en el español de México y España).

Desde un punto de vista más computacional, la extracción automática de términos puede ser empleada como base para mejorar la clasificación automática de documentos, permitiendo realizar búsquedas de información más certeras dentro de un universo significativo de documentos de algún área de especialidad. Incluso puede ser una buena herramienta para el desarrollo de las ontologías que dan sustento al web semántico y conformar bases de conocimiento organizadas por medio de ligas semánticas que pueden ser explotadas en sistemas de búsqueda y de pregunta-respuesta.

Aunque se trata de un trabajo interdisciplinario de procesamiento de lenguaje natural que requiere conocimientos tanto de lingüística como de computación, no se debe pasar por alto que este trabajo se desarrolla dentro del Posgrado en Ciencia e Ingeniería de la Computación, por lo que, si bien se detalla todo el conocimiento lingüístico necesario para esta implementación, se da mayor importancia a los elementos computacionales tratados.

El objetivo principal de este trabajo consiste en:

---

[1] Proyecto CONACYT 46832 H.

Adaptar una metodología, tanto en el ámbito lingüístico como computacional, para la extracción de candidatos a término de cualquier longitud en textos de especialidad en español, en particular aquellos que ocurren al interior de un contexto definitorio.

Se presenta la modificación de un algoritmo para la extracción automática de términos en textos técnicos y científicos conocido como C-value/NC-value. Dicha adaptación no sólo es de lenguaje (pues el algoritmo original fue diseñado para el tratamiento de textos en inglés y se busca ahora utilizarlo para textos en español) sino de funcionamiento, pues se ha observado la posibilidad de mejorar el rendimiento de dicho algoritmo por medio de la modificación en algunas etapas de la extracción.

Resta solamente, antes de entrar de lleno al tema, describir brevemente cómo se encuentra organizado el contenido del trabajo:

En el capítulo 1, titulado *Conceptos básicos para la extracción de términos*, se ofrece una breve introducción al estudio de la terminología desde el punto de vista de la lingüística, incluyendo una descripción de la propia disciplina y su objeto de estudio: el término. Además se ubica a la tarea de extracción de términos como una tarea de la *extracción de información*.

Por otro lado, se presenta en la segunda parte de este capítulo una descripción de los contextos definitorios. Dicha explicación es necesaria debido a que, como ya se señaló, la primera aplicación que tendrá este extractor es la ubicación de los términos que se encuentran al interior de un contexto definitorio (tanto aquel que esté siendo definido como los que son utilizados para este fin).

Contando ya con estas bases, en el capítulo 2, *Panorama actual de la extracción automática de términos*, se ofrece una descripción de diversas técnicas que se han utilizado en aras de la extracción automática de términos. Este capítulo incluye un análisis de algunos de los sistemas de extracción de términos que han sido desarrollados hasta el momento.

El capítulo 3, denominado *Metodología*, trata ya sobre el desarrollo de la aplicación. Se describen en él los distintos corpus de prueba que han sido utilizados para el diseño y prueba de la herramienta desarrollada. Se describen las herramientas adicionales necesarias para el pretratamiento que requiere un corpus antes de proceder a la detección de los candidatos a término que contiene. Se determinan los patrones lingüísticos necesarios para la búsqueda de candidatos a término dentro del texto analizado. Este capítulo cierra con una descripción del algoritmo C-value/NC-value original para finalmente dar paso a la descripción de las adaptaciones que se le han hecho.

Por último, en el capítulo 4, *Evaluación*, se muestran y evalúan los resultados obtenidos por el extractor.

Capítulo 1: Conceptos básicos para la extracción de términos

Donde se verá qué son el término y la terminología, se ubicará el trabajo en el ámbito computacional y se abordarán los contextos definitorios.

Una de las áreas integrantes de la *lexicografía* es la terminología. Ésta se encarga de estudiar subconjuntos bien definidos de la lengua (al interior de textos de especialidad), en los que se hallan palabras que son utilizadas para nombrar objetos (lógicos y físicos), procesos, acciones y diversos elementos más dentro de un círculo de especialistas en algún área en particular, los cuales son conocidos como terminologías.

El estudio de la terminología permite, entre otras cosas:

- La creación de diccionarios de distintos tipos, como los *semasiológicos*[2] y los *onomasiológicos*[3] [Sierra y McNaught 2000].
- El estudio de la evolución y el estado actual de una disciplina, ya que es posible, por ejemplo, determinar el momento en que, dentro de ella, se comienza a utilizar un término, lo que puede implicar su invención o su adopción a partir de otra disciplina.
- Desde un punto de vista más computacional, la creación de herramientas de búsqueda basadas no sólo en *técnicas estadísticas*, sino también *lingüísticas*.
- La implementación de programas de *agrupación de documentos* basados en datos que son más relevantes por su relación con el texto tratado, pues se pueden realizar con base no en el universo completo de palabras en el documento, sino sólo por medio de las palabras del documento pertenecientes al área de especialidad tratada.
- La generación de *ontologías* que sirvan como base para el *web semántico*, un enfoque de explotación del web en el que los recursos no se encuentran conectados por cadenas "sin significado", sino por medio de relaciones semánticas.

El estudio de los términos implica, antes que nada, su obtención. Para recabar el conjunto de términos de un área en particular, existen dos métodos totalmente distintos pero que no son necesariamente mutuamente excluyentes:

a) su obtención por medio de una interacción directa con un conjunto de especialistas (por medio de diversas actividades como cuestionarios, entrevistas o encuestas), y
b) su detección en un conjunto de documentos en los que se aborde la disciplina de interés (artículos, libros, tesis, informes), lo que implica leer una buena cantidad de dichos documentos.

Existen trabajos [Cardero 2000] que son prueba de la combinación de estas dos técnicas para la recopilación de una terminología.

Ambos métodos para la obtención de terminologías resultan muy costosos (en dinero y, sobre todo, en tiempo). La búsqueda de la automatización del proceso de obtención de términos puede traer grandes beneficios al trabajo del terminólogo.

El principal objetivo del presente trabajo es la generación de una aplicación capaz de realizar una *extracción de términos de manera automática* sobre un *corpus* de trabajo compuesto por documentos, de carácter técnico o científico[4], en español. En este primer capítulo, se ofrece una breve introducción al estudio de la terminología, incluyendo una descripción de la propia disciplina y su objeto de estudio, además de la ubicación de la extracción de términos como una

[2] Aquellos que van del término al concepto.
[3] Los que van del concepto al término.
[4] En realidad, los documentos no deben ser forzosamente de carácter técnico o científico, pero si lo que se busca son términos, es conveniente hacerlo en documentos en los que se espera que se encuentren.

tarea de la *extracción de información*.

Por otro lado, se ofrece, en la segunda parte, una descripción de los *contextos definitorios*, porciones de texto en los que un término es definido. Dicha explicación es necesaria debido a que la principal aplicación que tendrá este extractor, en principio, será la ubicación de los términos que se encuentran al interior de un contexto definitorio (tanto aquel que esté siendo definido como los que son utilizados para ese fin).

Con esto como base será posible abordar, en el tercer capítulo, la problemática que se busca solucionar. Como primer tema abordado, se encuentra la terminología, considerada como disciplina y como un elemento intrínsecamente constituyente de toda área de especialidad.

1.1 La terminología estudia a los términos, los términos conforman a la terminología

Este título "auto-referido" tiene su justificación en que, como se verá más adelante, la terminología y su objeto de estudio, el término, llevan entre sí una relación que puede ser considerada como circular [Kageura y Abekawa 2006]. A continuación se presentan las características de la terminología como disciplina para pasar, en el siguiente apartado, al estudio del término, considerado el objeto de estudio de la terminología y cuya obtención automatizada a partir del análisis de textos es materia principal de este trabajo.

1.1.1 La terminología

La terminología ha existido al interior de las áreas de especialidad desde su mismo surgimiento. Debido al desarrollo tecnológico, que ha dado uno de sus principales saltos durante la Revolución Industrial del siglo XVIII, y a la necesidad de estandarización de conceptos y términos en pro de la claridad y el mismo desarrollo, se produjo primero la necesidad del estudio de la terminología como parte de la ciencia y la tecnología.

Sin embargo, la terminología es una disciplina que ha florecido, de manera independiente, en los últimos tiempos, particularmente desde mediados del siglo XX. Su mayor impulso se ha dado en regiones en las que se combina el alto desarrollo tecnológico y la inmersión en una fuerte interacción entre lenguas. Así, entre las escuelas que más han destacado en el estudio de este tema están:

- La escuela rusa, considerada precursora de los estudios sobre terminología. El desarrollo de esta disciplina se vio impulsado, a mediados del siglo XX, por el interés de este país en el desarrollo tecnológico propio carente de intercambio con el exterior. Algunos de sus principales exponentes son D. S. Lotte, que diera las bases para la creación de *sistemas de términos* propios de cada área de especialidad, y A. A. Reformatskii quien se ocupara de la definición del término y la terminología [Cabré, et. al. 2001].
- La escuela austriaca. Su principal representante, E. Wüster es considerado el padre de la terminología clásica y el fundador de la escuela de Viena. Él se encargaría de sacar el estudio de las terminologías del interior de cada una de las áreas de especialidad para concebirla como una materia autónoma interdisciplinaria en permanente interacción con las diversas disciplinas [Wüster 2003].
- La escuela canadiense, impulsada por la interacción del francés y el inglés como lenguas

oficiales de Canadá. R. Dubuc es considerado el padre de la terminología de Québec. Su principal interés era la *terminología bilingüe*, en la que se realizan recopilaciones y estudios de las terminologías de manera paralela en varios idiomas (en este caso francés e inglés) y cuyo principal objetivo es la traducción de documentos[5]. El principal interés de Dubuc es llevar la terminología al trabajo práctico. Uno de los principales proyectos actuales en materia de terminología en Canadá es *Termium*[6], una base de datos terminológica a cargo de la Oficina de Traducción del gobierno canadiense.

- La escuela catalana, desarrollada al interior de una atmósfera en la que se mezclan el catalán y el español. Su principal representante es M. T. Cabré, fundadora de la conocida como *teoría comunicativa de la terminología*, opuesta a la teoría clásica de Wüster, en la que se considera a los términos como unidades cognitivas inexistentes fuera de la lengua natural. Concibe a la terminología como una disciplina flexible que debe adaptarse no sólo a la disciplina, sino al entorno en que se encuentra, por lo que no se encarga de dar los lineamientos que debe seguir una terminología, sino de estudiarla tal como existe [Cabré 1999].

Habiendo visto ya una breve reseña de las principales escuelas dedicadas al estudio de la terminología, queda ahora mostrar de qué se trata.

Dentro del enfoque de la terminología como disciplina, Cabré [1995] la define como "la disciplina que se ocupa de los términos especializados", a la vez que la considera como el "conjunto de directrices o principios que rigen la recopilación de términos" (lo que confirma lo dicho anteriormente, que Cabré considera que la terminología no debe regir a los términos, sino únicamente estudiarlos).

Ampliando esta definición, Cardero [1999] señala que la terminología se ocupa de "identificar el vocabulario de una especialidad en forma sistemática en una situación comunicativa específica en los textos propios de la especialidad y entre los profesionales del área, analizarlo desde la lingüística y, si es necesario, crearlo entre el especialista y el terminólogo, además de normalizarlo para un funcionamiento concreto con la finalidad de responder a las necesidades de expresión de sus usuarios".

Además, la terminología es considerada como una disciplina integrante de la lingüística y como un campo *multidisciplinario* construido a partir de las teorías del conocimiento, la comunicación y el lenguaje [Cabré 1999], teniendo a los términos, las unidades constituyentes de los vocabularios de especialidad, como su objeto de estudio.

Por otro lado, la terminología como elemento de la ciencia, es el vocabulario de especialidad con que cuenta un área en particular. Es decir, la terminología es el conjunto de los términos pertenecientes a un campo o área de especialidad.

De hecho, Reformatskii [1961] señalaría que "el sistema de conceptos de una ciencia en particular está relacionado íntimamente con su terminología. Los términos para cada ciencia (en cada una de sus tendencias) son calculables y obligatoriamente están relacionados con los conceptos de cada ciencia, puesto que reflejan el sistema de conceptos de la ciencia en cuestión".

En acorde con lo anterior, la misma Cabré [1995] considera que la terminología es parte de la lexicología y que los lenguajes de especialidad son subsistemas de la lengua general, por lo que tienen ciertas características "heredadas" por ellos.

Así que, parafraseando al título de este apartado, la terminología estudia a los términos y

[5] http://www.cee.umontreal.ca/robertdubuc.htm
[6] http://www.termium.gc.ca/

los términos conforman a la terminología. Se trata de una definición circular parecida a la de conjunto en matemáticas.

En resumen, la terminología es a la vez un sistema de conceptos (y sus términos) de un área de especialidad y una disciplina multidisciplinaria que se encarga de estudiar (y en ocasiones conformar) dicho sistema.

Sin embargo, aún no se ha abordado al término. Resulta pertinente tratarlo en este momento pues se trata del objeto de estudio de la terminología.

1.1.2 Los términos

El objeto de estudio de la terminología es el término. De la misma manera, los *términos* son las unidades constituyentes de los vocabularios de especialidad: las terminologías.

Para Dubuc [1992], el término es "el elemento constitutivo de cualquier nomenclatura terminológica que esté relacionada con una lengua de especialidad […], es la denominación de un objeto, propio de una determinada área de especialidad". Cabe señalar, primeramente, que un término es dependiente del contexto. Si es sacado de contexto, es decir, del entorno del área de especialidad a la que pertenece, pierde la categoría de término.

Sin embargo, con esta definición puede no quedar muy clara la diferencia que existe entre un término y cualquier otra palabra o, de manera más general, cualquier otro sintagma[7].

Cabré [1995] considera que "una palabra es una unidad descrita por un conjunto de características lingüísticas sistemáticas y dotada de la propiedad de referirse a un elemento de la realidad...", mientras que un término es "...una unidad de características lingüísticas parecidas, utilizada en un dominio de especialidad". Por tanto, considera término a toda palabra que forme parte de un ámbito especializado y que, en su interior, tenga un concepto asociado preciso.

Sin embargo, esto no es suficiente aún para diferenciar entre un término y otros tipos de sintagmas. Otra característica importante con la que cuentan los términos es que, al interior de un área de especialidad, no deberían (pues en ocasiones ocurre, como en el mismo caso del término *terminología*[8]) existir relaciones de *sinonimia* u *homonimia*. Por ende, un mismo término no debe tener asociados distintos conceptos (homonimia), ni deben existir varios términos asociados a un mismo concepto (sinonimia).

Esta relación biunívoca entre un término y su concepto permite que el lenguaje de especialidad sea claro y conciso, lo que evita cualquier posibilidad de caer en ambigüedades. Dicha ambigüedad podría no ser de gran relevancia en un ámbito coloquial, pero en uno científico podría resultar en grandes atrasos, confusiones e incluso catástrofes.

Entonces, si un término es sacado de contexto, es decir, del entorno del área de especialidad a la que pertenece, pierde la categoría de término. Como Reformatskii [1961] lo señalara: "el campo del término está presente en cada una de las terminologías, y fuera de éstas pierde su carácter de término". Ejemplo claro de ello, es el término *lengua*, que es utilizado a continuación para ilustrar este hecho.

Lengua, además de ser una palabra de uso general con relaciones de sinonimia y homonimia intrínsecas, es un término que existe al interior de varias disciplinas, entre las que se

---

[7] Un sintagma es el conjunto de una o más palabras cohesionadas que tienen un concepto asociado. P. ej. *control remoto* o *por favor*.

[8] Que, como ya se observó, se refiere tanto al estudio de los términos de un área de especialidad como al conjunto de términos de dicha área, por lo que presenta homonimia.

encuentran la lingüística y la anatomía. Dentro de dichas disciplinas, el concepto asociado al término *lengua* es totalmente distinto:

> Anatomía: La lengua es un órgano móvil situado en el interior de la boca, impar, medio y simétrico, que desempeña importantes funciones como la masticación, la deglución, el lenguaje y el sentido del gusto...
>
> Lingüística: El término lengua [natural] designa una variedad lingüística o forma de lenguaje humano con fines comunicativos que está dotado de una sintaxis y que obedece supuestamente a los principios de economía y optimidad[9]

Además, si se analiza la palabra *lengua* como una palabra del léxico común, fuera de un área de conocimiento en particular, se observa que se trata de una palabra que presenta homonimia, pues tiene varios conceptos asociados (al menos, en un nivel menos especializado, los dos mencionados anteriormente) y sinonimia con palabras como *idioma*. Sin embargo, si se analiza dentro de cada una de las disciplinas técnicas o científicas a las que pertenece, por separado, tanto la homonimia como la sinonimia desaparecen. *Lengua* se convierte en un término de especialidad con un concepto asociado distinto y único en cada una de estas disciplinas.

Con este sencillo ejemplo se puede observar que un mismo sintagma puede pertenecer a diversas disciplinas, teniendo asociado un concepto distinto y único en cada una de ellas, lo que le permite conservar la categoría de término al interior de cada una de ellas.

Retomando el ejemplo, si el sintagma *lengua* es sacado de su área de especialidad, se convierte en una palabra de la lengua general, que puede presentar homonimia o sinonimia sin restricciones o conflictos.

Entrando un poco al campo de la computación, considérese ahora al término *redes neuronales*. Si bien, en biología existe el término *redes neuronales biológicas* y en computación el de *redes neuronales artificiales*, los especialistas de ambas disciplinas a menudo se refieren a ellas sólo como *redes neuronales*, teniendo un concepto asociado totalmente distinto en cada una de las disciplinas en las que se encuentra inmerso:

> Biología: ...un conjunto de neuronas conectadas o relacionadas funcionalmente en el sistema nervioso periférico o sistema nervioso central...
>
> Computación: Arreglo de nodos diseñado para la modelación de funciones matemáticas...

Resulta claro que el término depende totalmente del área de especialidad abordada. En este caso, puede observarse que un término no está limitado a estar constituido por un sintagma de una sola palabra, como en el caso de *procesador* o *memoria*, sino que puede estar conformado por un sintagma multipalabra como *redes neuronales artificiales*. De hecho, las palabras que conforman a un término multipalabra pueden ser, por sí mismas, términos, como en el caso de *red*, *red de computadoras* y *red de computadoras inalámbrica*[10].

Así, es posible considerar, de manera concisa, que **un término es un sintagma asociado a un concepto, por medio de una relación biunívoca, dentro de un área de especialidad**.

Una discusión que sigue abierta en el estudio de la terminología es si los nombres propios,

[9] Estas definiciones y varias más de los siguientes ejemplos fueron obtenidas de *Wikipedia, La enciclopedia libre*, es.wikipedia.org, consulta: diciembre de 2006

[10] Dicha inclusión entre sintagmas terminológicos de distintas longitudes es de gran importancia para el algoritmo C-value/NC-value, aplicado en este trabajo y que será visto con detalle en el tercer capítulo.

como *Alan Turing* y *John Hopcroft*, y las marcas, como *GNU/Linux* y *OpenOffice*, deben ser considerados términos dentro de un área de especialidad o pueden ser casos de otros términos como en el caso de *GNU/Linux* con respecto al término *sistema operativo*.

Si bien, existen en la actualidad marcas registradas como *iPod* o *Windows* que definitivamente tienen un concepto asociado (pues se trata de un reproductor de música digital y un sistema operativo que son reconocidos mundialmente), la duda que queda es si estos sintagmas son realmente utilizados en un ambiente de especialidad, que como se señaló previamente, es el hábitat natural de los términos. Es verdad que sintagmas como *iPod* han pasado de ser una simple marca a tener toda una familia de objetos asociados, pero no se debe olvidar que el concepto que se les asocia existe en un ambiente coloquial y no pertenece a un área especializada. Por ende, a consideración del autor de este trabajo, ninguna marca que sea utilizada en un ambiente no especializado puede ser considerada un término.

En cuanto a las personas y siguiendo en el ámbito de la computación para realizar este análisis, nombres como *Alan Turing* o *Richard Stallman* sí pueden ser considerados términos pues dentro del área especializada de la computación (y no aquella cuya frontera es difusa con la lengua general) tienen un concepto asociado como el padre de la inteligencia artificial y del software libre respectivamente. Si bien, el autor no acepta que todos los nombres de personas sean términos (pues en realidad la mayoría son entidades nombradas[11]), algunos de éstos están cercanos a ser términos.

Un tema que no ha sido abordado aún en este apartado es la construcción morfosintáctica de los términos en español. Debido a que es materia importante del trabajo realizado, se aborda de manera detallada en el capítulo 3.

Hasta ahora se ha dado un panorama general de la terminología y el término. Resta ubicar esta investigación en el contexto de las ciencias de la computación. Es por esta razón que se analiza a continuación la posición de la extracción de términos dentro de la extracción de información.

1.2 Extracción automática de términos en el contexto computacional

¿Qué es la extracción automática de términos y por qué es objeto de estudio de las ciencias de la computación? Para explicarlo, es necesario saber qué es el *procesamiento de lenguaje natural* y la *lingüística computacional*. Pero antes, resulta interesante mostrar el evento que fue precursor de muchas de las investigaciones al respecto.

Tal vez el primer acercamiento formal de la computación con el lenguaje natural haya sido realizado por Alan Turing en su célebre artículo *Computing Machinery and Intelligence* [Turing 1950]. El tema principal de dicho artículo es la discusión sobre la posibilidad de que las máquinas sean o no capaces de pensar. Con el objetivo de dar respuesta a esta duda, propuso el que en ese momento llamó el *juego de la imitación*.

Visto de manera breve, el juego de la imitación consiste en un experimento en el que se cuenta con tres sujetos (llamémosles *A*, *B* y *C*) en tres habitaciones separadas, sin contacto visual entre ellos y teniendo como único medio de comunicación un dispositivo para escribir y leer los mensajes de los otros. *A* y *B* son un hombre y una mujer, respectivamente, mientras que *C* es un interrogador que, por medio de la respuesta obtenida de diversas preguntas hechas a *A* y *B*, debe

[11] Una entidad nombrada es un sintagma que denota un objeto del tipo persona, organización, lugar, fecha o cantidad.

determinar, con certeza, cuál de los sujetos de prueba es el hombre y cuál es la mujer[12]. Acto seguido, Turing propone modificar un poco la prueba cambiando a los sujetos participantes en ella:

> "... «¿Qué pasará cuando una máquina tome el lugar de *A* en este juego?» ¿El interrogador tendrá los mismos errores y aciertos que cuando se juega con un hombre y una mujer? Éstas preguntas reemplazan a la original, «¿Pueden las máquinas pensar?»"[13]

Ahora no se trata de identificar a un hombre y a una mujer, sino a un ser humano y una máquina. Si *C* no fuera capaz de diferenciar entre *A* y *B*, entonces se podría considerar, en palabras de Turing, que la máquina es capaz de pensar, lo que la convierte en una entidad *inteligente*[14]. Es, en buena parte, debido a este artículo que se considera a Alan Turing como el padre de la Inteligencia Artificial. De hecho, una parte de los esfuerzos en esta área se han destinado exclusivamente a la creación de máquinas capaces de superar la después rebautizada como *prueba de Turing*[15].

Con el tiempo, se observaría que para que una computadora fuera capaz de comprender y generar lenguaje natural (aunque la capacidad de realizar estas tareas no implique intrínsecamente que esté pensando) era necesario contar con conocimiento lingüístico, desde el conocimiento de un léxico determinado y los lineamientos gramaticales de una lengua en particular hasta la capacidad de interpretación del significado, no sólo de las palabras u oraciones, sino de las ideas implicadas: la semántica.

Es así como, desde distintos frentes, surgen varias disciplinas de las que dos en particular difícilmente pueden diferenciarse entre sí: el procesamiento de lenguaje natural (PLN) y la lingüística computacional (LC). Jurafsky y Martin [2000] consideran, en conjunto con estas dos disciplinas, dos más en el seno de cuatro distintas áreas de conocimiento:

- *Lingüística computacional* en lingüística.
- *Procesamiento de lenguaje natural* en ciencias de la computación.
- *Reconocimiento del habla* en ingeniería eléctrica.
- *Psicolingüística computacional* en psicología.

De acuerdo con las ideas de Jurafsky, se debe entender al PLN como el "conjunto de técnicas computacionales que procesan el lenguaje humano, tanto hablado como escrito".

Lo que distingue al PLN de otras aplicaciones (como la de un programa de shell encargado de obtener una lista con los nombres de los usuarios que se han conectado en las últimas 24 horas a un sistema por medio del análisis del texto al interior de una bitácora del sistema operativo) es la necesidad de explotar *conocimiento lingüístico*.

Entre los objetivos principales del PLN se encuentra la recuperación y extracción de información, la clasificación de documentos y, de manera general, la creación de sistemas de

[12] Algunas personas señalan que este juego se le ocurrió a Turing, originalmente, para determinar la heterosexualidad u homosexualidad de una persona, pero ese tema no atañe a este contexto.

[13] Turing, op. cit.

[14] Existe también una amplia discusión sobre qué es la inteligencia y qué factores pueden llevar a considerar que una entidad es inteligente. Para profundizar al respecto, desde el punto de vista de la inteligencia artificial, se recomienda consultar [Russel y Norvig 2003].

[15] Si bien, la prueba de Turing no ha sido superada aún, se han propuesto ya versiones más complicadas de ella en las que se incluye la interacción entre los sujetos a través de elementos adicionales como sonido o imagen, lo que ha impulsado el desarrollo de sistemas de generación de lengua hablada y de procesamiento de imágenes.

procesamiento de lenguaje natural tanto hablado como escrito. Para lograr estos objetivos, es posible realizar desde procesos que pueden ser considerados sencillos, como el conteo de palabras en un texto, hasta la desambiguación de palabras por medio de diccionarios (enfoque lingüístico) o probabilidades de ocurrencia (enfoque estadístico).

Por otro lado, para Grishman [1986], la LC es "el estudio de sistemas computacionales para entender y generar lenguaje natural". La *Association for Computational Linguistics* señala que el principal interés de la LC es la creación de modelos computacionales para varios tipos de fenómenos lingüísticos con base en dos elementos principales: conocimiento (recabado manualmente) y datos (obtenidos a través de métodos estadísticos o empíricos)[16].

Algunos de los objetivos de la LC son la traducción automática, la recuperación y extracción de información y la creación de interfaces humano-máquina.

Como ocurre en otras áreas en la que es aplicada la computación (como en astronomía, por ejemplo), en la lingüística computacional se intenta, por un lado, dar una explicación computacional a algún fenómeno lingüístico y, por otro, de manera mucho más práctica, generar herramientas para el tratamiento de lenguas (como en el reconocimiento de habla, motores de búsqueda, traductores automáticos o editores de texto).

¿Cuál es la diferencia entre PLN y LC? En realidad, pueden ser consideradas como lo mismo, aunque sean abordadas desde distintas áreas y con diferentes contextos y conocimientos. **Un lingüista que desea hacer LC debe traspasar la barrera de la computación mientras que un computólogo debe traspasar la barrera de la lingüística para abordar el PLN**, por lo que de hecho, es común encontrar grupos interdisciplinarios con participantes de ambas áreas que se unen para un fin común.

Como se verá a continuación, cuando se habla de extracción de términos, resulta necesario ubicarse dentro de la amplia intersección existente entre la LC y el PLN.

Una de las áreas, tanto del PLN como de la LC, es la extracción de información. Jackson y Moulinier [2002] definen a la extracción de información como "una aplicación de tecnología computacional para la adquisición, organización, almacenamiento, recuperación y distribución de información".

Dicha disciplina se encuentra inmersa dentro del PLN y se encarga de encontrar "hechos" en documentos no estructurados[17]. Es importante recalcar que las computadoras o, de manera más específica, los programas que se encargan de realizar la extracción de información (y muchas otras tareas del procesamiento de lenguaje natural) no son aún inteligentes y no entienden realmente el contenido de los documentos. Para la resolución de este tipo de problemas se basan, en general, en el reconocimiento de patrones lingüísticos y la comparación de cadenas de texto.

Algunas de las aplicaciones de la extracción de información son la generación de metadatos para la publicación de documentos en Internet (por ejemplo, documentos XML etiquetados de manera automática), agrupamiento y ordenación de resultados con respecto a conceptos clave, resumen automático de documentos e identificación de eventos (por ejemplo para saber cuándo el mandatario de algún país hizo una promesa importante que se exige sea cumplida).

Sin embargo, queda la pregunta de cuál es la diferencia entre recuperación y extracción de información. A continuación, se utiliza un ejemplo para mostrar esta diferencia.

---

[16] En el capítulo 3 se observará que en este trabajo se utilizan ambos enfoques durante las distintas etapas del desarrollo.

[17] Documentos en "texto plano" como este mismo trabajo. En los que no existe ningún tipo de marcaje que delimite los campos del documento, como ocurre en una base de datos (documento estructurado) o en un documento XML (documento semiestructurado).

Para ilustrar la recuperación de información, considérese una aplicación (digamos Google[18]) cuyo objetivo es obtener documentos con base en la búsqueda de ciertas *palabras clave*, para este ejemplo se utilizarán "*procesamiento*, *lenguaje* y *natural*"[19]. Al hacer una solicitud por medio de este motor de búsqueda, se obtienen aproximadamente 549,000 documentos que contienen estas tres palabras[20]. Sin embargo, el hecho de que aparezcan ahí, no implica que en dichos documentos se encuentre la definición de PLN o que, al menos, lo aborden como tema principal. La recuperación de información busca, como principal objetivo, obtener documentos que contengan cierta información solicitada sin tomar en consideración la situación de dicha información al interior del documento (lo cual, como el mismo Google lo ha demostrado, no está destinado a generar malos resultados).

Por otro lado, la extracción de información no tiene como objetivo encontrar documentos, sino hallar en su interior la información útil para el usuario. Para describirla, se utiliza como ejemplo la extracción de contextos definitorios[21]. Teniendo como origen de la búsqueda *procesamiento de lenguaje natural*, lo que se espera obtener de un sistema de extracción de información (y en este caso en particular de contextos definitorios) no es un conjunto de documentos, sino un conjunto de definiciones de esta disciplina[22].

Un ejemplo de la salida esperada de un sistema de extracción de información (que obtenga definiciones de términos) para la búsqueda *procesamiento de lenguaje natural* sería:

> El **Procesamiento de Lenguaje Natural** (PLN) es una subdisciplina de la Inteligencia Artificial y la rama ingenieril de la lingüística computacional. El PLN se ocupa de la formulación e investigación de mecanismos eficaces computacionalmente para la comunicación entre personas o entre personas y máquinas por medio de lenguajes naturales. El PLN no trata de la comunicación por medio de lenguajes naturales de una forma abstracta, sino de diseñar mecanismos para comunicarse que sean eficaces computacionalmente —que se puedan realizar por medio de programas que ejecuten o simulen la comunicación—. Los modelos aplicados se enfocan no sólo a la comprensión del lenguaje de por sí, sino a aspectos generales cognitivos humanos y a la organización de la memoria. El lenguaje natural sirve sólo de medio para estudiar estos fenómenos.[23]

En diversas tareas, resulta de mayor utilidad obtener solamente un bloque de texto con la información requerida (sean contextos definitorios o cualquier otra cosa) que un conjunto de documentos.

La extracción, tanto de términos como de contextos definitorios, es sólo una de las aplicaciones de la extracción de información y puede ser utilizada en el desarrollo de herramientas lexicográficas, en la delimitación de glosarios, en el análisis diacrónico de la lengua (para encontrar el momento en que un término surge en una disciplina, sólo por dar un ejemplo), para mejorar la clasificación automática de documentos, para generar ontologías que den soporte al web semántico, sólo por dar algunos ejemplos.

El presente trabajo se encuentra inmerso en un proyecto de extracción de contextos definitorios, tema que se aborda a continuación.

---

[18] www.google.com

[19] Debido a que, como primera etapa, Google elimina las palabras que no tienen un significado relevante para la búsqueda, como las preposiciones.

[20] Búsqueda realizada el 6 de diciembre de 2006.

[21] Los cuales serán abordados en el siguiente apartado.

[22] No se presenta ningún ejemplo particular debido a que en este momento nos encontramos trabajando en el tema.

[23] www.wikipedia.org

1.3 Extracción automática de términos en contextos definitorios

Si la extracción de términos se hace, no sobre un corpus general, sino sobre un corpus que contenga sólo contextos definitorios, será posible hallar fácilmente al término que se está definiendo y a los otros términos que son utilizados para este fin (primitivos).

Para el desarrollo de este tipo de aplicaciones, es necesario contar con un corpus de contextos definitorios, que contenga términos asociados con sus definiciones.

Los contextos definitorios son fragmentos de texto que se encuentran al interior de textos especializados y que contienen información relevante que define a un término dado. Un contexto definitorio está conformado por un término y una definición ligados, la mayoría de las veces, por un verbo definitorio (como ser o definir) o por un marcador tipográfico (como ":").

No es necesario dar un ejemplo adicional de contexto definitorio, pues el primer enunciado del párrafo anterior, en el que se definen a los contextos definitorios, lo es. En este caso, *contextos definitorios* es el término, *son* es el verbo definitorio y *fragmentos de textos*[...] es la definición.

Existen cinco tipos de definiciones que pueden hallarse al interior de un contexto definitorio:

a) Analítica (genus + diferencia). El término es descrito por una categoría más extensa (el genus) para luego ser distinguido de los otros elementos pertenecientes a esa categoría.

genus

el pingüino es un ave incapaz de volar que habita en el hemisferio sur del planeta

diferencia

Como puede observarse, el genus es ave, la clase de animales a la que pertenecen los pingüinos y la diferencia con otras aves es que no puede volar y sólo habita en el hemisferio sur. Este tipo de definición es la base para los siguientes tipos.

b) Genus exclusivo. Solamente incluye el genus, sin proporcionar mayor precisión sobre cualquier diferencia con otros elementos de la misma categoría.

genus (exclusivo)

un disco duro es un dispositivo de almacenamiento

c) Sinonímica. El genus tiene una estrecha relación con el término, tanto que se puede considerar que entre ellos existe una relación de sinonimia.

genus (sinónimo)

un conjunto de números es una colección de números

d) Funcional. La diferencia está basada en la función del término.

el carburador es el encargado de hacer la mezcla de combustible y aire

d*iferencia (funcional)*

e) Extensional. En la diferencia se encuentran las partes que componen al término

el microprocesador cuenta con una unidad aritmética lógica, unidad de control y registros

d*iferencia (componentes)*

Entre las tareas en las que la extracción automática de contextos definitorios es útil se encuentran:

- El análisis lingüístico de definiciones. Para obtener la clasificación de las definiciones presentada anteriormente, fue necesario que expertos en lingüística hicieran un análisis de textos en los que existieran contextos definitorios, por lo que la segunda etapa, después de conformar el corpus de trabajo, fue la detección de contextos definitorios. Con una herramienta capaz de reconocer estos contextos de manera automática, el tiempo de análisis se vería reducido dramáticamente.
- La creación de herramientas lexicográficas. La extracción automática de definiciones puede ser utilizada en la terminografía (para la creación de diccionarios especializados) y para todo tipo de diccionarios, como los semasiológicos mencionados anteriormente.

La localización no sólo de términos de especialidad en los documentos, sino de aquellos que son definidos en el mismo, al interior de un contexto definitorio, resulta de gran relevancia, entre otras cosas, para la realización de análisis lingüísticos tales como la investigación sobre el momento de aparición de un término en un área en particular o la creación de aplicaciones lexicográficas (diccionarios). Además, puede ser una buena herramienta en el desarrollo de las ontologías que dan sustento al web semántico o la creación de glosarios e índices de manera automática.

La línea principal de investigación del Grupo de Ingeniería Lingüística (al interior del cuál se desarrolla esta tesis), es la elaboración de diccionarios onomasiológicos [Sierra y McNaught 2000] que permiten al usuario expresar un concepto en lenguaje natural con el objetivo de obtener el término asociado a ese concepto. Para su desarrollo, la creación de herramientas de extracción de contextos definitorios y de términos es de gran importancia.

En el siguiente capítulo, se presenta un panorama del estado del arte de la extracción automática de términos.

Capítulo 2: Panorama actual de la extracción automática de términos

Donde se muestran los distintos enfoques que han sido abordados para la extracción de términos (lingüístico, estadístico e híbrido) y se muestran algunos de los extractores desarrollados hasta ahora.

La traducción automática, la indización de documentos, la recuperación y extracción de información, la generación de texto y la generación de herramientas lexicográficas son algunas de las tareas en las que la extracción automática de términos en documentos especializados puede ser de utilidad.

Sin embargo, la recopilación de dichos términos suele ser muy costosa, principalmente en cuestión de tiempo. Sólo por dar un ejemplo, cuando un lexicógrafo desea crear un diccionario especializado, un diccionario de términos, debe enfrentarse no sólo con la bibliografía, sino incluso con los especialistas en el tema que desea abordar, lo cual implica invertir una cantidad considerable de tiempo y dinero.

Las entradas para los diccionarios semasiológicos son, para el caso de diccionarios de especialidad, sus términos. La detección automática de los términos facilita mucho el trabajo de recopilación.

Otro ejemplo interesante es el de la taxonomía en la que se basa Yahoo[24]. En este sitio, la información se encuentra categorizada dentro de una taxonomía por temas como autos, música o tecnología. Dicha categorización sería imposible sin la ayuda de métodos automáticos de indización de documentos en los que la detección de términos puede resultar de gran utilidad.

Por ello, se ha visto la necesidad de crear herramientas computacionales que automaticen la obtención de términos en documentos especializados.

El desarrollo de los extractores automáticos de términos se ha basado principalmente en dos enfoques: el lingüístico y el estadístico. Un tercer enfoque, conocido como híbrido, se basa en la combinación de los dos tipos de conocimiento para refinar los resultados.

Para realizar su tarea, los extractores de términos se valen de las llamadas *herramientas de procesamiento de corpus de bajo nivel* [Heid 1999], entre las que se encuentran las siguientes:

a) Tokenizadores. También conocidos como segmentadores de palabras, separan una cadena de caracteres en unidades significativas llamadas tokens[25]. Parece una tarea trivial de búsqueda de espacios, pero no se deben olvidar signos tales como puntos, comillas o paréntesis que llevan a la tokenización a ser una tarea un tanto más complicada. Cadenas como $3,500.00 no deberían ser separadas. Incluso, en idiomas como el francés es necesario descomponer palabras como l'homme (el hombre) y en otros como el alemán, una lengua aglutinante, las complicaciones crecen aún más.
b) Lematizadores. La lematización es un intento por buscar el lema o lexema de la forma derivada de una palabra [Manning y Shütze 2001]. Por ejemplo, el lema de palabras como *la* y *las* es *el*.
c) Etiquetadores morfosintácticos[26]. Estas herramientas dependen de los tokenizadores y su tarea es etiquetar cada elemento en un enunciado con su categoría gramatical, sea nombre, adjetivo, número, signo de puntuación, etc.[27] Más adelante se muestra un ejemplo de esta tarea.

Es importante señalar que, debido a que no ha sido posible desarrollar aún un sistema de extracción de términos que funcione perfectamente (es decir, que extraiga todos los términos de un documento sin obtener ninguno que en realidad no lo sea), se considera que la salida de un

[24] http://www.yahoo.com
[25] Jackson, et. al., op. cit., p.10
[26] También se les conoce como etiquetadores de partes de la oración, POS o simplemente POST (*Part of Speech Tagging*), por lo que a lo largo de este trabajo serán utilizados invariablemente los tres términos.
[27] Jackson, et. al., op. cit., p.12-13

extractor está conformada por *candidatos a término* en lugar de términos. Dependiendo del uso que se desee dar a esta lista, puede ser revisada a mano por un especialista o puede ser utilizada tal cual, aceptando el ruido que los errores en ella pudieran producir.

Cabré, et. al. [2001] hicieron una recopilación sobre los sistemas para la extracción automática de términos que existían a la fecha. El primer detector de términos que consideran es TERMINO [Plante y Dumas 1998], cuya versión original fue desarrollada en 1990.

Se presenta a continuación una breve reseña de algunos de los extractores desarrollados hasta este momento, divididos con base en el enfoque (sea lingüístico, estadístico o híbrido) en el que se basan[28].

2.1 Extractores de términos basados en lingüística

Cuando se habla del tratamiento de terminologías, en este caso su extracción, se habla de una tarea lingüística. Es por eso que la mayoría de extractores de términos se basa, al menos en alguna de sus etapas, en la explotación de conocimiento lingüístico para la adquisición de candidatos a término[29].

Los términos de las diversas áreas de especialidad siguen ciertos patrones sintácticos. En el texto aparecen combinaciones de palabras que tienen cierta categoría gramatical, las cuales tienen un patrón común típico de los términos de especialidad. La explotación del conocimiento sobre estos patrones resulta de gran utilidad en la detección automática de términos, ya que si se conoce la categoría gramatical de las palabras en un texto, la detección de candidatos a término puede reducirse a la detección de los patrones sintácticos que estos presentan en un área de especialidad determinada.

Para realizar esta detección es necesario que las palabras en el corpus de entrada, aquel del que serán extraídos los términos, tengan su categoría gramatical identificada por medio de etiquetas. Como ejemplo de este etiquetado, se presenta a continuación un enunciado obtenido del Corpus Lingüístico de Ingeniería [Medina, et. al. 2004]. Primero, se puede observar la versión original, sin etiquetas.

> El objeto del estudio fue el de analizar el comportamiento estático y dinámico de placas de vidrio de fabricación nacional, y, el de proponer un método para su análisis estructural.

Al aplicar el etiquetador morfosintáctico TreeTagger [Schmid 1994], desarrollado en la Universidad de Stuttgart, se obtiene un archivo con una palabra del texto por línea, que tiene asignada una etiqueta con su categoría gramatical. TreeTagger, como puede observarse, es capaz también de determinar el lema de la palabra; sin embargo, no todos los etiquetadores morfosintácticos proveen esta información[30]:

---

[28] Si bien, el estudio de Cabré fue utilizado aquí como base para determinar el conjunto de extractores que serán descritos, en varios casos se describen versiones más actualizadas de las que originalmente fueron analizadas en ese trabajo.

[29] En el apartado 2.2 se observará que el conocimiento lingüístico no es imprescindible en la extracción automática de términos.

[30] Debido a que el etiquetado morfosintáctico es una de las etapas para la extracción de términos en este trabajo, será abordado con mayor profundidad en el apartado 3.2.

El	ART	el	de	PREP	de
objeto	NC	objeto	fabricación	NC	fabricación
del	PDEL	del	nacional	ADJ	nacional
estudio	NC	estudio	,	CM	,
fue	VSfin	ser	y	CC	y
el	ART	el	,	CM	,
de	CSUBI	de	el	ART	el
analizar	VLinf	analizar	de	CSUBI	de
el	ART	el	proponer	VLinf	proponer
comportamiento	NC	comportamiento	un	ART	un
estático	ADJ	estático	método	NC	método
y	CC	y	para	PREP	para
dinámico	ADJ	dinámico	su	PPO	suyo
de	PREP	de	análisis	NC	análisis
placas	NC	placa	estructural	ADJ	estructural
de	PREP	de	.	FS	.
vidrio	NC	vidrio			

Luego de un estudio lingüístico a mano sobre un corpus de especialidad, es posible determinar un modelo de los patrones sintácticos que siguen los términos. En el fragmento de corpus anterior, el término "análisis estructural" presenta un patrón típico de los términos en español (*NC+ADJ*).

Es sobre estas etiquetas, que identifican la categoría gramatical de las palabras, que se hace la búsqueda de patrones que distinguen a los candidatos a términos de los otros sintagmas que pueden ser considerados del léxico común.

Algunos de los extractores que explotan conocimiento lingüístico para realizar su tarea tienen como primera etapa del proceso de extracción el etiquetado morfosintáctico del corpus de entrada. Otros requieren tener a la entrada un corpus previamente etiquetado.

A continuación se presentan algunos de los extractores basados en conocimiento lingüístico. Cabe señalar que estas técnicas son dependientes del lenguaje, pues los términos y frases en general se componen de manera distinta en cada lengua.

2.1.1 Lexter

Lexter [Bourigault 1992] es un extractor de términos para el procesamiento de documentos en francés que fue programado en lenguaje C. Fue desarrollado para mantener actualizado un tesauro de términos de la compañía *Electricité de France* cuyo principal objetivo era la indización automática de documentos.

Esta herramienta basa su funcionamiento en dos etapas principales: la de análisis y la de parseo. Como entrada para la primera fase, Lexter requiere que el corpus haya sido previamente analizado por un etiquetador morfosintáctico.

La primera etapa es el análisis por identificación de fronteras. En lugar de realizar una búsqueda directa de los términos en el corpus, Lexter utiliza lo que Bourigault ha llamado *conocimiento negativo*[31]. Lo que busca es identificar todas aquellas palabras cuya categoría gramatical no se espera encontrar en un término y que, por ende, son consideradas fronteras potenciales de los candidatos a término que serán identificados más adelante. Las palabras que son fronteras potenciales de términos tienen categorías como conjunciones, preposiciones y

[31] Bourigault, op. cit. p. 3

artículos.

Es en este momento en el que Lexter procede a la búsqueda de frases nominales[32] de la mayor longitud posible que se encuentren entre las palabras de frontera.

Estas frases nominales, las cuales son ya consideradas candidatos a término, son la entrada para la segunda etapa del extractor: el parseo (análisis sintáctico), debido a que las frases nominales halladas pueden estar compuestas por varios términos, como en el caso de la frase nominal "disque dur de la station de travail" (disco duro de la estación de trabajo), en la que "disque dur" y "station de travail" son dos términos diferentes [Bourigault 1994].

La etapa de parseo se basa en un conjunto de reglas que contienen la estructura sintáctica que se espera que tengan los términos en francés. La salida es un conjunto de candidatos a término que, previa validación por el experto, son integrados al tesauro para la indización de documentos.

Lexter ha evolucionado a una herramienta más completa denominada Syntex [Bourigault, et. al. 2005]. Sin embargo, el objetivo de ésta es mucho más amplio que la extracción de términos.

2.1.2 HEID[33]

Toca el turno a un extractor de términos basado en conocimiento lingüístico que requiere contar con un conjunto de términos del área de especialidad tratada.

Esta aplicación, desarrollada por Ulrich Heid [1999] en la Universidad de Stuttgart tiene como objetivo intermedio la extracción de candidatos a términos sobre documentos en alemán y como objetivo final la creación de glosarios bilingües y memorias de traducción[34] (combinando inglés y alemán) utilizados en la traducción de documentos asistida por computadora.

Un buen corpus de prueba es vital en el desarrollo de un extractor de términos. El corpus utilizado en este proyecto fue proporcionado por el departamento de traducción de la compañía Daimler Chrysler. Este corpus está conformado por manuales de mantenimiento de autos (tanto en alemán como en inglés) y tiene varios millones de palabras.

La idea en la que se basa este extractor es la identificación de sintagmas de una sola palabra que sean relevantes en el "sublenguaje especializado" estudiado. Este conjunto de sintagmas es luego utilizado como base para la obtención de sintagmas más complejos, que contengan palabras fuertemente relacionadas entre sí que pudieran ser consideradas como candidatos a términos.

Para ello, se basa en una interesante dependencia circular entre memorias de traducción y extractores de términos. Como ya se observó, los extractores de términos pueden ser útiles para la obtención de términos relevantes que pueden ser integrados en las memorias de traducción. Por

---

[32] Combinaciones del tipo *NC+NC* o *NC+ADJ*. Dichos patrones, como se verá en el capítulo 3.3, también son comunes en los términos en español.

[33] Si bien, en versiones actuales de este extractor, incluida la presentada aquí, la extracción no sólo se basa en conocimiento lingüístico, sino también en estadístico, en un principio era una herramienta puramente lingüística [Heid, et. al. 1996], es por ello que se ha optado por abordarlo en este apartado. Debido a que este extractor no ha sido bautizado, se nombra aquí con el nombre de su creador.

[34] Las memorias de traducción son programas para la manipulación de bases de datos paralelas en las que se relacionan diversas palabras y sus traducciones en otros idiomas. Debido a que el trabajo de un traductor suele centrarse en distintas áreas específicas, una memoria de traducción con la terminología especializada en cada idioma puede ser de gran utilidad. Un ejemplo de este tipo de herramientas es SDLx, para mayor información, consultar http://www.trados.com

otro lado, las palabras que ya se encuentran en estas memorias (en particular, aquellas que sean términos), pueden ser utilizadas como punto de partida para el reconocimiento de otros términos que incluso podrían ser buscados en otro de los idiomas con los que cuente la memoria de traducción.

Para realizar la detección de candidatos a sintagmas terminológicos, Heid se basa en:

- La clase de la palabra (nombre, preposición, verbo, etc.).
- Información acerca de la subcategorización de verbos, adjetivos y nombres.
- Información acerca de las preposiciones y conjunciones que son más frecuentes al interior de los términos multipalabra en cada una de las áreas de especialidad tratadas.

El proceso de extracción en Heid se resume a dos etapas principales. Primero, la extracción de candidatos a términos compuestos por una sola palabra para después, en la segunda etapa, obtener colocaciones y otros candidatos a término multipalabra.

El procesamiento lingüístico se divide en las siguientes etapas.

- Etiquetado morfosintáctico. El texto se etiqueta con la categoría gramatical y el lema de cada palabra.
- Se realiza la extracción de candidatos a término basada en el reconocimiento de patrones sintácticos y en el reconocimiento de lemas en listas de palabras con las que se cuenta previamente (las que se encuentran en las memorias de traducción).
- Con la lista de términos, previa validación del experto, se generan los nuevos diccionarios de términos.

Lo más interesante en el enfoque de este extractor son los tres distintos niveles en los que opera la detección de patrones:

- Sobre caracteres, para la identificación de palabras “especiales”, como abreviaciones.
- Sobre morfemas, que permiten la identificación de candidatos a término de una sola palabra basada en su lema. Con ello, es posible extraer todas las derivaciones de un morfema presentes en el corpus aunque no todas sean realmente términos.
- Sobre secuencias de palabras, etapa en la que se toman en cuenta las etiquetas morfosintácticas de cada uno de los componentes del candidato a término.

La determinación de los patrones que se buscarán para encontrar los candidatos a término se basa en la baja frecuencia de ocurrencia que tienen en textos no especializados y en que, si bien no ocurre que todas estas secuencias presentes en un corpus especializado sean términos, buena parte de ellos sí lo son.

Es aquí donde comienza la parte estadística del proceso de extracción. Debido a que irremediablemente la extracción de términos basada en patrones genera ruido, es necesaria una etapa posterior que determine el potencial de un candidato de ser realmente un término.

La etapa estadística se basa en las frecuencias relativas de aparición de los sintagmas candidatos a término. Ésta consiste en hacer una comparación entre la cantidad de ocasiones que una palabra ocurre en el corpus especializado con la cantidad de veces que ésta aparece en un

corpus no especializado[35]. Evidentemente, la posición del candidato en la lista es directamente proporcional a su frecuencia en el corpus especializado e inversamente proporcional a su frecuencia en el no especializado.

La lista de términos filtrada por medio de la comparación de frecuencia relativa (que contiene aquellos candidatos a término que hayan superado cierto umbral), es la que se proporciona como salida del sistema.

Heid se enfrenta al mismo problema que nosotros, la existencia de una amplia variedad de herramientas para la explotación de corpus en inglés y su escasez para documentos en otras lenguas, como el alemán o el español, ya que el alemán, al ser una lengua aglutinante, no tiene las mismas estructuras y morfemas que el inglés.

Si bien la mayoría de los extractores basan su funcionamiento, al menos en alguna de sus etapas, en conocimiento lingüístico, hay algunos basados puramente en estadística, en el conteo de cadenas, como podrá observarse a continuación.

2.2 Extractores de términos basados en estadística

Las palabras que ocurren con mayor frecuencia en un texto, sin importar si es especializado o no, son las palabras funcionales[36]. Además de ellas, en un corpus especializado existen otras palabras que suelen ocurrir frecuentemente y que poseen una fuerte relación con el área de especialidad a la que pertenecen. Es por ello que la aparición frecuente de un sintagma en un corpus[37], implica una alta probabilidad de que se trate de un término.

La frecuencia de sintagmas y otras técnicas estadísticas, como la de información mutua, que calcula matemáticamente la similitud entre dos conjuntos de palabras, pueden ser explotadas para la extracción automática de terminologías. Se presenta a continuación una breve reseña de uno de los extractores que basan su funcionamiento exclusivamente en conocimiento estadístico.

2.2.1 ANA

ANA [Enguehard 1993], cuyo significado es *Automatic Natural Acquisition*, es un extractor de términos totalmente distinto a los demás. No sólo no utiliza conocimiento lingüístico para la detección de candidatos, sino que además, no fue diseñado para procesar únicamente textos "formales" como tesis o artículos, sino para cualquier tipo de documentos, como entrevistas o ponencias transcritas. El único requisito es que se encuentren en un ámbito especializado.

Es debido a la existencia de neologismos y estructuras sintácticas incorrectas en este tipo de material que no consideran conveniente el uso de cualquier tipo de conocimiento lingüístico como diccionarios o herramientas de análisis sintáctico.

Este sistema desarrollado en Lisp requiere contar previamente con tres listas que contienen tres tipos distintos de palabras[38] (debido a que la generación de estas listas, que requiere conocimiento lingüístico, es ajena al proceso de extracción, se considera a ANA como

[35] Este análisis sólo es realizado con sintagmas de longitud 1.

[36] Preposiciones y artículos, entre otras.

[37] Dependiendo de la lengua tratada, dicho sintagma puede incluir palabras funcionales como la preposición *de* en español. Éste es el caso de términos como *estación de trabajo*.

[38] Desafortunadamente, la forma en que estas tres listas son generadas automáticamente, como en el material consultado se menciona, es descrita en la tesis de doctorado de Enguehard, la cual no fue posible conseguir.

un extractor puramente estadístico). Las listas de palabras son:

a) Lista de palabras funcionales, las cuales tienen poco contenido semántico, como artículos, preposiciones e incluso algunos verbos como *ser*. Esta lista cuenta con entre 60 y 100 palabras como *à* (a), *alors* (entonces), *car* (ya que), *dans* (en), *le* (el), *mais* (pero), *vraiment* (verdaderamente).
b) Lista de palabras esquemáticas, compuesta por algunas de las palabras funcionales debido a que pueden indicar una relación semántica entre dos palabras, como en el caso de *réseau d'ordinateurs* (red de computadoras)[39]. Cuentan con alrededor de diez de estas palabras, como *de* (de), *de la* (de la), *des* (de), *du* (del) y *en* (en).
c) Palabras base (*bootstrap*), conformada por un conjunto de términos del área de especialidad, los cuales constituyen el núcleo de conocimiento necesario para la detección de nuevos términos en los textos tratados. Al inicio, esta lista debe contar con alrededor de 25 términos y va creciendo al agregar los nuevos términos hallados.

Con estas tres listas, es posible comenzar con la etapa de aprendizaje. El hecho de que el sistema sea llamado natural se debe a que está basado en la observación del proceso de adquisición de léxico que tienen los seres humanos durante la niñez, en la que sin conocer ningún tipo de regla gramatical o contar con algún diccionario, se van generando frases cada vez más complejas y con mayor léxico. Tomando esta idea como base y con el principio de que "la ocurrencia frecuente de un evento es significativa" [Enguehard 1993], se considera que:

- Un evento es la ocurrencia de cualquier palabra en el texto (sin importar que exista en alguna de las tres listas con las que se cuenta o no).
- Dos eventos son considerados coocurrentes si son separados por *n* palabras funcionales o menos. La ocurrencia frecuente de eventos (coocurrencias de palabras) en el texto es la que determinará si una cadena es un término o no.

Las coocurrencias detectadas son de tres tipos:

a) Expresión, la coocurrencia de dos términos. Ej. ... la *terminal* del *servidor* es...
b) Candidato, la coocurrencia de un término y de una palabra separados por una palabra esquemática. Ej. ... la *computadora* del gerente ...
c) Expansión, la coocurrencia de un término y una palabra. Ej. ... los *servidores* internos ...

En resumen, el algoritmo de descubrimiento de términos tiene cuatro etapas:

1. Reducción. Se eliminan signos de puntuación, todas las letras son convertidas en minúsculas y el alfabeto se reduce a veintiséis letras, diez cifras (0-9) y el espacio.
2. Análisis léxico. Se identifican las palabras base que ocurren en el texto.
3. Obtención de coocurrencias. Todas las coocurrencias de los eventos identificados en el punto anterior son almacenadas.
4. Inducción de nuevos términos. Las coocurrencias que se presentan con mayor frecuencia

[39] Si bien, estas palabras pueden presentarse en contextos en los que no existe una relación semántica real entre las palabras vecinas como en el caso de "casa de madera", vale la pena tener este ruido debido a la productividad que representan.

en el texto, superando un umbral previamente establecido, son memorizadas como nuevos términos.

Una de las mayores ventajas de ANA es que, al no requerir conocimiento lingüístico (aunque como ya se vio, en la etapa previa al proceso de extracción lo requiere), es independiente de la lengua tratada. Este es el único extractor encontrado que se basa puramente en estadística.

Sin embargo, uno de los errores cometidos en este extractor es la eliminación de signos de puntuación, cuya función en un documento es romper las relaciones entre las palabras que se encuentran antes y después de ellos.

2.3 Extractores híbridos de términos

El hecho de que existan pocos sistemas basados exclusivamente en lingüística o estadística obedece a que, en general, la combinación de ambas técnicas genera mejores resultados. A continuación se presentan algunos sistemas más robustos en los que los dos enfoques se ven complementados.

Estos sistemas se valen de la aplicación en conjunto de los dos enfoques para refinar la salida. Logran localizar términos que por un sólo método no se hubieran detectado y, sobre todo, descartan candidatos a término que en realidad no lo son, lo que disminuye el ruido de manera significativa.

Se presentan a continuación tres de los principales extractores híbridos de términos que han sido desarrollados para distintas lenguas y aplicaciones.

2.3.1 ACABIT

ACABIT [Daille 2003], desarrollado para extraer términos en textos en francés, requiere de un corpus no sólo etiquetado morfosintácticamente sino también lematizado. Con dicho corpus como entrada, es capaz de conformar una lista de candidatos basada en patrones sintácticos y relaciones semánticas.

Durante la primer etapa, ACABIT realiza un filtrado de cadenas por medio del reconocimiento de patrones morfosintácticos previamente establecidos. Luego de contar con esta lista, las distintas variantes de cada candidato son agrupadas. La agrupación se hace no sólo con base en la lematización realizada previamente al corpus, sino con base en una estructuración conceptual.

La *estructuración conceptual* consiste en la búsqueda de adjetivos relacionales en los candidatos a términos. Si dos candidatos con estructura NC_1 +ADJ y NC_1+PREP+NC_2 respectivamente, contienen el mismo NC_1 y ADJ es una versión "adjetivada" de NC_2, como en el caso de *industrie de l'alimentation* (industria de la alimentación) e *industrie alimentaire* (industria alimenticia), se considera al adjetivo como relacional. Todos los adjetivos relacionales son almacenados en una lista.

Es con la lista de adjetivos relacionales que se comienzan a buscar relaciones de hiperonimia[40] entre los candidatos previamente adquiridos. Si un mismo adjetivo relacional se

[40] El significado de esta palabra, abarca a otras. Por ejemplo, perro es el hiperónimo de dutchhound, una variedad de perros.

encuentra en un conjunto de candidatos, se infiere que existe una relación de hiperonimia entre ellos. Si además dentro del candidato hay negaciones, se infiere que se trata de una relación de antonimia y también es agrupado.

Finalmente, se genera una lista de términos "representativos" que están ligados a todas sus variantes (homónimos, hiperónimos, antónimos). La etapa estadística se encarga únicamente de ordenar esta lista de candidatos con base en su frecuencia de ocurrencia en el corpus.

Si bien la etapa estadística de ACABIT puede no resultar de vital importancia para la detección de términos, en los siguientes dos casos se observará que ésta puede tener mucho mayor relevancia.

2.3.2. TermoStat

TermoStat [Drouin 2003] es una herramienta para la extracción de terminologías desarrollada en Perl. Originalmente fue diseñada para la extracción de términos en documentos en francés e inglés, pero actualmente cuenta también con versiones para el español e italiano[41].

Su principio básico, similar a lo señalado por Heid, es la consideración de que los términos están fuertemente relacionados con el área de especialidad a la que pertenecen. Sin embargo, TermoStat no explota recursos como diccionarios o memorias de traducción, sino que se basa en la comparación de un corpus especializado con uno de carácter general.

Por ello, TermoStat requiere de un corpus de entrada conformado por dos subcorpus: el corpus en el que se quiere hacer la búsqueda de términos, que es de carácter técnico y se conoce como *corpus de análisis*, y un corpus no técnico, el *corpus de referencia*[42], que sirve para determinar qué tan estrecha es la relación de una palabra con respecto al corpus técnico. La diferencia en el léxico que se utiliza en los dos corpus es explotada para la identificación de candidatos a términos.

La determinación de la especificidad de una palabra con respecto al corpus de análisis se basa en su frecuencia de aparición en el corpus de análisis con respecto a su frecuencia de aparición en el corpus de referencia. Dicha comparación es realizada por medio de la medida estadística de *distribución normal estándar*.

Además de la lista de candidatos obtenida (de una sola palabra), se conservan únicamente aquellas palabras cuya categoría gramatical sea nombre o adjetivo, ya que son las categorías más comunes de palabras con fuerte contenido semántico en los términos. Estas palabras son llamadas *pivotes lexicales especializados*.

Luego de la identificación de pivotes, es posible comenzar con la extracción final de candidatos a término. Todas las palabras del corpus que no se encuentren dentro de la lista de pivotes son consideradas las fronteras entre los posibles candidatos a término (como se puede inferir, este concepto de frontera está basado en el trabajo de Bourigault descrito anteriormente).

A continuación, se realiza la búsqueda de candidatos con base en su patrón morfosintáctico. Dichas cadenas candidatos son secuencias de ADJ+NC (para el caso del inglés) en las que al menos uno de sus elementos se encuentran en la lista de pivotes lexicales.

Una de las ventajas que implica esta técnica sobre otras es la capacidad de extraer

---

[41] Esta herramienta puede ser utilizada a través de su sitio Web en http://olst.ling.umontreal.ca/~drouinp/termostat_web/?lang=fr_CA

[42] Dicho corpus general puede estar compuesto, por ejemplo, por documentos provenientes de artículos de periódico o novelas.

términos de una sola palabra, lo que con otros métodos implica la generación de mucho ruido.

2.3.3. TerMine[43]

TerMine [Frantzi, et. al. 2000] es el extractor de términos del Centro Nacional de Minería de Textos en Manchester, Inglaterra y fue desarrollado para la extracción de términos multipalabra en textos biomédicos en inglés. Este extractor se basa en el algoritmo C-value/NC-value compuesto por dos etapas que funcionan en cascada, una lingüística-estadística y una estadística.

La primera parte de la etapa conocida como C-value es la única de carácter lingüístico y consiste en un filtrado de candidatos a término sobre un corpus etiquetado morfosintácticamente. Como en la mayoría de los extractores vistos anteriormente, cuenta con un conjunto de patrones representados por reglas cuyas ocurrencias son buscadas en el corpus.

La segunda parte del C-value, la estadística, considera la longitud de cada uno de los candidatos obtenidos (en número de palabras) y su frecuencia de aparición tanto en el corpus completo como al interior de otros candidatos de mayor longitud para generar una lista ordenada. Dicho valor es llamado C-value y representa el potencial de que cada uno de los candidatos sea realmente un término.

La segunda etapa, la fase NC-value, se basa en el análisis del contexto de cada uno de los candidatos hallados en la lista de C-value. Considera que las palabras que ocurren en la vecindad de los candidatos a término que están en la parte más alta de la lista ordenada por medio de C-value son palabras que tienden a ocurrir, en ocasiones con una alta frecuencia, junto a términos reales, por lo que su ocurrencia en la vecindad de cualquier otro de los candidatos, sin importar su posición en la lista, es favorable para su potencial.

Como ya se señaló, una explicación detallada del algoritmo C-Value/NC-Value, incluida su adaptación para la extracción de términos en documentos en español, es ofrecida a lo largo del capítulo 3.

Una de las principales ventajas de esta metodología es que, además del etiquetado morfosintáctico realizado previamente al corpus, no requiere de ningún otro recurso para funcionar, ni diccionarios, ni corpus adicionales ni una lista inicial de términos.

2.3.4. Señalamientos finales sobre el estado del arte

Para cerrar este capítulo, se presentan las observaciones finales alrededor del estado del arte de la extracción automática de términos. Como se ha observado, una gran diversidad de técnicas ha sido explotada en búsqueda de la extracción certera de términos[44].

Si bien, han sido clasificados en tres categorías principales, se ha podido observar que es prácticamente imposible generar un extractor de términos sin contar al menos en una mínima etapa con conocimiento tanto estadístico como lingüístico.

Debido a que para comparar la calidad de la salida de cada uno de los extractores sería necesario realizar experimentos sobre el mismo corpus y bajo las mismas condiciones (lo cual es

[43] El algoritmo C-Value/NC-Value, elemento medular de TerMine, es en el que se basa el presente trabajo, por lo que se ofrece una breve descripción de él ahora para abordarlo con mayor profundidad en el capítulo tres.

[44] Medidas comunes para medir esta certeza son los llamados precisión y recuerdo (precision y recall). Estas medidas serán descritas en el capítulo 4 en el que se presenta la evaluación de la herramienta desarrollada.

imposible en principio porque los distintos extractores requieren de un conjunto de recursos muy variado), dicha comparación no será realizada.

En lugar de eso, se presenta en la tabla 2.1 un resumen de las propiedades de cada uno de los extractores, incluyendo la o las lenguas que son capaces de procesar, las principales ideas en las que basan su funcionamiento y algunas observaciones sobre ellos. Debe recordarse que el enfoque de Lexter y Heid es lingüístico mientras que el de ANA es estadístico y el de ACABIT, TermoStat y TerMine puede considerarse híbrido.

A excepción de la versión de TermoStat para el español, poco se ha trabajado en la extracción de términos en este idioma. Además, como se ha podido observar, la detección de términos al interior de contextos definitorios para el desarrollo de herramientas lexicográficas y de pregunta respuesta no ha sido suficientemente explotada.

Extractor	*Lengua(s)*	*Principal idea explotada*	*Observaciones*
Lexter	francés	La detección inicial de palabras de frontera para la subsecuente identificación de frases nominales entre ellas que, luego de un filtrado basado en patrones sintácticos, son los candidatos a término.	Computacionalmente el proceso es muy eficiente pues al detectar el conjunto de palabras funcionales, la cantidad de palabras a analizar sintácticamente decrece de manera importante.
Heid	alemán	La ubicación de términos complejos con base en una primer lista de términos de longitud uno y su expansión a términos de mayor longitud con base en el conocimiento de las combinaciones de categorías gramaticales comunes en los términos.	Si bien no realiza una extracción de términos bilingüe por sí mismo, Heid puede ser utilizado para ello. La relación circular entre las memorias de traducción y diccionarios con el extractor puede ser muy benéfica. El problema es que, en general, no se cuenta con estos recursos adicionales.
ANA	multilenguaje	La explotación de listas de distintos tipos de palabras (que se espera que aparezcan en términos o no) son utilizadas como semilla para la detección "natural" de candidatos a término.	Requiere contar previamente con una lista de términos y palabras funcionales y esquemáticas, pero contando con dicha lista puede ser utilizado para cualquier lengua.
ACABIT	francés	Se basa en la agrupación de los términos obtenidos por medio de un análisis sintáctico que se hace con base en la inferencia de relaciones semánticas (por hiperonimia por ejemplo).[45]	Además de etiquetado morfosintáctico, requiere de un lematizador y de un reconocedor de raíces para establecer las relaciones entre términos. Sin embargo, estas relaciones pueden ser de gran utilidad para la eliminación de redundancias y la expansión de términos.
TermoStat	francés, inglés, español e italiano	Los términos tienen una fuerte relación con el área a la que pertenecen, por lo que la probabilidad de que aparezcan en un corpus especializado es mucho mayor a la de aparecer en un corpus no especializado.	Requiere de un corpus general además del que se desee analizar. Sin embargo, el enfoque permite obtener candidatos a término de longitud uno sin tener que lidiar con grandes cantidades de ruido.
TerMine	inglés	La frecuencia de aparición de una cadena con un patrón sintáctico típico de un término no sólo dentro del corpus, sino al interior de otros candidatos de mayor longitud es de gran relevancia para determinar si una cadena es en verdad un término o no.	Su ventaja es que además de un corpus previamente etiquetado, no requiere de ningún otro recurso. Sin embargo, su debilidad es que no es capaz de detectar candidatos a término de una sola palabra.

Tabla 2.1: Características de los extractores

[45] Cabe señalar que en realidad el programa no se desempeña con base en conocimiento semántico real, simplemente realiza suposiciones con base en los lemas y raíces de las palabras.

Capítulo 3. Metodología

Donde se analiza la técnica utilizada para la extracción automática de términos y se muestran las modificaciones que han sido realizadas al algoritmo original.

En este capítulo se hará una descripción detallada del trabajo práctico desarrollado durante esta investigación.

Como puede observarse en el capítulo anterior, se ha buscado realizar la tarea de extracción automática de términos por medio de diversas técnicas, las cuales pueden ser englobadas dentro de tres grandes enfoques: el lingüístico, el estadístico y el híbrido. El enfoque que mejores resultados ha dado es el híbrido, que implica el uso de técnicas lingüísticas y estadísticas para la extracción de términos.

Luego de realizar el análisis sobre las diversas técnicas para la extracción automática de términos, se ha optado por utilizar una versión modificada del algoritmo C-value/NC-value utilizado en TerMine[46]. Los motivos para elegir este algoritmo se presentan a continuación:

- Se ha observado que para la implementación de la etapa lingüística de C-value sólo se requiere modificar las reglas para la detección de patrones sintácticos, de las que fueron diseñadas para el inglés a reglas para la extracción en español.
- La generación de la lista de paro puede resumirse a un conteo de palabras sobre el mismo corpus. Las palabras de esta lista serán seleccionadas del grupo de palabras con mayor frecuencia en el corpus.
- Debido a que la etapa estadística de C-value se basa en la longitud de cadenas (en número de palabras) y su frecuencia de ocurrencia en el corpus, su implementación no requiere mayores modificaciones.
- La entrada para el extractor es simplemente el texto que se desea analizar. No es necesario recurrir a otros recursos tales como corpus adicionales, diccionarios o tesauros. Sólo se requiere de un etiquetador morfosintáctico.
- El desempeño de la versión para el inglés es satisfactorio ya que presenta una precisión de 75.70%[47].
- Existe la posibilidad de tener comunicación con los creadores de la herramienta original para el inglés.
- Algunas de las debilidades que presenta el algoritmo pueden ser atacadas. Una de estas debilidades es que no es capaz de extraer términos de una sola palabra[48].

A continuación se hace una descripción de los corpus de prueba que han sido utilizados para el desarrollo de la herramienta. Más adelante se abordarán las distintas etapas del proceso de obtención de la terminología hallada en un corpus.

Debido a que la etapa de etiquetado morfosintáctico se considera previa al proceso de extracción, ésta se aborda de manera individual en el apartado 3.2. El trabajo de determinación de reglas lingüísticas para la detección de términos en español es abordada en el apartado 3.3. En el apartado 3.4, se describe con detalle el algoritmo C-Value/NC-value para finalmente abordar, en el apartado 3.5, las adaptaciones y mejoras que se le han hecho.

La realización de pruebas y la evaluación de la herramienta han sido dejadas para el

---

[46] http://www-tsujii.is.s.u-tokyo.ac.jp/termine/

[47] La precisión es la parte de los resultados obtenidos que es realmente relevante. En este caso, aquellos candidatos a términos obtenidos que realmente son términos. Este concepto será dado de manera más detallada en el cuarto capítulo, ya que es utilizado para la evaluación del extractor.

[48] Esta limitación está en cierta forma justificada debido a que, según los análisis lingüísticos realizados en los documentos de biomedicina utilizados para el desarrollo de TerMine, los términos de una sola palabra son prácticamente inexistentes.

capítulo 4.

3.1 Los corpus

Contar con un corpus de prueba resulta de gran importancia en este tipo de desarrollos, ya que de él se obtienen los patrones lingüísticos de los términos y sobre él mismo se realizan las pruebas y evaluación. De manera precisa, un corpus lingüístico es la "recopilación de un conjunto de textos de materiales escritos y/o hablados para realizar ciertos análisis lingüísticos; estos textos deben ser representativos y se recogen según criterios lingüísticos para poder ser utilizados en el análisis" [Sierra 2005][49].

McEnery y Wilson [1996] señalan entre las características principales con las que un corpus lingüístico debe contar:

- La representatividad tanto de la lengua como del área de estudio que se desea estudiar. Se puede tratar de un corpus general, sin algún área o tema en particular o de uno especializado.
- Manejable por computadora, en un sentido amplio, pues no basta con que se encuentre en forma de archivo electrónico, sino que debe ser posible acceder a sus elementos constituyentes (las palabras) por medio de una interfaz o programa.

Hay una gran cantidad de parámetros con los que se puede clasificar a un corpus. Entre ellos está su origen (oral o textual), su lengua (monolingüe o multilingüe) y, lo que para nuestro caso tiene especial relevancia, su especificidad (general, especializado, sincrónico, diacrónico)[50].

Como es de esperarse, en el caso de la extracción de términos se requiere contar con un corpus especializado, textual y, en nuestro caso, monolingüe[51]. Si la investigación así lo requiere, el corpus podría ser diacrónico (si se quiere saber, por ejemplo, el momento en que un término fue creado en un área de especialidad o adoptado de otra, lo que podría implicar su invención). Aquí se usa uno de carácter sincrónico.

Además, la tarea de extracción de términos, sobre todo cuando buena parte de la discriminación de candidatos se basa en información estadística (tomando en cuenta principalmente su frecuencia de aparición en el texto), es conveniente contar con un corpus de varios cientos de miles de palabras. En los experimentos que fueron realizados durante el desarrollo de TerMine, el corpus de prueba tenía 810, 719 palabras[52].

Durante el desarrollo de este trabajo se ha contado con dos corpus de prueba: el Corpus Lingüístico de Ingeniería[53] del Grupo de Ingeniería Lingüística (UNAM) y el Corpus de Informática en Español [L'Homme y Drouin 2006] del *Observatoire Linguistique Sense-Texte* (UdeM). A continuación una descripción de cada uno de ellos.

---

[49] Obtenido de la página web del curso "Lingüística de corpus", http://www.iling.unam.mx/CursoCorpus/default.html

[50] Para mayor información sobre corpus lingüísticos, incluyendo sus características, clasificación y métodos de creación, se recomienda consultar http://www.iling.unam.mx/CursoCorpus/default.html

[51] Como ya se observó, en el extractor Heid se utiliza un corpus bilingüe (inglés y alemán).

[52] Frantzi, et. al., op. cit., p. 6.

[53] Medina, et. al., op. cit.

3.1.1 El Corpus Lingüístico de Ingeniería

El CLI (siglas de Corpus Lingüístico de Ingeniería) está conformado por un conjunto de documentos de distintas áreas de la ingeniería, como la ambiental y la mecánica. En su mayoría, dichos documentos están escritos en español de México, pues contiene también un pequeño conjunto de documentos en español peninsular. La gama de documentos va desde tesis doctorales y artículos en revistas hasta reportes de estudiantes y profesores de ingeniería.

El corpus está compuesto por 23 archivos que tienen en total 274,672 palabras. La tabla 3.1 muestra una descripción de los documentos subdivididos con base en las subáreas a las que pertenecen.

El CLI se encuentra dividido en archivos de texto plano sin ningún tipo de etiquetado o anotación especial. El siguiente extracto es una muestra de él:

> Para obtener la curva carga-desplazamiento, se aplicó una carga uniformemente distribuida en un período de tiempo de dos horas, y después de doce horas, se descargó en el mismo período de tiempo que en el que se cargó (dos horas). Los incrementos de carga que se utilizaron fueron de 44.8 Kg. y 104 Kg., aplicados uniformemente, y en ese orden, hasta llegar a 687.9 Kg.

Para las pruebas se ha optado por utilizar un subcorpus del CLI con un área de especialidad más específica: ingeniería mecánica. Esto se debe en principio, a que cuenta con una buena cantidad de palabras (en total 10,191), además de estar compuesto por cinco documentos de tamaño similar que han sido escritos por cinco autores distintos, lo que le da buena representatividad.

subtema	*título del archivo*	*documento*	*palabras*
ingeniería de sistemas	Utilidad y dotación óptima de la biblioteca pública de México	informe	5,910
ingeniería lingüística	Resolución automática de la anáfora indirecta en el Español	tesis doctoral	45,049
ingeniería ambiental	Transformación genética de la especie Galphimia glauca CAV. (MALPIGHIACEAE) por Agrobacterium rhizogenes para la producción de NOR-FRIEDELANOS	tesis doctoral	22,540
	Producción de saponinas antifúngicas en biorreactores	tesis doctoral	26,675
	Estudio del efecto de algunos reguladores del crecimiento vegetal en la inducción a callos de Ipomoea murucoides Roem. et Schult (CONVOLVULACEAE): Actividad insecticida y la relación con sus características estructurales	tesis de maestría	30,679
	Control, medición y tratamiento de partículas emitidas por los motores diesel	tesis de maestría	1,988
	Estudios para mejorar la confiabilidad del funcionamiento del sistema Cutzamala	informe	13,669
	Establecimiento del perfil fitoquímico, propiedades biológicas y condiciones de cultivo in vitro de células y raíces transformadas de las plantas mexicanas Polypodium leucotomos, Marsdenia lanata, Marsdenia zimapanica y Cordia morelosana.	plan de trabajo	5,699
energía	¿Abandonar la energía nuclear?	art. de revista	838
	La energía nuclear es la única solución ecológica	art. de revista	1,525
	¿Hay que apostar de nuevo por la energía nuclear?	art. de revista	999
	Esperando la fusión	art. de revista	951
ing. de software	RDM: Arquitectura Software para el Modelado de Dominios en Sistemas Informáticos	tesis doctoral	66,596
ingeniería mecánica	Transmisión para un pequeño tracor agrícola	reporte	2,056
	Comportamiento estático y dinámico de placas de vidrio	reporte	1,782
	Dispositivos anticontaminantes para motores de combustión interna	reporte	2,883
	Análisis para la instrumentación de un enganche de tres puntos	reporte	1,848
	Módulo para cosecha no selectiva de tunas	reporte	1,622
geotecnia	Evaluación de los problemas geotécnicos planteados para la construcción del metro a gran profundidad	reporte	10,352
ingeniería eléctrica	Aparato de corte simple cíclico	reporte	3,503
ingeniería industrial	Metodología para el diseño de estrategias de transportes de cargas en el marco de programas regionales de desarrollo	informe	20,569
ingeniería sísmica	Bases de confiabilidad estructural para determinar espectros de sitio para diseño sísmico en el DF	informe	6,939

Tabla 3.1: Los archivos que componen al CLI

3.1.2 El corpus de informática en español

Este corpus [L'Homme y Drouin 2006] ha sido recabado en la Universidad de Montreal y tiene como principal objetivo el desarrollo de la versión en español de la herramienta DicoInfo, "el diccionario fudamental sobre informática e Internet"[54]. Los documentos que lo componen fueron extraídos de revistas electrónicas como Windows TI[55] y enciclopedias electrónicas tales como Wikipedia en español[56].

La clasificación del corpus se ha hecho con base en las subáreas de especialidad con las que cuenta. Dicha distribución se presenta en la tabla 3.2[57].

Debido a que el web es la fuente principal de los documentos en este corpus, la variedad del español que prevalece es la de España[58]. En este caso, se ha optado por utilizar el subcorpus de Hardware debido a que la cantidad de palabras con que cuenta es una de las más altas, además de que la cantidad de archivos que tiene implica una alta representatividad pues no están escritos por un sólo autor. Un ejemplo del texto hallado en este corpus es:

> La CPU está compuesta por: registros, la Unidad de control, la Unidad aritmético-lógica, y dependiendo del procesador, una unidad en coma flotante. Cada fabricante de microprocesadores tendrá sus propias familias de estos, y cada familia su propio conjunto de instrucciones. De hecho, cada modelo concreto tendrá su propio conjunto, ya que en cada modelo se tiende a aumentar el conjunto de las instrucciones que tuviera el modelo anterior.

[54] http://olst.ling.umontreal.ca/dicoinfo/

[55] http://www.windowstimag.com/

[56] http://es.wikipedia.org/

[57] Debido a que este corpus no pertenece a nuestro grupo de investigación, sino al *Équipe Éclectik* de la Universidad de Montreal, no se proporcionan tantos detalles sobre él como en el caso del CLI.

[58] Si bien no ha sido comprobado, se puede asumir que la cantidad de sitios web en España es mayor a la de los otros países de habla hispana en el mundo (al menos en computación). Un experimento empírico proporciona pruebas de ello: si se buscan en Google cadenas como "lenguaje de programación" o "editor de textos" y la búsqueda se limita a un país de habla hispana a la vez, se observará que la cantidad de páginas españolas recuperada es superior a la de otros países (el 10 de enero de 2007 eran 927,000 con sufijo *.es* contra 337,000 con sufijo *.mx*). Por supuesto, en este experimento no se consideran variaciones léxicas (como el caso de computadora y ordenador), pero es una buena aproximación de la realidad.

tema	*archivos*	*palabras*
Derechos de autor	4	2,137
Editores de texto	3	1,065
Empresas de informática	4	4,111
Google	1	564
Hardware	219	142,981
Inteligencia artificial	2	1,428
Internet	59	51,679
Programacion	8	4,183
Redes informaticas	59	47,513
Seguridad informatica	54	78,601
Sistemas operativos	19	40,711
Software	144	126,618
TOTAL	576	501,591

Tabla 3.2: Temas de los documentos del corpus de informática

Al igual que en el caso del CLI, el Corpus de Informática en Español no cuenta con ningún tipo de etiquetado o preprocesamiento, por lo que como primera etapa para la extracción de términos se requiere realizar el etiquetado morfosintáctico del corpus.

3.2 Etiquetado morfosintáctico

Ahora se presenta una breve descripción de la etapa previa a la extracción automática de términos: el etiquetado de partes de la oración. Además, se muestran dos etiquetadores que fueron contemplados y evaluados para realizar esta tarea: TreeTagger y Freeling[59].

Como se señaló en el capítulo anterior, el etiquetado de partes de la oración (o morfosintáctico) es un proceso que consiste en asignar a cada palabra de un texto una etiqueta con la categoría gramatical a la que pertenece.

La complejidad de esta tarea está en decidir qué etiqueta se debe asignar a las palabras cuya categoría es ambigua. Por ejemplo, obsérvese el caso de la palabra *bajo*, que dependiendo del contexto en el que se encuentre puede tener varias categorías gramaticales:

- Nombre, “la armonía entre *bajo* y guitarra en esta canción es maravillosa”
- Verbo, “en un momento *bajo*, espere por favor”
- Adjetivo, “tome su lugar al lado del hombre *bajo* que está a la izquierda”
- Preposición, “el castor está *bajo* el árbol de maple”

La desambiguación de categorías gramaticales es una tarea complicada que frecuentemente se resuelve por medio de métodos estadísticos que determinan cuál es la etiqueta

[59] En el apéndice 1 “Etiquetas de las herramientas para el etiquetado de partes de la oración”, se muestra el significado de cada una de las etiquetas utilizadas por estas herramientas.

que le corresponde a cada palabra con base en el entorno en el que ésta se encuentra. La probabilidad de que un nombre aparezca después de una preposición, como en el primer caso, es mayor que la de aparecer luego de un verbo, como en el cuarto[60].

En este trabajo se han analizado dos etiquetadores: TreeTagger, desarrollado en la Universidad de Stuttgart y la herramienta Freeling de la Univesidad Politécnica de Catalunya.

Cabe señalar antes de entrar de lleno en el análisis de los etiquetadores, que ambos utilizan técnicas lingüísticas y probabilísticas muy similares. Sin embargo, el primero da mucho mayor peso a la parte probabilística mientras que el otro lo hace con la parte lingüística.

3.2.1 TreeTagger, un etiquetador morfosintáctico probabilístico

TreeTagger [Schmid 1994] es una herramienta de etiquetado de partes de la oración y lematización desarrollada en el Instituto para el Procesamiento de Lenguaje Natural de la Universidad de Stuttgart[61]. Esta herramienta puede ser utilizada para una buena cantidad de lenguas como el alemán, inglés, francés, ruso, griego y, por supuesto, el español.

Muchos etiquetadores basados en probabilidad, como TreeTagger, definen las etiquetas de un trigrama de palabras de manera recursiva con base en la estimación de la probabilidad de transición $p(t_n|t_{n-2}t_{n-1})$, que es la probabilidad de que una palabra p_n tenga una etiqueta t con base en las etiquetas de las palabras previas t_{n-2} y t_{n-1}.

TreeTagger determina esta probabilidad de transición por medio de un árbol de decisión binario (de ahí el nombre del etiquetador) como el que se muestra en la figura 3.1[62]. La probabilidad de las etiquetas en un trigrama es determinada, como en todo árbol de decisión, por la hoja que es alcanzada luego de recorrer un camino determinado. Este árbol de decisión es construido de manera recursiva a partir de un conjunto de trigramas de entrenamiento[63].

TreeTagger cuenta con un lexicón de palabras (generado a partir de un corpus de entrenamiento previamente etiquetado y validado) en el que se incluye la probabilidad de que una palabra tenga una etiqueta determinada. Dicho lexicón está dividido en tres partes: la forma completa de la palabra, un sufijo y la etiqueta por defecto.

Para la versión en inglés, el lexicón estuvo compuesto por aproximadamente dos millones de palabras provenientes del *Penn Treebank Corpus* de la Universidad de Pennsylvania[64]. En el caso del español, el corpus utilizado fue el *Crater Multilingual Aligned Annotated Corpus*, de la Universidad de Lancaster[65].

Cada una de las palabras en un texto es buscada primero en la lista de formas completas. Si se encuentra ahí, el vector de etiquetas con sus probabilidades es obtenido, de lo contrario, se busca su sufijo para obtener un vector similar. Si éste tampoco se encuentra, se devuelve un valor por defecto. Es la combinación de este vector de probabilidades con el vector obtenido por medio del árbol de decisión la que define la etiqueta que será asignada a cada palabra.

---

60 En una muestra tomada al azar del corpus de informática con 13,334 palabras, la combinación de preposición y nombre ocurrió 924 ocasiones mientras que la de verbo y nombre se presentó solamente 157 veces.

61 http://www.ims.uni-stuttgart.de/

62 Schmidt, op. cit., p. 3

63 Para observar la explicación detallada de este procedimiento, véase, Schmidt, op. cit. pp. 2-4

64 http://www.cis.upenn.edu/~treebank/

65 http://www.comp.lancs.ac.uk/linguistics/crater/corpus.html

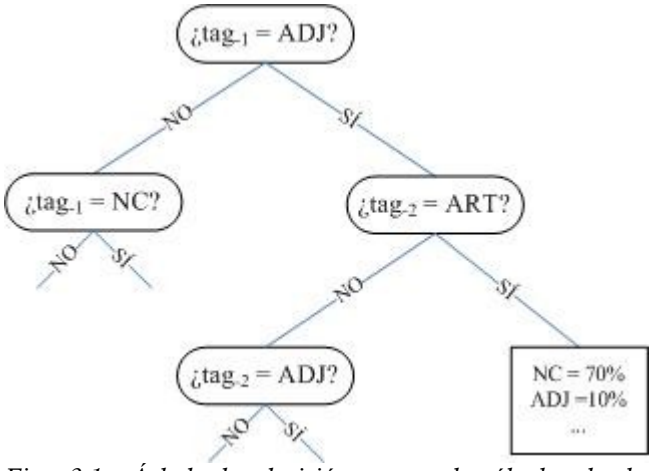


Fig. 3.1: Árbol de decisión para el cálculo de la probabilidad de transición

Un ejemplo del etiquetado de TreeTagger se puede ver a continuación (primero se muestra el texto original[66] y luego el texto etiquetado y lematizado).

> Una computadora u ordenador es un sistema digital con tecnología microelectrónica capaz de procesar información a partir de un grupo de instrucciones denominado programa.

Una	ART	un
computadora	NC	computadora
u	CC	o
ordenador	NC	ordenador
es	VSfin	ser
un	ART	un
sistema	NC	sistema
digital	ADJ	digital
con	PREP	con
tecnología	NC	tecnología
microelectrónica	ADJ	microelectrónica
capaz	ADJ	capaz
de	CSUBI	de
procesar	VLinf	procesar
información	NC	información
a	PREP	a
partir	VLinf	partir
de	PREP	de
un	ART	un
grupo	NC	grupo
de	PREP	de
instrucciones	NC	instrucción
denominado	VLadj	denominar
programa	NC	programa
.	FS	.

La evaluación de este etiquetador se muestra en la sección 3.2.3 en la que además se le compara con el etiquetador descrito a continuación: Freeling.

3.2.2 Freeling, un etiquetador basado en diccionario

En realidad Freeling [Atserias, et. al. 1998] no es solamente un etiquetador morfosintáctico, es toda una plataforma de herramientas para el procesamiento de lenguaje natural que cuenta con tokenizador, etiquetador y detector de entidades nombradas, entre otras tareas del PLN. Esta herramienta puede ser utilizada en el análisis de documentos en varios idiomas como el catalán,

[66] Tomado del Corpus de informática en español.

inglés y español.

Freeling es desarrollado en la Universidad Politécnica de Catalunya y una de las mayores ventajas que ofrece es que se trata de una biblioteca que puede ser accedida por programas ajenos a la aplicación por medio de clases de interfaz de la biblioteca. Además, al tratarse de software libre, es posible modificarlo según las necesidades del proyecto.

Como se observa en el título de este apartado, el etiquetador morfosintáctico de Freeling tiene un diccionario de palabras en español que cuenta además con un vector con las etiquetas que puede tomar cada palabra y sus probabilidades. Dicho diccionario ha sido generado por medio de la obtención, en primera instancia, de una lista de raíces de palabras (*apag* es la raíz de apagar, apago y apagaré, por ejemplo) obtenidas por medio del análisis de varios corpus. Con dichas raíces, se generaron todas las inflexiones posibles de las palabras de manera automática. Así, de una lista de 90,000 lemas fueron generadas 800,000 formas. Las palabras del texto a analizar son buscadas en este diccionario para determinar la etiqueta que les corresponde.

Sin embargo, como se observó también en el caso de TreeTagger, existen palabras cuya categoría gramatical resulta ambigua. De hecho, en el corpus utilizado para la realización de pruebas de este etiquetador, el 39.26% de las palabras presentaba ambigüedad en cuanto a su categoría con un radio de ambigüedad de 2.63 etiquetas/palabra[67].

En el caso de que la categoría gramatical de una palabra sea ambigua, ya sea porque no se encontró en ninguna de las listas o porque tiene varias categorías relacionadas, la categoría gramatical de la palabra se determina con base en árboles de decisión de manera muy similar a la descrita en el caso de TreeTagger.

A continuación el mismo texto que se usó para mostrar la salida de TreeTagger ahora etiquetado con Freeling

> Una computadora u ordenador es un sistema digital con tecnología microelectrónica capaz de procesar información a partir de un grupo de instrucciones denominado programa.

```
Una uno DI0FS0
computadora computadora NCFMS000
u u CC
ordenador ordenador NCMS000
es ser VSIP3S0
un uno DI0MS0
sistema sistema NCMS000
digital digital AQ0CS0
con con SPS00
tecnología tecnología NCFS000
microelectrónica microelectrónica AQ0FS0
capaz capaz AQ0CS0
de de SPS00
procesar procesar VMN0000
información información NCFS000
a_partir_de a_partir_de SPS00
un uno DI0MS0
grupo grupo NCMS000
de de SPS00
instrucciones instrucción NCFP000
denominado denominar VMP00SM
programa programa NCMS000
. . Fp
```

[67] Atserias, et. al., op. cit., p. 2.

3.2.3 Evaluaciones y selección del etiquetador

Como se ha observado en los dos apartados anteriores, existen dos etiquetadores para documentos en español al alcance que pueden ser explotados en este trabajo. Para definir cuál es el que se utilizará, se ha realizado una comparación del desempeño que ambos muestran sobre un mismo texto.

Debido a que para evaluar la calidad en el etiquetado de un documento se requiere contar con texto etiquetado y validado (sea a mano o de manera automática), se ha recurrido a texto previamente etiquetado perteneciente al Corpus Técnico del IULA [Bach, et. al. 1997], un corpus etiquetado con el formato EAGLES[68] revisado manualmente que cuenta con textos en catalán, español, inglés, francés y alemán, y cuyos documentos incluyen temas tan diversos como derecho, economía e informática.

Para obtener el texto etiquetado, se realizaron búsquedas de verbos comunes tales como *ser* o *tener* por medio de la herramienta de explotación del corpus IULA denominada Bwananet[69]. Como resultado de estas búsquedas se obtuvo un documento etiquetado con 2,208 palabras y signos de puntuación[70]. Con este texto etiquetado como referencia, el procedimiento de evaluación seguido fue el siguiente:

1. Las etiquetas en el documento de referencia fueron eliminadas para dejar un documento en texto plano que pudiera ser etiquetado con cada una de las herramientas a evaluar.
2. El documento limpio fue etiquetado tanto con TreeTagger como con Freeling.
3. De manera manual, la salida de los tres procesos de etiquetado fue revisada, localizando aquellas etiquetas que fueron asignadas de manera incorrecta.
4. Se obtuvo el porcentaje de error de los etiquetadores evaluados.

A continuación, se puede observar un extracto del texto limpio seguido por las tres versiones de etiquetado. Cabe señalar que tanto Freeling como TreeTagger proporcionan, además de la categoría sintáctica, el lema de cada palabra[71]. La salida de estos dos etiquetadores presenta una palabra por renglón, sin embargo se presenta aquí sin saltos de línea por razones de espacio y claridad en la comparación.

---

68 http://www.ilc.cnr.it/EAGLES/home.html

69 http://bwananet.iula.upf.edu/indexes.htm

70 Estas búsquedas fueron realizadas por Rodrigo Alarcón Martínez en la Universidad Pompeu Fabra de Barcelona.

71 Que como se verá más adelante, ha permitido obtener una mejor salida del extractor de términos.

Texto original

El ADN formado a partir del genoma de ARN por la transcriptasa inversa, se integra al cromosoma de una célula hospedera como un provirus. Un retrovirus de suma importancia es el VIH, virus de la inmunodeficiencia humana, que provoca el SIDA, síndrome de inmunodeficiencia adquirida. Además, el ADN codifica para el ADN durante la replicación. Por lo tanto, también puede ocurrir el flujo de información ARN – ARN.

Etiquetado IULA

El/AMS ADN/N4666 formado/VC--SM a/P partir/VI---- de/P l/AMS genoma/N5-MS de/P ARN/N4666 por/P la/AFS transcriptasa/N5-FS inversa/HFS ,/Z se/R6EZZZZ integra/V8R6S- a/P l/AMS cromosoma/N5-MS de/P una/E6--FS célula/N5-FS hospedera/N5-FS como/D4 un/J6--MS provirus/N5-M6 ./Z Un/J6--MS retrovirus/N5-M6 de/P suma/JQ--FS importancia/N5-FS es/VDR3S- el/AMS VIH/N4666 ,/Z virus/N5-M6 de/P la/AFS inmunodeficiencia/N5-FS humana/JQ--FS ,/Z que/RR---66 provoca/V8R6S- el/AMS SIDA/N4666 ,/Z síndrome/N5-MS de/P inmunodeficiencia/N5-FS adquirida/VC--SF ./Z Además/D4 ,/Z el/AMS ADN/N4666 codifica/V8R6S- para/P el/AMS ADN/N4666 durante/D4 la/AFS replicación/N5-FS ./Z Por lo tanto/D4 ,/Z también/D4 puede/V8R6S- ocurrir/VI---- el/AMS flujo/N5-MS de/P información/N5-FS ARN/N4666 -/Z ARN/N4666 ./Z

Etiquetado Freeling

El#el/DA0MS0 ADN#adn/NP00000 formado#formar/VMP00SM a_partir_de#a_partir_de/SPS00 el#el/DA0MS0 genoma#genoma/AQ0FS0 de#de/SPS00 ARN#arn/NP00000 por#por/SPS00 la#el/DA0FS0 transcriptasa#transcriptasa/NCFS000 inversa#inverso/AQ0FS0 ,#,/Fc se#él /0300000 integra#integrar/VMIP3S0 a#a/SPS00 el#el/DA0MS0 cromosoma#cromosoma/NCMS000 de#de/SPS00 una#uno/DI0FS0 célula#célula/NCFS000 hospedera#hospedera/AQ0FS0 como#como/CS un#uno/DI0MS0 provirus#provirus/NCMN000 .#./Fp Un#uno/DI0MS0 retrovirus#retrovirus/NCMN000 de#de/SPS00 suma#suma/NCFS000 importancia#importancia/NCFS000 es#ser/VSIP3S0 el#el/DA0MS0 VIH#vih/NP00000 ,#,/Fc virus_de_la_inmunodeficiencia_humana#virus_de_la_inmunodeficiencia_humana/NCMN000 ,#,/Fc que#que/PR0CN000 provoca#provocar/VMIP3S0 el#el/DA0MS0 SIDA#sida/NP00000 ,#,/Fc síndrome#síndrome/NCMS000 de#de/SPS00 inmunodeficiencia#inmunodeficiencia/NCFS000 adquirida#adquirir/VMP00SF .#./Fp Además#además/RG ,#,/Fc el#el/DA0MS0 ADN#adn/NP00000 codifica#codifica/VMIP3S0 para#para/SPS00 el#el/DA0MS0 ADN#adn/NP00000 durante#durante/SPS00 la#el/DA0FS0 replicación#replicación/NCFS000 .#./Fp Por_lo_tanto#por_lo_tanto/RG ,#,/Fc también#también/RG puede#poder/VMIP3S0 ocurrir#ocurrir/VMN0000 el#el/DA0MS0 flujo_de_información#flujo_de_información/NCFS000 ARN#arn/NP00000 -#-/Fg ARN#arn/NP00000 .#./Fp

Etiquetado TreeTagger

El_el/ART ADN_ADN/ACRNM formado_formar/VLadj a_a/PREP partir_partir/VLinf del_del/PDEL genoma_genoma/NC de_de/PREP ARN_ARN/ACRNM por_por/PREP la_el/ART transcriptasa_transcriptasa/NC inversa_inverso/ADJ ,_,/CM se_se/SE integra_integrar/VLfin al_al/PAL cromosoma_cromosoma/NC de_de/PREP una_un/ART célula_célula/NC hospedera_hospedera/VLfin como_como/CSUBX un_un/ART provirus_provirus/NC ._./FS Un_un/ART retrovirus_retrovirus/NC de_de/PREP suma_suma/NC importancia_importancia/NC es_ser/VSfin el_el/ART VIH_VIH/NP ,_,/CM virus_virus/NC de_de/PREP la_el/ART inmunodeficiencia_inmunodeficiencia/NC humana_humano/ADJ ,_,/CM que_que/CQUE provoca_provocar/VLfin el_el/ART SIDA _SIDA/NP ,_,/CM síndrome_síndrome/NC de_de/PREP inmunodeficiencia_inmunodeficiencia/NC adquirida_adquirir/VLadj ._./FS Además_además/ADV ,_,/CM el_el/ART ADN_ADN/ACRNM codifica_codificar/VLfin para_para/PREP el_el/ART ADN_ADN/ACRNM durante_durante/PREP la_el/ART replicación_replicación/NC ._./FS Por_por/PREP lo_el/ART tanto_tanto/ADV ,_,/CM también_también/ADV puede_poder/VMfin ocurrir_ocurrir/VLinf el_el/ART flujo_flujo/NC de_de/PREP información_información/NC ARN_ARN/ACRNM -_-/DASH ARN_ARN/ACRNM ._./FS

Como puede observarse, los formatos de las etiquetas son distintos en los tres etiquetados. Mientras que el etiquetado original del IULA y Freeling siguen variedades del formato EAGLES, el de TreeTagger utiliza una versión traducida al español de las utilizadas en el proyecto Penn Treebank[72].

En dicha evaluación, además de considerar el conocimiento propio sobre las palabras y su categoría gramatical, la etiqueta determinada por las dos herramientas evaluadas fueron comparadas con las etiquetas originales del documento[73].

El porcentaje de error de las tres aplicaciones, incluyendo qué proporción de los errores ocurrió en la asignación de las categorías gramaticales que con mayor frecuencia muestran los términos (nombres comunes, adjetivos y verbos) se muestra en la tabla 3.3.

etiquetado	*total*	*nombres*	*adjetivos*	*verbos*
IULA	0.67%	6.66%	66.66%	13%
TreeTagger	0.72%	35%	25%	35%
Freeling	1.13%	48.38%	41.95%	6.45%

Tabla 3.3: Porcentaje de error de los etiquetadores sobre el corpus del IULA

Por ende, en el caso del etiquetado del IULA, el 0.67% de las etiquetas fueron asignadas de manera incorrecta, de las cuales el 13% eran verbos, el 66.66% adjetivos y el 6.66% nombres comunes.

No conforme con esta comparación, se etiquetó un archivo de 2,381 palabras perteneciente al corpus de informática con ambas herramientas. Un enunciado de dicho corpus se presenta como ejemplo en la tabla 3.4 mostrando en cada columna (una palabra por línea) el texto limpio y el etiquetado de Freeling y TreeTagger.

De nueva cuenta, la salida de ambos etiquetadores fue revisada a mano incluyendo en esta ocasión la lematización. El porcentaje de error se resume en la tabla 3.5.

Observando estos datos, queda claro que el desempeño de TreeTagger es mejor que el de Freeling (incluso se observó que la velocidad de TreeTagger es mayor). Es por estas razones que se ha optado por utilizar TreeTagger.

Por último, resulta pertinente señalar que ambas herramientas pueden ser entrenadas para mejorar los resultados que generan. En ambos casos el entrenamiento consiste en proporcionar un corpus previamente etiquetado y validado para generar los distintos diccionarios de etiquetas y lematizaciones realizadas y los árboles de decisión. Aunque este entrenamiento no fue realizado en este trabajo, puede implicar la obtención de mejores resultados en la extracción.

[72] http://www.cis.upenn.edu/~treebank/

[73] Cabe señalar que a pesar de que el Corpus Técnico del IULA está revisado a mano, sigue presentando errores de etiquetado.

texto	*etiquetado Freeling*		*etiquetado TreeTagger*	
La	el	DA0FS0	el	ART
característica	característica	NCFS000	característica	NC
principal	principal	AQ0CS0	principal	ADJ
que	que	PR0CN000	que	CQUE
la	él	PP3FSA00	él	PPO
distingue	distinguir	VMIP3S0	distinguir	VLfin
de	de	SPS00	de	PREP
otros	otro	DI0MP0	otro	QU
dispositivos	dispositivos	NCMP000	dispositivo	NC
similares	similar	AQ0CP0	similar	ADJ
,	,	Fc	,	CM
como	como	CS	como	CSUBX
una	uno	DI0FS0	un	ART
calculadora	calculadora	NCFS000	calculadora	NC
no	no	RN	no	NEG
programable	programable	AQ0CS0	programable	ADJ
,	,	Fc	,	CM
es	ser	VSIP3S0	ser	VSfin
que	que	CS	que	CQUE
puede	poder	VMIP3S0	poder	VMfin
realizar	realizar	VMN0000	realizar	VLinf
tareas	tarea	NCFP000	tarea	NC
muy	mucho	RG	muy	ADV
diversas	diverso	AQ0FP0	diversos	QU
cargando	cargar	VMG0000	cargar	VLadj
distintos	distinto	DI0MP0	distintos	QU
programas	programa	NCMP000	programa	NC
en	en	SPS00	en	PREP
la	el	DA0FS0	el	ART
memoria	memoria	NCFS000	memoria	NC
para	para_que	CS	para	PREP
que			que	CQUE
los	él	PP3MPA00	él	PPO
ejecute	ejecutar	VMSP3S0	ejecutar	VLfin
el	el	DA0MS0	el	ART
procesador	procesador	NCMS000	procesador	NC
.	.	Fp	.	FS

Tabla 3.4: Comparación del etiquetado de un enunciado con Freeling y TreeTagger

	TreeTagger	*Freeling*
etiquetado	0.48%	0.67%
lematización	0.82%	1.39%

Tabla 3.5: Porcentaje de error en etiquetado y lematización para Freeling y TreeTagger

3.3 La construcción de términos en el español

Para realizar la detección de términos en un corpus técnico, es necesario saber cuáles son los principales patrones que presentan en la lengua que se desea tratar pues son los patrones que se buscarán en el texto y que devolverán a los candidatos a término.

Cardero [2003] ofrece una lista con los patrones que con mayor frecuencia presentan los términos en el español, los más comunes se muestran en la tabla 3.6[74].

patrón	*ejemplo*
NC	amplificador
NC+PREP+NC	tarjeta de red
NC+ADJ	disco duro

Tabla 3.6: Patrones sintácticos más comunes de los términos según Cardero

Si bien se ha visto que una buena proporción de los términos presenta alguno de estos tres patrones, existen muchos otros que presentan patrones distintos. Por ello, se ha optado por hacer una búsqueda manual sobre el corpus para ver cuáles son los patrones que presentan todos los términos hallados y decidir si vale la pena considerarlos en el filtro lingüístico o no.

Para realizar la detección de términos, que permitan determinar sus patrones sintácticos, se ha utilizado una técnica planteada por L'Homme y Bae [2006] que consiste en los siguientes cuatro pasos:

a) Se detectan las unidades léxicas relacionadas con el campo de conocimiento abordado (en este caso ingeniería y computación[75]). Dichas unidades léxicas incluyen elementos como *procesador*, o *computadora personal*; unidades de medida como *Hz* y *byte*; y actores como *programador*.
b) Si existen unidades léxicas que sean predicativas (verbos, nominalizaciones, adjetivos, etc.), éstas son seleccionadas si sus actores son entidades que fueron aceptadas en el punto anterior. Un ejemplo de ello está en la frase "el usuario **carga** un programa en memoria". Sin embargo, si la unidad léxica tiene el mismo significado en áreas especializadas y no especializadas, no debe ser considerada.
c) Si hay una unidad léxica derivativa, es seleccionada si tiene una relación semántica con un término seleccionado en alguno de los dos pasos anteriores. Ejemplo de ello es comprimir y *descomprimir*, *acceso* y *accesibilidad*.
d) Cualquier unidad léxica con una relación paradigmática con algún término seleccionado en los pasos anteriores. Por ejemplo, *impresora* y *escáner*.

Esta metodología fue aplicada tanto en el CLI como en el corpus de informática. En el caso del CLI, se eligieron de manera aleatoria dos de los cinco documentos del área de ingeniería mecánica, sumando un total de 4,465 palabras. Algunos de los términos que fueron obtenidos son *motor diesel* y *motocultor de alto despeje*. Por medio de esta detección manual de términos, los

[74] Cardero, op. cit., pp. 95-99.

[75] En francés se conoce a la computación como informatique, razón por la cual el corpus fue incorrectamente bautizado como corpus de informática, debido a un calco hacia el español, cuando en realidad se trata de un corpus de computación.

patrones encontrados con mayor frecuencia para esta área de la ingeniería son:

a) NC+ADJ
b) NC+PREP+NC
c) NC+NC

El mismo proceso, pero con la ventaja de la experiencia obtenida en esta etapa, fue realizado sobre el corpus de informática. De manera aleatoria, se tomó 10% de los archivos que componen a este corpus, es decir 20 archivos. En total, éstos archivos tienen 16,629 palabras.

Entre las más de 16,000 palabras, fueron detectados 615 términos. Los patrones que presentaron mayor frecuencia se encuentran en la tabla 3.7.

patrón	*ocurrencias*	*ejemplos*
NC/NP	327	interfase, adaptador, usuario, sesión, CPU, puerto, computadora, programa
NC+ADJ	90	red local, computadora personal, sonido envolvente, disco duro
NC+PDE+NC[76]	49	dispositivo de almacenamiento, código de barras
NC+NP / NC+NC	60	tarjeta madre, puerto USB, tecnología RISD, memoria RAM
V	21	navegar, correr, ejecutar, emular

Tabla 3.7: Patrones sintácticos con mayor frecuencia en el corpus de muestra de computación

La proporción en la que aparecen los distintos patrones de los términos se presenta en la figura 3.2[77]. Algunos de estos patrones, debido a que presentan poca variación con respecto a otros con mayor frecuencia, son incluidos en las reglas que detectan a estos patrones. Otros no han sido considerados debido a que sólo han ocurrido en una ocasión en el corpus[78].

Al comparar los patrones hallados en estos corpus con los definidos por Cardero, se puede observar que el comportamiento de los términos encontrados es muy similar al señalado por ella.

[76] Debido a que se observó que la única preposición incluida en los términos obtenidos es *de*, en lugar de incluir cualquier preposición, sólo ésta se contempla.

[77] Cabe señalar que la forma gaussiana de la gráfica fue definida "a mano" en aras de la claridad de la misma. En realidad no tiene ningún significado.

[78] Estos patrones son: "NP PDE NP NP", "NP NP NP", "NC NC PDE NC", "NC PDE NC NP", "PE NC", "NMEA", "NC PDE NC PDE ADJ NC", "NC PDE NC PDE NC", "ACRNM PDE NC", "NC PE ACRNM", "NP ADJ".

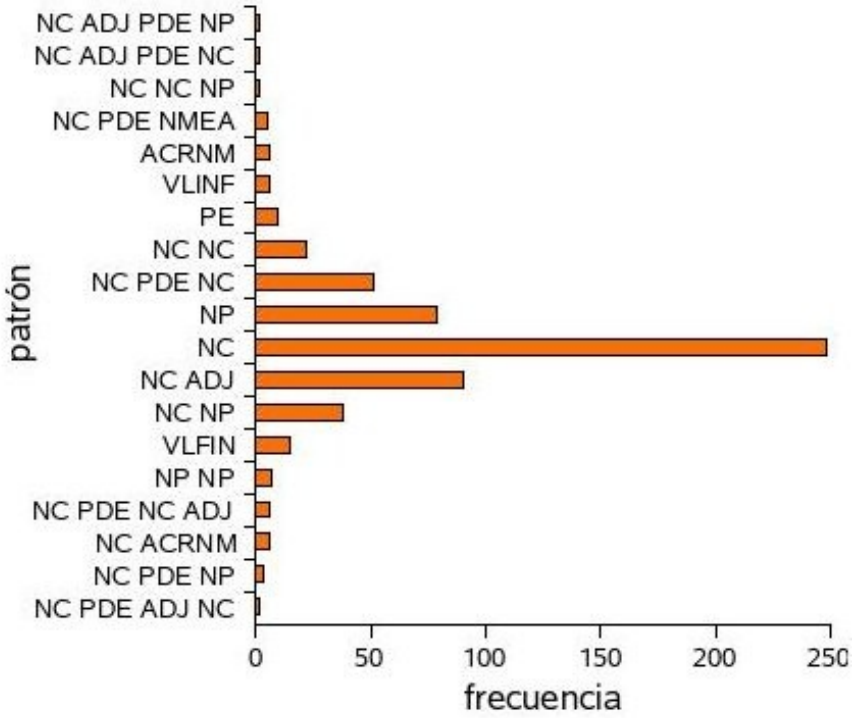


Fig. 3.2: Patrones de los términos en el corpus de informática

En el apartado 3.5.1.1 se describe cómo se han utilizado los patrones definidos en esta etapa para la creación de las reglas del filtro lingüístico y la lista de paro de la etapa lingüística de C-value.

3.4 El algoritmo para la extracción automática de términos C-value/NC-value

El algoritmo C-value/NC-value[79], que como se señaló en el segundo capítulo es un método híbrido para la extracción de términos, fue originalmente diseñado para obtener candidatos a término del área de biomedicina en inglés. Antes de abordar las adaptaciones que le han sido realizadas a este algoritmo, se ofrece una descripción del original.

La etapa inicial, denominada C-value, es en realidad la única híbrida del algoritmo. La primera parte de esta etapa consiste en la generación de una lista de candidatos a término por medio de un filtro lingüístico basado en la búsqueda de patrones sintácticos (como los descritos en el apartado anterior para el español) y la aplicación de una lista de paro para eliminar candidatos que contengan palabras que no se espera que formen parte de los términos del área tratada. La segunda parte consiste en el cálculo del potencial de cada candidato de ser un verdadero término por medio de su longitud y frecuencia de aparición en el corpus (este potencial es llamado precisamente C-value).

Por su parte, la etapa final, llamada NC-value, considera que el contexto en el que aparecen los términos es significativo para su discriminación, por lo que considerar la vecindad

[79] Frantzi, et. al., op. cit.

de cada uno de ellos puede favorecer la calidad de los resultados.

A continuación, se describen con detalle las dos etapas del algoritmo C-value/NC-value.

3.4.1 C-value, la etapa híbrida de la extracción

C-value es un "método independiente del dominio para el reconocimiento automático de términos multipalabra"[80] que se divide en una etapa lingüística y una estadística. Su objetivo es obtener, por medio de un filtro sintáctico y la aplicación de una lista de paro, un conjunto de candidatos que son ordenados con base en una medida que toma en cuenta la frecuencia de aparición y la longitud de cada candidato.

La primera etapa, la lingüística, se encarga de la detección de candidatos a término por medio de la detección de patrones sintácticos en el texto. Esta etapa se divide en tres pasos:

1. Etiquetado de partes de la oración del corpus.
2. Aplicación del filtro lingüístico al texto etiquetado para la obtención de candidatos.
3. Selección de buenos candidatos por medio de la lista de paro.

El etiquetado de partes de la oración ha sido descrito en el apartado 3.2, por lo que no vale la pena hacerlo de nuevo. Solo resta señalar que, para la versión original en inglés, se utiliza el etiquetador de partes de la oración de Eric Brill [1992].

En cuanto al filtro lingüístico, existe la ventaja de que han sido desarrollados varios extractores para procesar documentos en inglés, por lo que son bien conocidos los conjuntos de reglas necesarios para detectar buenos candidatos a término. Entre las tres reglas lingüísticas que se han analizado para la versión original del algoritmo, la más interesante es:

$$((Adj|Noun)^{+} \mid ((Adj|Noun)^{*}\ (NounPrep)^{2}\ )(Adj|Noun)^{*}\ )\ Noun$$

Esta regla [Justeson y Katz 1995] es capaz de detectar candidatos que presentan varios patrones como *Adj+Noun*, *Noun+Noun* y *Noun+Prep+Noun*, extrayendo candidatos como *optic nerve* que es un término, pero detectando también otros como *planes of section*, que no lo son. En la detección de términos en inglés, se ha mostrado que una de las mejores reglas (que mantiene un buen valor tanto de precisión como de recuerdo) es la regla:

$$(Adj|Noun)^{+}\ Noun$$

que sólo detecta candidatos como *fibrous tissue* o *lens capsule*[81]. Este patrón es el más característico de los términos en inglés, cuya construcción suele ser mucho más simple que la de los términos en español.

Con este filtro lingüístico son buscados candidatos a término, que son aquellos sintagmas en el texto que cumplen con los patrones definidos por las reglas. Aquellos candidatos que no tengan una frecuencia superior a un umbral previamente determinado, son descartados. Este proceso genera la lista de candidatos C_1.

Luego, las palabras en la lista de paro son buscadas en la lista C_1; en caso de que una

[80] Frantzi, et. al., op. cit., p. 1.

[81] De hecho, esta regla es la única que TermoStat utiliza en su versión para el inglés.

palabra de la lista de paro se halle en un candidato, éste es eliminado. La lista de paro consiste en un conjunto de palabras cuya categoría gramatical se incluye en el filtro lingüístico (*Adj* y *Noun* si la única regla del filtro fuera la última mostrada), que aparecen frecuentemente en el corpus, pero que no se espera que formen parte de los términos del área tratada.

En el caso del inglés, la lista de paro incluye adjetivos como *great* y *several* y nombres como *year*. Si un candidato tiene alguna de estas palabras, simplemente es eliminado. Los candidatos que persisten en la lista luego de este proceso forman la lista de candidatos C_2. Es aquí donde termina la etapa lingüística del algoritmo. El siguiente paso es ordenar la lista de candidatos C_2 por medio del método estadístico que se describe a continuación.

La etapa estadística de C-value consiste en el cálculo del potencial de un candidato de ser un término basado en su frecuencia de aparición y longitud. Este potencial es el valor por el que la lista de candidatos es ordenada.

Dicho potencial, en otros enfoques, ha sido determinado simplemente por medio de la frecuencia de aparición de los candidatos. Sin embargo, se ha observado que esta medida no es suficiente para realizar un ordenamiento certero de candidatos. Los cuatro parámetros utilizados para determinar el C-value de los candidatos son[82]:

- La frecuencia total de aparición del candidato en el corpus.
- La frecuencia del candidato como parte de otros candidatos de mayor longitud.
- El número de estos candidatos de mayor longitud.
- La longitud del candidato.

La ocurrencia frecuente de un candidato a término en un corpus implica una buena probabilidad de que sea realmente un término, sin embargo, una baja frecuencia de un candidato no implica forzosamente que no se trate de un verdadero término. Por ejemplo, el sintagma *disco duro* apareció 18 veces en un documento de 1,608 palabras en el que precisamente se da una explicación de qué es un disco duro mientras que sólo apareció 2 veces en un documento de 603 palabras que simplemente habla de los accesorios de una computadora. Aún presentando esta baja frecuencia, *disco duro* no deja de ser un término del segundo documento, la diferencia es el nivel de especialización de éste. Es por ello que contar con un corpus bien nivelado es de gran importancia.

Por otro lado, el que un candidato forme parte de otro de mayor longitud, puede implicar que se trata en realidad de una simplificación lingüística del término más largo (como en el caso de *red* y *red de computadoras*). Es por ello que la frecuencia del candidato más largo hace que el potencial del candidato analizado decrezca (lo que se verá a continuación en la fórmula de cálculo de C-value).

Sin embargo, que el candidato analizado ocurra en varios candidatos diferentes de mayor longitud, implica cierta independencia con respecto a ellos. Incluso, podría darse el caso de que los candidatos más largos sean casos especiales del candidato analizado, como en *red de computadoras*, *red de computadoras inalámbrica* y *red de computadoras de área local*. En este caso, *red de computadoras* es por sí mismo un término y los otros dos casos son algunas de sus variantes. Por ello, entre mayor sea la cantidad de candidatos de longitud mayor en los que el candidato analizado aparezca, la relevancia del factor negativo de formar parte de ellos disminuye.

Finalmente, queda hablar de la longitud de los candidatos. Entre más largo sea un

[82] Frantzi, et. al., op. cit., p. 3.

sintagma que aparece con cierta frecuencia, habrá una relación más fuerte entre sus palabras constituyentes, por lo que la probabilidad de que en conjunto tengan un concepto asociado es alta. Es por esta razón que la longitud de los candidatos es tomada en cuenta como un factor favorable en el cálculo del C-value.

Basado en estos cuatro parámetros, el algoritmo C-value determina la posibilidad de que un sintagma candidato sea en verdad un término mediante la fórmula 1.

$$C\text{-}value = \begin{cases} \log_2|a| * f(a) & si\ a\ no\ aparece\ en\ otros\ candidatos \\ \log_2|a|\left(f(a) - \frac{1}{P(T_a)}\sum_{b\in T_a} f(b)\right) & de\ lo\ contrario \end{cases} \quad (1)$$

donde: a es el sintagma candidato
$|a|$ es la longitud del sintagma candidato a
$f(a)$ es la frecuencia de ocurrencia del sintagma a en el corpus
T_a es el conjunto de candidatos de mayor longitud que contienen a a
$P(T_a)$ *es el* número de esos candidatos (tipos) e incluye al mismo candidato analizado
$\Sigma f(b)$ *es la* ocurrencia total de a como subcadena del sintagma candidato b tal que $|a|<|b|$

Como ejemplo digamos que se desea calcular el C-value de algunos de los sintagmas mencionados anteriormente: *red de computadoras* y *red de computadoras inalámbrica*. Supóngase que estos sintagmas aparecen 3 y 2 veces en el corpus respectivamente. Debido a que el segundo sintagma no aparece en otros de mayor longitud, su C-value se calcula con la primera parte de la fórmula 1:

$$C\text{-}value(redDeComputadorasInalambrica) = \log_2(4) * 2$$
$$C\text{-}value(redDeComputadorasInalambrica) = 4$$

el cálculo del C-value del sintagma candidato *red de computadoras* se calcula con la segunda parte de la fórmula 1:

$$C\text{-}value(redDeComputadoras) = \log_2(3)\left(3 - \tfrac{1}{2}1\right)$$
$$C\text{-}value(redDeComputadoras) = 3.96$$

La primera parte de la fórmula se usa sólo en los candidatos que presentan la de mayor longitud,

$$C\text{-}value(redDeComputadoras) = 4.75$$

llamémosla N, en el corpus, porque evidentemente no se han hallado candidatos de mayor longitud que los pudieran contener. Sólo para los candidatos de longitud $l<N$ que se encuentren dentro de otros candidatos de mayor longitud se aplica la segunda parte de la fórmula.

Como se ha observado, se toman en cuenta tres frecuencias de cada candidato:

- $f(b)$ es la frecuencia total del candidato b en el corpus.
- $t(b)$ es la frecuencia de aparición de b dentro de otros candidatos.

- *c(b)* es el número de candidatos de mayor longitud que *b* en los que éste aparece.

Este proceso sobre la lista de candidatos C_2 genera la lista ordenada de candidatos con base en la posibilidad de que un sintagma candidato sea en verdad un término y se le denomina C_3. El algoritmo simplificado se presenta a continuación:

etiquetado del corpus
extracción de cadenas por medio del filtro lingüístico
eliminación de cadenas con frecuencia menor a un umbral definido
filtrado de candidatos por medio de la lista de paro
para todas las cadenas *a* con la mayor longitud:

$$C\text{-}value(a) = log_2|a| * f(a)$$

si el *C-value(a)* ≥ umbral
agregar *a* a la lista de salida
para todas las subcadenas *b* en *a*:
actualizar *t(b)*
actualizar *c(b)*
para todas las cadenas de longitud menor a la máxima en orden descendiente:
si *a* aparece por primera vez:

$$C\text{-}value(a) = log_2|a| * f(a)$$

de lo contrario:

$$C\text{-}value(a) = log_2|a|\left(f(a) - \frac{1}{c(a)}t(a)\right)$$

si el *C-value(a)* ≥ umbral:
agregar *a* a la lista de salida[83]
para todas las subcadenas *b* en *a*:
actualizar *t(b)*
actualizar *c(b)*

Si bien la lista ordenada de candidatos C_3 que resulta del cálculo de C-value de cada uno de los candidatos proporciona ya un mejor resultado que un simple conteo de frecuencias, el resultado puede mejorarse aún más.

3.4.2 NC-value, considerando el contexto del candidato

NC-value es "una extensión del algoritmo C-value que considera información contextual para la extracción de términos"[84], pues no sólo la frecuencia y longitud de un sintagma son importantes para determinar si éste es un término.

El contexto en el que aparece una palabra puede ser utilizado para identificar sinónimos (los sinónimos suelen presentarse con el mismo conjunto de palabras). Además, las palabras que se encuentran en la vecindad de otra pueden ser útiles para identificar su significado. Es por eso que se espera que las palabras que ocurren junto a los candidatos a término sean relevantes para ellos.

[83] Debido a que la lista de candidatos debe estar siempre ordenada, cada nuevo candidato es colocado en la posición que le corresponde con base en su C-value, razón por la cual no es necesario hacer una ordenación final de la lista.

[84] Frantzi, et. al., op. cit., p. 9.

Para Sager [1978], los sintagmas terminológicos se diferencian de los sintagmas comunes en que no interactúan libremente con cualquier palabra. Por ejemplo, *disco duro* frecuentemente tiene en su vecindad palabras como *capacidad*, *GB*, *guardar* o *formatear*, pero no tiene otras como *compilar* o *editor*. Por ende, las palabras significativas que aparecen en la vecindad de un término suelen repetirse en distintos contextos además de afectar al mismo tipo de términos; como en el caso de la palabra de contexto *compilar* frente a términos como *programa* o *código fuente*.

Una palabra de contexto es una palabra significativa que aparece en la vecindad de un candidato a término[85]. Estas palabras de contexto pueden ser utilizadas para que los verdaderos términos entre los candidatos aparezcan más arriba dentro de la lista ordenada. Las palabras consideradas de contexto son los verbos, nombres y adjetivos que ocurren en la vecindad de los candidatos a término.

Queda ahora señalar cómo es definido y utilizado el conjunto de palabras de contexto. Una lista ordenada de palabras de contexto es obtenida de las palabras significativas que aparecen en la vecindad de los candidatos a término en el tope de la lista ordenada con base en el C-value. El hecho de que una palabra significativa aparezca en la vecindad de estos candidatos, cuya probabilidad de ser verdaderos términos es alta, puede ser explotado bajo el supuesto de que pueda aparecer en el corpus con otros candidatos que sean también términos. Formalmente el peso dado a una palabra de contexto se define por medio de la fórmula 2[86].

$$weight(w) = \frac{t(w)}{n} \qquad (2)$$

donde: *w* es la palabra de contexto analizada (nombre, adjetivo o verbo)
weight(w) es el peso asignado a la palabra *w*
t(w) es el número de candidatos con los que aparece la palabra *w*
n es el número total de candidatos considerados (para expresarlo como una probabilidad)

Se ha demostrado que las palabras que aparecen con frecuencia con un candidato a término cuyo C-value es alto, suelen aparecer con otros candidatos con menor C-value que son términos reales[87].

Abordemos ahora de lleno el NC-value. Este algoritmo se divide en tres etapas. La primera etapa es el cálculo del C-value tal como fue descrito en el apartado anterior. La segunda etapa consiste en la extracción de palabras de contexto y la tercer etapa considera los pesos obtenidos para cada una de las palabras de contexto para recalcular la potencialidad de ser término de los candidatos. A continuación se describen estas etapas.

La primer etapa ha sido descrita en el apartado anterior y no sufre ninguna modificación. Ésta genera una lista de candidatos ordenada con base en el C-value calculado por medio de la fórmula 1. Como se señaló en el apartado anterior, esta lista se ha denominado C_3.

La segunda etapa consiste en tomar a los candidatos que aparecen al principio de la lista C_3 y obtener una lista con sus palabras de contexto y peso calculado por medio de la fórmula 2. La razón por la que se considera únicamente a los candidatos con mayor C-value es que este

[85] Si bien en el trabajo original en el que se describe este algoritmo no se detalla la cantidad de palabras a la izquierda y derecha que se consideran para cada uno de los términos, como se verá en el siguiente apartado se ha optado por utilizar una ventana de cinco palabras.

[86] Frantzi, et. al., op. cit., p. 10.

[87] Para profundizar en esta evaluación, consúltese Frantzi, et. al., op. cit., pp. 10-11.

conjunto presenta mayor precisión que el resto de la lista.

Finalmente, la tercer etapa consiste en reordenar la lista C_3 considerando a las palabras de contexto de cada uno de los candidatos. De esta manera, lo que se intenta es llevar a los verdaderos términos a una posición más alta en la lista mientras que los que no lo son, van bajando poco a poco a través de ella.

El procedimiento consiste en tomar las palabras no funcionales halladas en la vecindad de cada uno de los candidatos y buscarla en la lista de palabras de contexto previamente generada. A la palabra se le asigna el peso calculado si se encuentra en la lista o cero de no ser así. Es con base en el peso de cada palabra de contexto y su frecuencia de aparición con el candidato en cuestión que se calcula el denominado "factor de contexto". Éste consiste simplemente en la sumatoria de las multiplicaciones de la frecuencia y peso de cada palabra de contexto.

Por ejemplo, sean w_1, w_2 y w_3 las palabras de contexto del candidato a término *a* (que aparecen con éste 10, 20 y 30 veces respectivamente), el cálculo del factor de contexto del candidato *a* será $10*w_1 + 20*w_2+30*w_3$. La fórmula 3 es la utilizada para el cálculo del NC-value de cada uno de los candidatos.

$$NC\text{-}value(a) = 0.8C\text{-}value(a) + 0.2 \sum_{b \in C_a} f_a(b) weight(b) \qquad (3)$$

donde: *a* es el candidato a término
C_a es el conjunto de palabras de contexto del candidato *a*
b es una palabra de contexto del candidato *a*
$f_a(b)$ es la frecuencia de aparición de *b* como palabra de contexto del candidato *a*
weight(b) es el peso asignado a *b* como palabra de contexto

El hecho de asignar el 80% del potencial final al valor de C-value con respecto al 20% del NC-value fue determinado después de varios experimentos en los que se variaron estos parámetros.

Como se señaló en un principio, ni el cálculo de C-value ni el de NC-value sirven para discriminar candidatos a términos. Su objetivo es simplemente hacer que los verdaderos términos se localicen en la parte superior de una lista ordenada con base en sus valores calculados, lo que no forzosamente implica que un candidato en la parte inferior de la lista no sea un término.

Si se desea, se puede determinar un umbral para que sólo los candidatos cuyo NC-value sea superior sean considerados en la lista de salida, lo que mejora la precisión pero empeora el recuerdo.

3.5 Adaptación del algoritmo C-value/NC-value

Se ha probado que el algoritmo C-value/NC-value tiene un buen desempeño en la tarea de extracción de candidatos a término multipalabra en un corpus del área de biomedicina en inglés [Frantzi et. al. 2000] y de tecnología de la información en chino [Ji, et. al. 2007] ente otros. A continuación, se presentan las adaptaciones realizadas a este algoritmo para su aplicación en un corpus de textos en español (hasta el momento de las áreas de computación e ingeniería) en el que además sea posible obtener candidatos de una sola palabra.

3.5.1 Adaptación de C-value

El filtrado de candidatos se realiza en el algoritmo C-value (la etapa lingüística). Es por ello que una de las principales modificaciones debe realizarse aquí. Dicha modificación consiste en la creación de reglas lingüísticas apropiadas para la detección de los patrones más comunes que presentan los términos en español. Además, la etapa estadística es modificada ligeramente para permitir el manejo de candidatos de longitud 1.

3.5.1.1 Modificación de la etapa lingüística

Como se observó en el capítulo 3.2.3, el etiquetador morfosintáctico que ha sido implementado para la primera etapa de la extracción es TreeTagger.

En cuanto a la detección de candidatos, en el apartado 3.3 se describe la extracción manual de términos de un subconjunto de los corpus de informática e ingeniería y se presentan los patrones presentados por dichos términos. A continuación se muestran las reglas[88], en formato de gramáticas formales, que fueron obtenidas a partir de los patrones sintácticos de este conjunto de términos, las cuales aceptan a todos los términos que fueron detectados manualmente y pueden ser utilizadas como filtro lingüístico del extractor[89].

a) <NC | NP | PE>$^{+}$
b) <NC> <ADJ> (<PDE> <NC | NP>)*
c) <NC> <PDE> <NC | NP | NMEA>
d) <VLFIN | VLINF>
e) <NC>? <ACRNM>
f) <NC> <PDE> ((<NC> <ADJ>) | (<ADJ> <NC>))

En la tabla 3.8 se ofrece una muestra de los candidatos extraídos incluyendo tanto los que en verdad son términos como los que no y un conteo de éstos. Esta extracción fue realizada de manera automática sobre el mismo subcorpus de computación que fue utilizado para la definición de las reglas.

El siguiente proceso en C-value es la eliminación de todos aquellos candidatos cuya frecuencia sea menor a un umbral determinado. Sin embargo, este paso no ha sido considerado en el prototipo[90].

A continuación es necesario implementar la lista de paro, compuesta por palabras que no se espera que formen parte de los términos de un área en particular pero que debido a su categoría gramatical y frecuencia, se espera que aparezcan entre los candidatos.

---

88 Con el formato necesario para su implementación por medio de la biblioteca nltk-lite, disponible para el lenguaje de programación Python.

89 En el siguiente capítulo se observará que, dependiendo de la aplicación para la que se use el extractor, el conjunto de reglas lingüísticas que se utilice puede ser distinto.

90 Esto se debe principalmente a que nuestros corpus de prueba son, en general, relativamente pequeños. Debido a eso, no hay una garantía de que todos los términos reales en dichos corpus aparezcan con una frecuencia superior a un umbral previamente establecido.

regla	*términos detectados*	*no términos detectados*	*correctos*	*incorrectos*
a	computadora, programa, tarjeta madre, sistemas UPS, dispositivos E/S, servidor, Ghz, MB, puerto USB, arquitectura von Neumann, programador	usuario, sistema, dispositivo, Creative Zen Touch, adaptador Extol, Home Theater, formato VHS	164	889
b	sistema operativo, disco duro, cámara digital, sonido envolvente, computadora personal, zoom digital, computadora digital, operación lógica, pantalla táctil	tiempo real, cajero automático, altavoz trasero, modelo anterior, diseño elegante, hogar digital, características técnicas	35	302
c	ángulo de visión, slots de expansión, lenguaje de máquina, dispositivo de almacenamiento, ancho de banda, tasa de transferencia	tecnología de Toshiba, Platinum de MSI, datos de configuración, fabricante de sistemas, calidad de sonido	55	248
d	procesar, programar, correr, capturar	utilizar, diseñar, limitar, capacitar	20	85
e	DDR2, terminal ISDN, IP, slot ISA	P Neo, Pro, LED, EE. UU	14	12
f	computadora de uso general	pantalla de manera lateral, procesador de computadora personal	23	71

Tabla 3.8: Muestra de los candidatos a término detectados con cada una de las reglas

Las palabras que ocurren con mayor frecuencia en un documento suelen ser las funcionales (preposiciones como *de* y *a*, artículos como *los* y *la*). Sin embargo, este tipo de palabras no será considerado para la lista de paro. La razón es muy sencilla: el filtro lingüístico que detecta a los candidatos a término no considera a este tipo de palabras. De hecho, la única palabra funcional que pueden contener los términos es *de* debido a que se observó que forma parte de varios términos en español y se incluye en una de las reglas lingüísticas. Por ello, resulta evidente que aunque presenta una frecuencia muy alta en los documentos, no es considerada en la lista de paro.

Así, las palabras en la lista de paro son únicamente aquellas cuya categoría gramatical es nombre (sea común o propio) y adjetivo. Para conocer algunas de las palabras que se encuentran en la lista de paro obsérvese la tabla 3.9 que contiene las palabras que no ocurren en los términos del área y que aparecieron con mayor frecuencia en el corpus. La lista completa se encuentra en el apéndice 2.

No se espera que los adjetivos *grande* o *nuevo* aparezcan en un término de computación. Sin embargo, otros parecidos como *alto* sí pueden esperarse, como en el caso de *programación de alto nivel*. El caso del nombre común *producto* es muy similar. Luego de realizar una inspección manual en busca de términos y considerando expertos e incluso diccionarios, hasta ahora no se ha encontrado un caso de un término que cuente a esta palabra como una de sus integrantes. Es este tipo de palabras las que forman parte de la lista de paro.

frecuencia	*palabra*
480	grande
345	nuevo
284	producto
270	característica

Tabla 3.9: Algunas palabras de la lista de paro y su frecuencia

Ya en el apartado 1.1.2 se presentó una breve discusión sobre la pertenencia de las marcas registradas y las personas al conjunto de términos de un área de especialidad. Si bien se ha concluido que en general no pertenecen a este conjunto, no se debe olvidar el objetivo que este extractor de términos tendrá en principio. Lo que buscará es detectar los candidatos a término que se encuentren dentro de un contexto definitorio. Dentro de la definición de un término se encuentra a menudo quién lo inventó, en qué compañía y términos de otras áreas de especialidad que no son la abordada. Por ello, es momento de hacer dos consideraciones importantes aplicables para la detección de los términos que aparecen dentro de un contexto definitorio y que pueden ser replanteadas, sustituidas o simplemente eliminadas dependiendo de la aplicación en turno:

- Tanto las marcas registradas (IBM, GNU/Linux, iPod) como los nombres propios (Alan Turing, Richard Stallman, John Hopcroft) serán considerados pseudo-términos válidos.
- Los términos que no sean parte del área de especialidad tratada, pero que formen parte de la definición de otro término en el corpus también serán considerados términos válidos.

Considerando estos dos puntos, la lista de paro contiene 223 palabras que presentaron una alta frecuencia en el corpus de informática, pero que no son o forman parte de un término (incluidos los casos especiales apenas señalados). Ejemplo de estas palabras son: accesorio, característica, calidad, compañía, corporativo, largo, mundo, superior, tamaño, través y vida.

Todas las palabras no funcionales de los candidatos detectados por medio del filtro lingüístico son buscadas en la lista de paro. Lo más sencillo sería eliminar completamente al candidato en el caso de que una de sus palabras se encontrara en la lista de paro (como ocurre en la versión original del algoritmo). Sin embargo, el que uno de los integrantes del candidato se encuentre en esta lista, de ninguna manera implica que el resto de palabras no conforma uno o varios términos.

Ejemplo de ello es el candidato *computadora grande*. Si simplemente es eliminado debido a que grande aparece en la lista de paro, también se eliminará a un verdadero término. Es por casos como éste que se ha optado por no eliminar por completo al candidato, sino únicamente las palabras de éste que aparezcan en la lista de paro y a aquellas que actúen sobre ella, lo que se explica a continuación.

Considérese al candidato *desarrollo de LCDs*. La palabra desarrollo se encuentra en la lista de paro, razón por la que debe ser eliminada quedando ahora *de LCDs*. Pero la función de la preposición *de* era relacionar a ambos nombres. El patrón PDE NC no es característico de los términos, por lo que este candidato sería incorrecto. La solución es no sólo eliminar al nombre, sino también a su preposición vecina, dando como resultado *LCDs* que es un candidato a término que cumple con uno de los patrones característicos.

Un caso un tanto más complicado es el de candidatos como *pantalla de manera lateral*. En este caso, la palabra incluida en la lista de paro es *manera* por lo que debe ser eliminada. Además, el adjetivo *lateral* actúa sobre este nombre eliminado por lo que debe ser descartado también. La preposición *de* se elimina por las razones detalladas en el párrafo anterior quedando como candidato únicamente *pantalla*.

Si la palabra en la lista de paro es un adjetivo, simplemente se elimina del candidato, pero si se trata de un nombre común, las siguientes reglas deben ser aplicadas:

Si la palabra eliminada p_i[91] en el candidato es NC:
    Si la palabra p_{i-1} es una PDE[92]:
        eliminarla
    Si la palabra p_{i+1} es una PDE o un ADJ:
        eliminarla
        Si la palabra p_{i+2} es una PDE o un ADJ:
            eliminarla

A continuación, se proporciona un resumen de las principales adaptaciones realizadas a la etapa lingüística, seguida de una breve discusión sobre las ventajas y desventajas de contar con una lista de paro.

Como puede observarse, además de la forzada modificación de reglas lingüísticas para su aplicación en documentos en español, se ha considerado la detección de candidatos de longitud 1 (NC). Más adelante se verá que si bien esta inclusión induce a la obtención de una mayor cantidad de candidatos a término erróneos, vale la pena soportarlos debido a los buenos candidatos que se consigue detectar.

Además, los candidatos que contienen palabras en la lista de paro no son eliminados por completo. Esta lista de paro permite deshacerse de una buena cantidad de falsos términos. Sin embargo, tiene una fuerte unión con el área a tratar, pues palabras como alto y objeto, que aparecen en términos de computación, no son tan comunes en otras áreas de especialidad. El uso de una lista de paro debe ser definitivamente opcional debido a que las palabras que la conforman dependen del dominio tratado.

3.5.1.2 Adaptación de la etapa estadística

Por medio de las reglas lingüísticas se obtienen buenos candidatos a término, pero también (y desafortunadamente en mayor proporción) otros que definitivamente no lo son. Esto se debe a que sintácticamente los términos no se conforman de una manera especial o única con respecto a otros sintagmas de la lengua. La lista de paro resulta de gran ayuda para evitar candidatos erróneos, pero no es suficiente. Por tanto, la ordenación de la lista de salida con base en el potencial de los candidatos a ser términos es de gran importancia.

En síntesis, la etapa estadística del C-value consiste en el conteo de ocurrencias de los sintagmas que cumplen con ciertos patrones sintácticos característicos de los términos en español. La longitud de los candidatos y su frecuencia de aparición (incluso al interior de otros candidatos) son las variables utilizadas para determinar el potencial de cada candidato de ser en verdad un término.

Sin embargo, debido a que se ha optado por considerar a los candidatos cuya longitud es de una palabra, la fórmula 1, encargada de calcular el potencial C-value de los candidatos, no puede ser utilizada tal como está debido a que el logaritmo de 1, sin importar su base, es siempre 0. Por ello, sin importar la frecuencia de aparición de un candidato de longitud 1, su C-value sería siempre 0.

La única modificación que se realiza a la fórmula para el cálculo de C-value es la adición de 1 al resultado del logaritmo (sin importar la longitud del candidato), lo que resuelve el

[91] El subíndice i denota la posición de la palabra en el candidato.
[92] Preposición de.

problema de la multiplicación por cero[93]. Así, la fórmula adaptada para el cálculo del C-value capaz de manejar candidatos a término de longitud 1 es:

$$C\text{-}value = \begin{cases} (1 + log_2|a|) * f(a) & si\ a\ no\ aparece\ en\ otros\ candidatos \\ (1 + log_2|a|) * \left(f(a) - \frac{1}{P(T_a)} \sum_{b \in T_a} f(b) \right) & de\ lo\ contrario \end{cases} \quad (4)$$

Ésta es la única modificación realizada a la etapa estadística de C-value. Los detalles de conteo de frecuencia y longitud han sido descritas en el apartado 3.4.1 y no presentan ninguna variación, por lo que no se describen de nuevo.

3.5.2 Adaptación de NC-value

Como ya se señaló, NC-value explota el contexto en el que aparecen los candidatos a término para reordenar la lista de salida. En realidad no se ha realizado ninguna modificación importante a esta etapa. Sólo cabe señalar que se ha optado por considerar como palabras de contexto de los candidatos a todas aquellas palabras que, en acorde con [Ji, et. al. 2007], aparezcan en una ventana de 5 palabras[94]. Sólo se han tomado las palabras no funcionales dentro de esta ventana.

Cabe señalar que si un término aparece dentro de la ventana de otro, no hay ninguna razón para no considerar sus palabras no funcionales como palabras de contexto del otro.

Además, se ha considerado que los signos de puntuación que aparecen en una ventana la rompen debido a que su función en el texto es romper una secuencia. Éste es el caso de la coma, punto, punto y coma, paréntesis, etc. Por ello, si bien la longitud por defecto de la ventana es 5, en caso de hallarse algún signo de puntuación éste será la frontera de la ventana.

Para asegurar que quede clara la implementación de la adaptación del algoritmo C-value/NC-value, se presenta a continuación un breve ejemplo.

La tabla 3.10 presenta seis sintagmas candidatos obtenidos por medio del filtro lingüístico, ya que presentan algunos de los patrones característicos de los términos en el español[95]. El cálculo de C-value se realiza paso a paso sobre estos candidatos.

---

[93] Originalmente se había optado por sumar $1x10^{-4}$ para afectar lo menos posible el cálculo del potencial de los distintos candidatos. Sin embargo, experimentalmente se observó que los candidatos de longitud 1 siempre aparecían muy por debajo de la lista de salida aún siendo términos reales. Al sumar la unidad, se obtuvo una mejor distribución.

[94] La ventana de una palabra es el conjunto de palabras que aparecen antes y después de ésta, en este caso, son cinco palabras antes y cinco después.

[95] Estos candidatos han sido tomados del subcorpus del área de hardware del Corpus de Informática en Español, el cual tiene 139,027 palabras.

candidato	*patrón*
disco duro	NC ADJ
estación de trabajo	NC PDE NC
mercado internacional	NC ADJ
sistema operativo	NC ADJ
unidad de disco duro	NC PDE NC ADJ
usuario	NC

Tabla 3.10: Candidatos obtenidos por medio del filtro lingüístico

La palabra *mercado* se encuentra en la lista de paro, por lo que debe retirarse del candidato. Debido a que el adjetivo internacional actúa sobre este nombre común que ha sido retirado y a que un adjetivo no cumple con los patrones de los términos, el candidato se elimina por completo.

El siguiente paso es realizar el conteo de frecuencia y longitud de cada uno de los candidatos. Los valores obtenidos en el corpus se presentan en la tabla 3.11 que incluye para cada candidato su longitud, frecuencia de aparición, frecuencia de aparición al interior de otros candidatos, y cantidad de estos términos de mayor longitud[96], las cuales se denominan *f(b)*, *t(b)* y *c(b)* respectivamente.

long.	*f(b)*	*t(b)*	*c(b)*	*candidato*
4	7	0	1	unidad de disco duro
3	54	16	16	estación de trabajo
2	47	28	22	sistema operativo
2	56	33	19	disco duro
1	289	81	58	usuario

Tabla 3.11: Candidatos con su longitud y frecuencias

El candidato *unidad de disco duro* no aparece en ningun otro candidato de mayor longitud, por lo que se le aplica la primera sección de la fórmula 4 como se observa a continuación.

$$C\text{-}value = (1 + log_2|a|) * f(a)$$
$$C\text{-}value('unidadDeDiscoDuro') = (1 + log_2|4|) * 7$$
$$C\text{-}value('unidadDeDiscoDuro') = 21$$

El caso de los demás candidatos es distinto debido a que sí aparecen en otros candidatos de mayor longitud, por lo que su potencial debe calcularse con la segunda parte de la fórmula 4 como se muestra ahora con el candidato *estación de trabajo*.

[96] Cabe señalar que el valor mínimo que toma esta variable es 1 debido a que se considera que cada candidato se contiene a sí mismo.

$$C\text{-}value = (1 + log_2|a|) * \left(f(a) - \frac{1}{P(T_a)} \sum_{b \in T_a} f(b) \right)$$

$$C\text{-}value('estacionDeTrabajo') = (1 + log_2|3|) * \left(54 - \frac{1}{16} * 16 \right)$$

$$C\text{-}value('estacionDeTrabajo') = 137.003$$

Realizando este mismo cálculo sobre los otros candidatos se obtienen los valores de C-value que se muestran en la tabla 3.12. También se muestra en esta tabla la posición que cada uno de los candidatos ocupa en la lista completa ordenada por el C-value.

En esta tabla se puede observar que si bien *unidad de disco duro* es un candidato con longitud alta, su poca frecuencia en el corpus le proporciona un C-value bajo. Por otro lado, aunque *disco duro* aparece 33 veces en otros candidatos de mayor longitud, su C-value no se ve seriamente afectado debido a que esto ocurre en 19 candidatos diferentes además de aparecer 56 veces por sí mismo.

C-value	*candidato*	*posición*
287.603	usuario	1
137.003	estación de trabajo	7
108.526	disco duro	20
91.455	sistema operativo	35
21	unidad de disco duro	217

Tabla 3.12: Candidatos con C-value calculado y posición en la lista

Sigue ahora considerar el contexto de los candidatos, calcular el NC-value. El primer paso es tomar los verbos, adjetivos y nombres en una ventana de 5 palabras de los candidatos que se encuentran en el tope de la lista de candidatos ordenada por medio del C-value[97] y calcular su peso por medio de la fórmula 2. En la tabla 3.13 se muestran de nuevo los candidatos del ejemplo pero ahora con el valor de NC-value que se les ha asignado.

En las tablas 3.12 y 3.13 se puede observar la distinta distribución de los candidatos en las listas basadas en C-value y NC-value.

Debido a que mostrar el cálculo de NC-value de alguno de estos candidatos (considérese, por ejemplo, al candidato *usuario* con 289 apariciones en el corpus que, suponiendo que en cada ocurrencia tenga sólo cuatro palabras significativas en su ventana de contexto, tendría alrededor de 1200 palabras de contexto con un peso asignado) a continuación se da un ejemplo ficticio de cálculo de NC-value.

[97] Si bien no hay una regla para determinar cuántos elementos tiene el llamado “tope de la lista”, se ha observado que dividir la lista en cinco conjuntos del mismo tamaño y tomar el que contiene los valores de C-value más altos proporciona buenos resultados.

NC-value	*candidato*	*posición*
241.662	usuario	1
111.859	estación de trabajo	8[98]
89.060	disco duro	20
75.229	sistema operativo	35
17.153	unidad de disco duro	228[99]

Tabla 3.13: Candidatos con NC-value calculado

Supóngase que el candidato *interfaz*, cuyo C-value es de 11.74, ocurre con las palabras de contexto mostradas en la tabla 3.14, en la que se incluye su peso como palabra de contexto y el número de veces que aparece con este candidato.

palabra	*peso*	*ocurrencias*
usuario	5.3	2
simple	0	1
comando	2.23	1
sencilla	4.47	3
accederse	0	1

Tabla 3.14: Palabras de contexto del candidato interfaz

El cálculo del factor de contexto del candidato interfaz se muestra a continuación:

$$weight(interfaz) = (5.3 * 2) + (2.23 * 1) + (4.47 * 3)$$
$$weight(interfaz) = 26.24$$

Por lo tanto, el NC-value de interfaz quedaría como:

$$NC\text{-}value('interfaz') = (0.8 * 11.74) + (0.2 * 26.24)$$
$$NC\text{-}value('interfaz') = 14.64$$

Lo último que queda por señalar es que, a diferencia de la implementación original de este algoritmo, en la que todas las variantes de los candidatos se manejan de manera independiente[100], sin importar que se trate, por ejemplo, de la versión en singular y plural de un mismo candidato, como en *disco duro* y *discos duros*; en este caso, la lista se conforma por los lemas de los candidatos, por lo que *discos duros* es sólo una ocurrencia más del candidato canónico *disco duro*.

[98] La posición del candidato *estación de trabajo* empeoró debido a que las palabras de contexto del candidato *servidor* (que en la lista ordenada por C-value ocupaba el octavo lugar y en la de NC-value ocupa el séptimo) le proporcionaron un factor de contexto más favorable.

[99] Candidatos como *chip*, *notebook* y *partición* aparecían debajo de este candidato en la lista de C-value y lo superaron en la de NC-value.

[100] Esto puede ser observado en la página 10 de Frantzi, et. al., op. cit. en la que se muestra una tabla con la salida de una corrida del extractor que contiene dos candidatos distintos llamados *B CELL* y *B CELLS*.

En el siguiente capítulo, en el que se presenta la evaluación del prototipo de extractor desarrollado, se mostrarán los beneficios de utilizar la forma canónica de los candidatos para realizar los distintos conteos y otras consideraciones.

Además, en el apéndice 3, llamado “Salida proporcionada por el prototipo”, se pueden observar los distintos archivos que el extractor proporciona como salida.

Capítulo 4. Evaluación

Donde pueden observarse extracciones realizadas sobre distintos corpus de prueba y se evalúa el extractor automático en términos de precisión y recuerdo.

Si bien Chomsky mostró que las máquinas de estados finitos no proporcionan buenos resultados en el procesamiento de lenguaje natural en general [Manning y Schütze 2001], se ha visto que sí son de utilidad en el área de la extracción de información. Esto se debe a que la función de un autómata es aceptar o rechazar una cadena de símbolos que pertenece a un lenguaje dentro de un alfabeto definido. En este caso, el alfabeto está conformado por las categorías gramaticales de las palabras (NC, ADJ, PREP, etc.) y las combinaciones de ellas que pertenecen al lenguaje son aceptadas por medio de las reglas lingüísticas.

Como se ha señalado a través de este trabajo, una gramática formal (que en este caso hace el papel de un filtro lingüístico) no es suficiente para dar una salida satisfactoria debido a que los patrones lingüísticos que presentan los términos no son exclusivos de ellos, por lo que se requiere de una etapa estadística que al menos ordene la salida con base en el potencial de cada uno de los elementos de ser relevante, de ser un término[101]. Con todos estos aspectos cubiertos resta finalmente analizar el desempeño del extractor.

¿Cómo evaluar la salida de un extractor de términos? Lo que importa es saber si todos los términos que se encuentran en un texto son obtenidos por medio del extractor, aquellos términos que sean ignorados son un factor negativo. Por otro lado, se espera que todos los candidatos que hayan sido extraídos sean verdaderos términos. Si el extractor obtiene candidatos que en realidad no son términos, también se trata de un factor negativo. La mejor forma de medir la calidad en la salida de un extractor de términos, de manera cuantitativa, es con las medidas de precisión y recuerdo.

Estas dos palabras han sido mencionadas en algunas ocasiones a lo largo de este documento, sin embargo, no han sido definidas formalmente todavía. Debido a que son el instrumento principal para la evaluación que se hará al extractor, se abordan a continuación de manera sencilla.

Si bien la precisión (precision) y el recuerdo (recall) son medidas que fueron creadas para evaluar sistemas de recuperación de información (en los que se espera obtener como el resultado de una petición un conjunto de documentos), también pueden ser utilizadas para evaluar aplicaciones de extracción de información, como es el caso de un extractor de términos. Por ello, a continuación se explica a la precisión y al recuerdo en el ámbito de la extracción de términos.

Considérese el diagrama en la figura 4.1[102]. $|T|$ es el conjunto de los términos que hay en un texto y $|S|$ es el conjunto de los candidatos a término proporcionado por la aplicación. Por ende, la intersección de los conjuntos $|T| \cap |S|$ contiene a todos los términos del documento que fueron hallados de manera automática. Es con estos dos conjuntos y su intersección que se definen en términos de porcentaje la precisión y el recuerdo.

[101] Ya se observó en el capítulo anterior que, si se considera conveniente, se puede realizar una discriminación de candidatos de acuerdo con los resultados de la etapa estadística.

[102] Diagrama adaptado de Baeza-Yates, et. al., op. cit., p. 75.

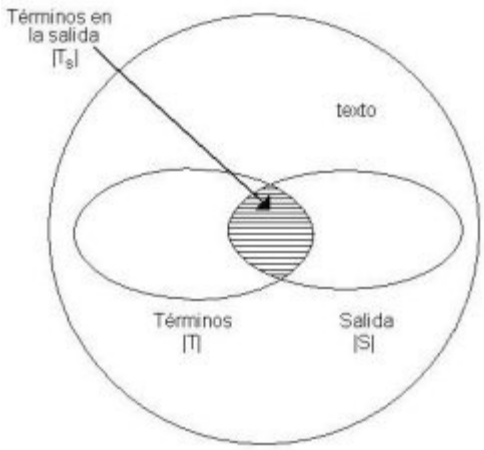


Fig. 4.1: Conjuntos |T| y |S| en el texto

El recuerdo es el conjunto de términos relevantes que son recuperados con respecto al número total de términos que existen en un documento analizado. Se expresa matemáticamente por medio de la fórmula 5.

$$Recuerdo = \frac{|Ts|}{|T|} \quad (5)$$

donde: $|Ts|$ es la cantidad de elementos en la intersección de los conjuntos $|T|$ y $|S|$
$|T|$ es la cantidad de términos en el documento

Por otro lado la precisión, cuya representación matemática se muestra en la fórmula 6, sirve para calcular qué proporción de los candidatos a término obtenidos de manera automática son realmente términos.

$$Precision = \frac{|Ts|}{|S|} \quad (6)$$

donde: $|Ts|$ es la cantidad de elementos en la intersección de los conjuntos $|T|$ y $|S|$
$|S|$ es la cantidad de candidatos a término obtenidos de manera automática

De manera más general, se puede considerar que el recuerdo mide qué porcentaje de los elementos relevantes en un universo han sido recuperados, mientras que la precisión determina cuántos de los elementos en el conjunto recuperado son realmente relevantes.

Al observar la información que se requiere para el cálculo de precisión y recuerdo, se puede comenzar a inferir la necesidad de contar con una referencia con la que se compare la salida generada de manera automática, sobre todo para el caso del recuerdo.

Cuando se habla de recuperación de información, la referencia está compuesta normalmente por un conjunto de documentos que se sabe de antemano que son relevantes para cierta búsqueda. En el caso de la extracción de información y particularmente en el de la extracción de términos lo que se requiere son los términos que hay en un texto, los cuales deben ser obtenidos de manera manual.

La muestra de términos para la evaluación de la aplicación es obtenida de un conjunto reducido de documentos debido a la evidente limitación, sobre todo en tiempo, de una persona para obtener los términos en una gran cantidad de documentos[103]. Por eso se ha seleccionado un

[103] Debe tomarse en cuenta, por ejemplo, que en una de las extracciones manuales que se realizaron una persona

pequeño conjunto de documentos del Corpus de Informática en Español para hacer una primera evaluación y una muestra de contextos definitorios, tomados prinicipalmente del Corpus Lingüístico de Ingeniería, para observar el desempeño del extractor en una aplicación real, particularmente la primera que se le dará al extractor.

4.1 Evaluación del extractor con el Corpus de Informática en Español

Debido a las limitaciones ya señaladas, no es posible hacer una evaluación sobre una cantidad importante de texto. Por ello, se han tomado al azar diecinueve documentos del Corpus de Informática en Español que contienen un total de 15,992 palabras. Dichos documentos presentan distinto nivel de especialidad y cuentan con información de carácter académico, técnico e incluso publicitario.

Para realizar esta evaluación se realizó una extracción manual de la terminología hallada en estos documentos. Cabe mencionar que originalmente se había pedido a dos personas[104] que luego de documentarse sobre una técnica de extracción de términos[105] realizaran la extracción manual. Sin embargo, esta extracción no dió buenos resultados debido a la falta de interacción con un experto del área de especialidad tratada y a la poca experiencia en la tarea de extracción de términos[106]. La tabla 4.1 muestra la cantidad de términos obtenidos en la extracción manual incluyendo su patrón gramatical.

categoría	*términos*	*categoría*	*términos*
ACRNM	54	NC PDE ACRNM	12
ACRNM NC	2	NC PDE ADJ NC	4
NC^{+}	153	NC PDE NC	83
NC ACRNM	52	NC PDE NC ADJ	20
NC ACRNM ADJ	4	NC PDE NC PDE NC	4
NC ADJ+	91	V	15
NC ADJ PDE NC	6	OTROS[107]	22

Tabla 4.1: Los patrones de los términos extraídos manualmente

Las categorías gramaticales de los componentes de los términos fueron determinadas a mano y no siempre coinciden con las asignadas por el etiquetador automático. Con esta lista de referencia, se puede calcular la precisión y recuerdo de la salida obtenida por el extractor.

Si bien el algoritmo para la extracción de términos es fijo, los parámetros de extracción pueden ser modificados. Uno de los factores que se pueden variar es el filtro lingüístico. El

tardó más de tres horas en extraer la terminología de un documento de 2,381 palabras.

104 Estudiantes de Lengua y Literaturas Hispánicas y servidoras sociales del Grupo de Ingeniería Lingüística

105 L'homme, et. al., op. cit.

106 Reflejo de ello es que de los 459 términos que fueron localizados por ellas sólo 283 eran verdaderos términos (por lo que su precisión fue de 0.6165) cuando en otra extracción se encontraron más de 520 (por lo que su recuerdo sería de 0.5442).

107 Incluye patrones como ACRNM NC, ACRNM PDE NC, NC ADJ ACRNM, NC ADJ CC ADJ, NC ADJ PDE NC ADJ, NC NC ACRNM, NC PDE ACRNM ADJ.

primer conjunto de reglas, que constituirán el filtro cerrado, cuenta únicamente con los tres patrones que con mayor frecuencia presentan los términos en el español (no únicamente los de computación) tal como fue mostrado en el apartado 3.3. Las tres reglas son:

a) <NC><ADJ>
b) <NC><PREP><NC>
c) <NC><NC>$^{+}$

La regla c toma en cuenta una de las debilidades del algoritmo C-value/NC-value original: la poca efectividad en el manejo de candidatos a término de una sola palabra. Es por ello que toma como candidatos secuencias con al menos dos nombres comunes en lugar de sólo uno.

El segundo conjunto de reglas, que constituyen el filtro abierto, es una simplificación de las mostradas en el apartado 3.5.1.1 y aceptan prácticamente todos los patrones que mostraron los términos en la extracción manual mostrada en el mismo apartado 3.3[108].

a) (<N.><PDE><ADJ>$^{?}$ <N.|NMEA|PE>$^{+}$ <ADJ>$^{?}$)$^{+}$
b) <NC><ADJ>$^{+}$(<ACRNM|N.>|(<CC><ADJ>)|(<PDE><N.>))*
c) (<NC><PE>$^{?}$)$^{?}$<ACRNM>(<PDE><NC>)*
d) <N.|PE>$^{+}$(<CODE>$^{+}$|(<PDE>?<PE|N.>)$^{+}$)$^{?}$

El segundo factor que se puede modificar es la lista de paro. En este caso las opciones son simplemente utilizarla o no.

Se realizaron cuatro extracciones variando estos dos parámetros: el tipo de filtro (abierto o cerrado) y el uso u omisión de la lista de paro. Las combinaciones de estos parámetros han dado como resultado cuatro experimentos cuyos resultados se presentan en la tabla 4.2.

Extracción	*Filtro*	*Lista de paro*	*Términos extraídos a mano*	*Candidatos extraídos*	*Candidatos que son términos*	*Precisión*	*Recuerdo*
A[109]	abierto	no	520	1,867	430	0.230	0.826
B	abierto	sí	520	1,554	413	0.265	0.794
C	cerrado	no	520	1,000	241	0.240	0.463
D	cerrado	sí	520	850	262	0.308	0.503

Tabla 4.2: Comparación de los resultados de las extracciones

El utilizar un filtro abierto beneficia al recuerdo, ya que debido a su flexibilidad pocos términos pasan desapercibidos. Sin embargo, un filtro de este tipo perjudica a la precisión pues muchos de los candidatos obtenidos resultan no ser verdaderos términos. Como es de esperarse, el contar con un filtro cerrado, que no sea tan flexible en cuanto a los patrones admitidos, resulta en un mejor valor de precisión, pero deja escapar algunos términos reales del texto, lo que perjudica al recuerdo. Tanto las reglas lingüísticas como la lista de paro pueden ser modificadas dependiendo de la precisión y el recuerdo que se desee obtener.

En el caso del recuerdo, éste decrece, tal como se espera, mientras los requisitos para los candidatos se hacen más extrictos. Sin embargo, puede observarse que cuando se utiliza el filtro

[108] Vale la pena recordar que por el momento no se consideran los términos que son verbos.

[109] En el apéndice 4 se ofrece una lista con los verdaderos términos que se encuentran en la lista de salida de este experimento.

cerrado el valor de recuerdo es mejor al aplicar la lista de paro que al no hacerlo. Este "curioso comportamiento" se debe a la adaptación que se hizo al algoritmo en la etapa de la aplicación de la lista de paro en la que, en vez de eliminar totalmente aquellos candidatos que cuentan con palabras halladas en la lista de paro, se mantienen en ella luego de la eliminación de las palabras no deseadas dentro de ellos, incluyendo a aquellas que interactúan directamente con ellas[110]. Con ello, algunos malos candidatos que fueron detectados por el filtro lingüístico, como *teclado profesional* y *sistema anterior*, luego de pasar por la lista de paro se convirtieron en *teclado* y *sistema*, que son términos reales de una palabra que sí fueron entregados en la lista de salida.

El comportamiento de la precisión y el recuerdo se muestra gráficamente en la figura 4.2. En ella se puede observar que el filtrar a los candidatos por medio de la lista de paro mejora la precisión (experimento B con respecto a A y D con respecto a C). Por otro lado, utilizar un filtro lingüístico abierto (experimentos A y B) proporciona un mejor nivel de recuerdo con respecto a uno cerrado (experimentos C y D)[111].

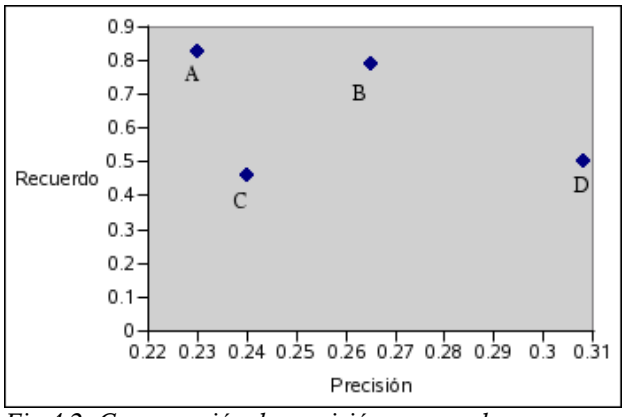


Fig.4.2: Comparación de precisión y recuerdo

Resulta interesante considerar que la extracción manual de términos tomó prácticamente un día de trabajo y que el nivel de la calidad de la extracción durante todo este tiempo no es constante. Por otro lado, el programa es capaz de realizar la extracción en unos minutos. Con el tiempo y esfuerzo que se ahorra con el programa, parece que el ruido en la salida puede ser tolerado.

4.2 Desempeño del extractor en el análisis de contextos definitorios

El objetivo principal de este extractor, en principio, es la detección de los términos que se hallan dentro de un contexto definitorio[112]. En esta aplicación, se ha determinado no utilizar ninguna lista de paro por tres razones:

1. La densidad de términos dentro de un contexto definitorio correctamente delimitado es alta, pues dentro de él no sólo se presenta el término que está siendo definido sino también los que lo definen.
2. La lista de paro es dependiente del área de especialidad tratada, sin embargo, los términos

[110] Ver apartado 3.5.1.1.

[111] En general, como ocurrió en este experimento, cuando la precisión es alta el recuerdo resulta bajo y viceversa, es decir, son dos factores que se mantienen en equilibrio pero sin ser totalmente complementarios.

[112] Véase el apartado 1.3.

de un área no siempre son definidos exclusivamente por medio de términos de ésta misma. Además, los contextos definitorios analizados en esta evaluación no pertenecen a una sola área de especialidad.

3. En el análisis de contextos definitorios las marcas y demás nombres propios sí son considerados relevantes, por lo que se constituyen pseudo-términos válidos[113].

Si bien el corpus de contextos definitorios está en este momento en etapa de construcción dentro del Grupo de Ingeniería Lingüística, se ha obtenido una pequeña muestra de estos contextos para observar el desempeño del extractor. Dicha muestra está compuesta por 33 contextos definitorios que contienen 1,341 palabras en total.

A continuación se presentan cuatro ejemplos de contextos definitorios. En la columna de la izquerda se muestra el contexto definitorio con los términos localizados a mano (en negritas) mientras que en la columna de la derecha se hallan los contextos con los términos identificados de manera automática; los términos identificados correctamente son mostrados en negritas mientras los falsos positivos se marcan subrayados y los falsos negativos en cursivas[114]:

Un **conjunto** de **equipo eléctrico** utilizado para un fin determinado se le conoce con el nombre de **SUBESTACIÓN ELÉCTRICA**.	Un **conjunto** de **equipo eléctrico** utilizado para un fin determinado se le conoce con el nombre de **SUBESTACIÓN ELÉCTRICA**.
Se conoce como **transformador de corriente** a aquél cuya función principal es cambiar el valor de la **corriente** de uno más o menos elevado a otro con el cual se puedan alimentar **instrumentos de medición**, **control** o **protección**, como **ampérmetros**, **wáttmetros**, **instrumentos registradores**, **relevadores de sobrecorriente**, etc. Su construcción es semejante a la de cualquier tipo de **transformador**, ya que fundamentalmente consiste de un **devanado primario** y un **devanado secundario**.	Se conoce como **transformador de corriente** a aquél cuya función principal es cambiar el valor de la **corriente** de uno más o menos elevado a otro con el cual se puedan alimentar **instrumentos de medición**, **control** o **protección**, como **ampérmetros**, **wáttmetros**, **instrumentos registradores**, **relevadores de sobrecorriente**, etc. Su construcción es semejante a la de cualquier tipo de transformador, ya que fundamentalmente consiste de un **devanado primario** y un **devanado secundario**.
La **Teoría del Buque** que estudia el **barco** considerado como un **flotador** que se mueve en un **líquido** y se refiere exclusivamente a las **formas exteriores** del mismo, que son las que determinan sus **condiciones de resistencia** al **movimiento** y **estabilidad**.	La *Teoría del Buque* que estudia el **barco** considerado como un **flotador** que se mueve en un *líquido* y se refiere exclusivamente a las **formas exteriores** del mismo, que son las que determinan sus **condiciones de resistencia** al **movimiento** y **estabilidad**.
Para esta **dirección** el **paisaje** se concibe como una **entidad espacial**, un ensamble de **ecosistemas** en **interacción** centrando su interés en los diferentes fenómenos relacionados con el intercambio entre los **sistemas** y la **heterogeneidad espacial**.	Para esta **dirección** el **paisaje** se concibe como una **entidad espacial**, un ensamble de **ecosistemas** en **interacción** centrando su interés en los diferentes fenómenos relacionados con el **intercambio** entre los **sistemas** y la **heterogeneidad espacial**.

Términos como *líquido* no fueron detectados debido a que el etiquetador no les asignó la etiqueta correcta. En este caso *líquido* fue considerado un adjetivo. En el caso de los falsos términos detectados, como *función principal* y *valor*, se podría optar por aplicar una lista de paro que los elimine asumiendo las consecuencias que ya fueron señaladas (en este caso, la posibilidad de perder términos de carácter financiero si componentes de los términos de esta área como *interés*

[113] Para más información sobre los puntos 2 y 3 véase el apartado 3.5.1.1.

[114] Los falsos positivos son aquellos candidatos identificados que no son verdaderos términos. Los falsos negativos son los términos hallados en el texto que no fueron detectados.

son agregados a la lista de paro).

En este pequeño experimento la precisión fue de 0.7601 mientras que el recuerdo obtenido tuvo un valor de 0.9121.

Es interesante observar el comportamiento de la precisión a través de la lista ordenada de salida del extractor. En la figura 4.3 se muestra la evolución de este valor tomando en cuenta el cálculo de la precisión sobre conjuntos de candidatos que contienen desde un elemento hasta los 246. Esta gráfica muestra que, tal como lo ofrece el algoritmo implementado, los verdaderos términos dentro de los candidatos tienden a aparecer en la parte superior de la lista de salida.

Se puede observar que al evaluar la precisión de los conjuntos que tienen desde 1 hasta 7 elementos el valor de la precisión se mantiene en 1, pero que al tomar el conjunto con los primeros 8 candidatos hay una caída considerable en la precisión (que va de 1 a 0.875). La razón de esta caída tan repentina es que en un conjunto con tan pocos elementos, cualquier fallo afecta de manera importante al cálculo del valor de la precisión. Sin embargo, más adelante se observa que entre más candidatos son tomados en cuenta, el valor de la precisión deja de tener variaciones tan drásticas.

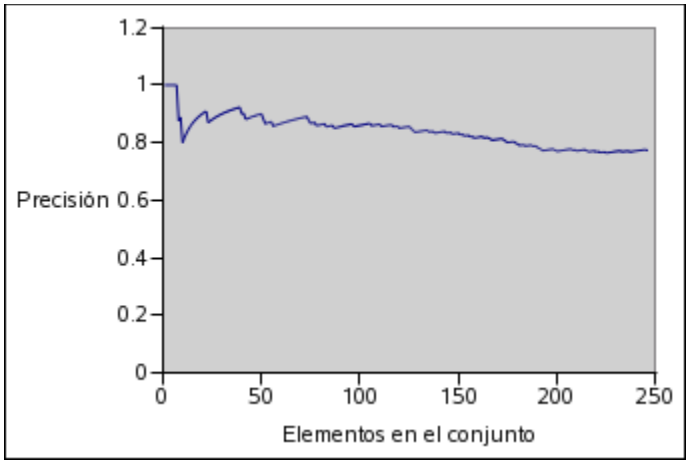


Fig. 4.3: Comportamiento de la precisión a través de la salida

El desempeño del extractor en el caso de la detección de los términos al interior de un contexto definitorio, al igual que para otras aplicaciones, puede mejorar modificando tanto las reglas del filtro lingüístico como la lista de paro.

4.3 Algunas consideraciones sobre la evaluación del extractor

El algoritmo C-value/NC-value ha sido implementado e incluso modificado en varias ocasiones [Ismail 2005, Li et. al. 2007], sin embargo, en este trabajo no se ha realizado una comparación explícita del prototipo desarrollado con algún otro. La razón es simple: debido a que se desearía evaluar el resultado de las adaptaciones hechas al algoritmo, sería necesario comparar los resultados con los obtenidos por otros prototipos bajo las mismas condiciones, como son tener el mismo corpus de entrada, el mismo etiquetador y la misma lista de paro.

Sin embargo, sin dejar de lado que estas condiciones no se cumplen, se ha optado por

comparar nuestros resultados con los obtenidos por Li et. al. [2007], con su prototipo desarrollado para extraer términos del área de tecnología de la información en chino.

Dicho prototipo sustituye la etapa de NC-value por una de verificación de términos. A grandes rasgos, esta etapa se basa en dos enfoques. El semántico busca las palabras de contexto del candidato en una lista de términos previamente obtenida. La existencia con cierta frecuencia de las palabras de contexto entre estos términos implica que el candidato es un término. El sintáctico se basa en el patrón que suelen presentar los términos y sus palabras de contexto. Por ejemplo, señalan que un término cuya categoría sea nombre suele estar seguido por un verbo y otro nombre, por lo que este patrón es indicador de la existencia de un verdadero término.

El experimento reportado, realizado sobre un corpus de 16 artículos con 1,500,000 carácteres y una lista de 288,000 términos de una palabra, obtuvo una precisión de 0.67 y un recuerdo de 0.42.

Sin contar con esta etapa de validación que requiere una lista de términos validados y los patrones presentados no sólo por los términos sino por su contexto, en el experimento B, realizado sobre el corpus de informática en español[115], nosotros hemos obtenido una precisión de 0.265 y un recuerdo de 0.794. Considerando que la precisión es mucho menor debido a que se incluyen candidatos a término de una sola palabra, el extractor muestra un buen desempeño sin necesitar recursos adicionales como los usados en el otro prototipo.

Además, la evaluación de extractores de términos y otras aplicaciones de extracción de información suelen ser muy subjetivas. El hecho de que una evaluación muestre que un extractor de términos cumple con las expectativas o resulta deficiente no implica forzosamente que al aplicarlo tendrá el mismo desempeño.

Un aspecto importante a considerar es que, en línea con lo dicho por R. Manning que considera que "el corpus reflejará el material con el cual fue construido"[116], la lista de candidatos obtenida por el extractor reflejará el tipo de texto que sea analizado por éste. Si el corpus de entrada es altamente especializado, la salida contendrá pocos errores. Sin embargo, mientras más general sea el carácter de los documentos analizados, más basura será arrojada por el extractor.

Por último, cabe señalar lo dicho por K. Kageura [2006] "los extractores de términos no son óptimos, ya se sabe que no son capaces de dar la salida esperada. ¿Por qué evaluar un extractor si sabemos que no va a obtener una buena calificación? La evaluación de un extractor de términos no resulta favorable si su precisión y recuerdo son buenos. La evaluación resulta favorable si la salida que proporciona es útil para el usuario".

En el caso del proyecto de *Extracción de conceptos en textos de especialidad a través del reconocimiento de patrones lingüísticos y metalingüísticos*, la detección de los términos ubicados al interior de los contextos definitorios que se ha realizado en el corpus que está hasta ahora disponible, ha mostrado ser de utilidad para el proyecto. Por ello, la evaluación del desempeño del extractor en esta aplicación puede ser considerada favorable.

[115] Tabla 4.2.
[116] Manning, et. al., op. cit. p. 21

Conclusiones y trabajo a futuro

En este trabajo se ha presentado la adaptación del algoritmo para la extracción automática de términos C-value/NC-value para extraer candidatos a término de cualquier longitud en documentos en español.

Las principales acciones realizadas en este proceso han sido:

- El análisis de los patrones lingüísticos presentados por los términos en español en las áreas de ingeniería y computación para la generación de las reglas lingüísticas apropiadas para su detección.
- La generación de una lista de paro con más de 200 palabras (nombres y adjetivos) que han mostrado una alta frecuencia de aparición en los corpus utilizados y que no se espera que constituyan términos de las áreas de especialidad tratadas.
- La modificación en la aplicación de la lista de paro para que aquellos candidatos que incluyeran palabras contenidas en ella no fueran eliminados por completo sino que se les hiciera un tratamiento de eliminación específica de estas palabras.
- La modificación a la fórmula de cálculo de C-value para que el algoritmo sea capaz de extraer candidatos a término de una sola palabra.
- El uso de fronteras móviles para la ventana de contexto de cada uno de los candidatos, teniendo una longitud por defecto de cinco palabras, pero siendo posible reducirla en caso de contener signos de puntuación.

La realización de este trabajo ha permitido llegar a varias conclusiones. A continuación se muestran las más relevantes.

En el caso del filtro lingüístico, encargado de localizar a los candidatos a término, una decisión de gran impacto en la salida es el usar un filtro abierto o uno cerrado, lo que depende en buena parte de la aplicación en turno. En el caso del análisis de la terminología de un área de especialidad se mostró la conveniencia de soportar el ruido generado por un filtro abierto con el objetivo de perder la menor cantidad de términos reales. Por otro lado, para la clasificación automática de documentos es más pertinente implementar un filtro cerrado que ofrezca una mayor precisión a la salida permitiendo obtener una clasificación más certera.

Ya sea un filtro cerrado o abierto, se ha visto que el hecho de que las reglas lingüísticas sean capaces de reconocer candidatos a término de una sola palabra mejora dramáticamente el recuerdo con el consecuente decremento de la precisión.

En línea con las etapas del proceso para la extracción de términos, la siguiente es la aplicación de la lista de paro. El utilizar una lista de paro que, en lugar de eliminar por completo a los candidatos que contengan palabras halladas en la lista, haga una eliminación específica de estas palabras y las que actúan directamente sobre ellas, mejora la salida debido a la posibilidad del descubrimiento de candidatos que de otra forma hubieran pasado desapercibidos y a que permite generar un mejor orden en la lista de salida.

Sin embargo, el usar o no esta lista de paro depende de la aplicación y del área de especialidad del corpus analizado. En el caso particular del análisis de los términos hallados en un contexto definitorio no es conveniente utilizar una lista de paro debido a que no sólo los términos del área de especialidad tratada son relevantes, pues en ocasiones, al definir un término, se usan términos de otras áreas. Además, las diversas entidades nombradas incluidas en un contexto definitorio pueden ser consideradas relevantes. Por otro lado, si se desea analizar un corpus de un área de especialidad nueva, es necesario generar una nueva lista de paro, pues éstas son dependientes del área, lo que conlleva un trabajo previo con un experto del área y no siempre

existe la posibilidad de realizar este trabajo previo.

Hablando ahora sobre el tipo de algoritmo implementado, un algoritmo híbrido para la extracción de términos, debido a su etapa estadística, demanda que el corpus de entrada contenga una buena cantidad de texto (miles de palabras) para ofrecer un buen ordenamiento de los candidatos en la lista de salida. Por supuesto, la longitud del corpus de entrada no tiene un umbral mínimo, pero es recomendable que sea grande.

Ahora que se habla un poco del corpus, es importante decir que si se desean obtener valores de precisión y recuerdo aceptables, se deben incluir en el corpus de entrada solamente documentos de carácter técnico y científico con un buen nivel de especialización. La inclusión de documentos de corte publicitario e incluso de divulgación generan mucho ruido.

Finalmente, cabe señalar que en este desarrollo ha sido de vital importancia la interacción, incluso traspasando fronteras, entre las disciplinas de la lingüística y la computación, lo cual en ocasiones es por demás complicado. Sin embargo, si se desea obtener resultados satisfactorios en esta interdisciplina, la relación debe ser estrecha y equilibrada.

Esto en cuanto a las conclusiones. Resta hablar de la etapa en la que se encuentra el proyecto actualmente y lo que queda por hacer.

El primer paso es, ya contando con el corpus de contextos definitorios completo, interactuar de una manera más fuerte con expertos lingüistas y terminólogos para comenzar a hacer la detección de términos del material, acción que contribuirá a la generación de un corpus que tenga sus componentes identificados. Cabe señalar que se plantea que este corpus contenga texto semiestructurado, pues cuenta con etiquetado XML para delimitar a cada uno de los contextos y sus elementos constitutivos.

Además, se creará una interfaz orientada a web para el extractor con el objetivo de que cualquier usuario de los corpus con los que cuenta el GIL tenga acceso a sus listas de candidatos a término correspondientes.

Por otro lado, el extractor se utilizará para extraer los términos en un corpus compuesto por miles de resúmenes de artículos de medicina. El objetivo es extender una ontología generada a mano a partir de cada uno de estos artículos. Esto permitirá clasificarlos con base en el tema de especialidad que abordan y sobre esta clasificación será posible obtener mejores resultados de las búsquedas realizadas.

El llenado automático de bases de conocimiento por medio de la extracción de términos será uno de los temas abordados en mi investigación de doctorado, la cual buscará la generación y explotación de bases de conocimiento para sistemas de búsqueda y de pregunta-respuesta.

Como puede observarse, éste ha sido apenas el inicio de un largo trabajo.

Bibliografía

The Association for Computational Linguistics. http://www.aclweb.org, 2006, consulta: diciembre de 2006.

Atserias J., Carmona J., Castellón I., Cervell S., Civit M., Màrquez L., Martí M.A., Padró L., Placer R., Rodríguez H., Taulé M. y Turmo J. 1998. "Morphosyntactic Analysis and Parsing of Unrestricted Spanish Text". *Proc. LREC'98 1st International Conference on Language Resources and Evaluation*. Granada, Spain,.

Atserias J., Casas B., Comelles E., González M., Padró L. y Padró M. 2006. "FreeLing 1.3: Syntactic and semantic services in an open-source NLP library". *Proc. LREC'06 5th International Conference on Language Resources and Evaluation.* Genoa, Italy.

Bach C., Saurí R., Vivaldi J., Cabré M. T. 1997. *El Corpus de l'IULA: descripció.* IULA, Universidad Pompeu Fabra, Barcelona. Informe 17.

Bourigault D. 1992. "Surface analysis for the extraction of terminological noun phrases". *Actes de COLING-92*, Nantes, pp. 22-28.

Bourigault D. *1994. LEXTER, un Logiciel d'EXtraction de TERminologie. Application à l'acquisition des connaissances à partir de texts*. PhD Thesis. Paris: Ecole des Hautes Etudes en Sciences Sociales, Paris.

Bourigault D., Fabre C., Frétot C., Jacques M. P., Ozdowska S. 2005. "Syntex, analyseur syntaxique de corpus". *TALN-2005*, Dourdan.

Brill E. 1992. "A simple rule-based part of speech tagger". Proc. ANLP'92. 3rd Conference of Applied Natural Language Processing.

Cabré M.T. 1995. *La terminología. Teoría, metodología, aplicaciones*, Antártida/Empúries, Barcelona.

Cabré M. T. 1999. "Una nueva teoría de la terminología: de la denominación a la comunicación", *La terminología. Representación y comunicación*, Universitat Pompeu Fabra, Barcelona.

Cabré M. T., Freixa J., Lorente M., Tebé C. 2001. *Textos de terminólogos de la Escuela Rusa*. Barcelona: IULA, 170 pp.

Cabré M. T., Estopà R., Vivaldi J. 2001. "Automatic term detection. A review of current systems". Bourigault, D., Jacquemin, C., L'Homme, M. C., *Recent Advances in Computational Terminology, vol. 2*, John Benjamins Publishing Company, Amsterdam/Filadelfia.

Campo A. (review) *La terminología: Representación y comunicación. Elementos para una teoría de base comunicativa y otros artículos*.

Cardero A.M. 1999. "Terminología y lexicografía", *Primeras jornadas de la Red Iberoamericana de Terminología*, Bogotá, Colombia.

Cardero A. M. 2000. "En torno a la frecuencia de algunas estructuras sintácticas en terminología". VII Simpósio Ibero-Americano de Terminología: Terminología e Indústruas da Língua, Lisboa, Portugal.

Cardero, A. M. 2003. *Terminología y Procesamiento*. Universidad Nacional Autónoma de México, Escuela Nacional de Estudios Profesionales Acatlán.

Centre de communicatión écrite, http://www.cce.umontreal.ca/robertdubuc.htm. Consulta: diciembre de 2006

Daille B. 2003. "Conceptual structuring through term variations". *Proc. ACL Workshop on MultiWord expressions: Analysis, Acquisition and Treatment.*

Drouin, P. 2003. "Term extraction using non-technical corpora as a point of leverage". *Terminology, Vol. 9, Number 1*, John Benjamins Publishing Company, pp. 99-115.

Dubuc R., *Manuel pratique de terminologie*, Linguatech éditeur, Canadá, 1992.

Enguehard C. 1993. "Acquisition de terminologie à partir de gros corpus". *Informatique & Langue Naturelle, ILN'93*, Nantes, pp.373-384.

Enguehard C., Pantera L. 1994. "Automatic Natural Acquisition of a Terminology". *Journal of Quantitative Linguistics*, pp. 27-32.

Fahmi I. 2005. "C-.value method for multi-word term extraction", In seminar in Statistics and Methodology.

Frantzi K., Ananiadou S. y Mima H. 2000. "Automatic recognition of multi-word terms: the C-value/NC-value method", *Int. J. on Digital Libraries* 3(2), pp. 115-130.

Grishman R. 1986. *Computational linguistics: an introduction.* Cambridge University Press, New York, 193 pp.

Heid U., Jauss S., Krüger K. y Hohmann A. 1996. "Term extraction with standard tools for corpus exploration - Experience from German". *Proc TKE '96. Terminology and Knowledge Engineering.* Frankfurt, pp. 139-150.

Heid U. 1999. "A linguistic bootstrapping approach to the extraction of term candidates from German text". *Terminology Vol. 5(2)*, John Benjamins: Amsterdam, pp. 161-181.

Jackson P., Moulinier I. 2002. *Natural Language Processing for Online Applications: Text Retrieval, Extraction and Categorization.* John Benjamins Publishing Company, Amsterdam/Filadelfia.

Ji L., Sum M., Lu Q., Li W., Chen Y. 2007. *Chinese Terminology Extraction Using Window-Based Contextual Information.* Proc. 8th International Conference CICLing, México, pp. 62-74.

Jurafsky D., Martin J. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*, Prentice Hall,, p. 10.

JustesonJ. S., Katz S. M. 1995. *Technical terminology: some linguistic porperties and an algorithm for identification in text*, en Natural Language Engineering.

Kageura K., Abekawa T. 2006. *Library and Informatios Science Course.*Séminaires RALI-OLST, Université de Montréal.

L'Homme M. C. 2004. "Lexical semantics for terminology, A case study:dictionary of computing". *Lengua y sociedad, investigaciones recientes en lingüística aplicada*, Universidad de Valladolid, pp. 233-245.

L'Homme M. C., Bae H.S. 2006. *A Methodology for Developing Multilingual Resources for Terminology.* Proc. LREC 2006. Language Resources and Evaluation, pp. 22-27, Genoa, Italia.

L'Homme M. C., Drouin P. 2006. *Corpus de Informática en Español*. Groupe Éclectik, Université de Montréal: http://www.olst.umontreal.ca/.

Manning C., Schütze H. 2001. *Foundations of Statistical Natural Language Processing*, The MIT Press, Foutrh Printing, pp. 680.

McEnery, T. & Wilson, A. 1996. *Corpus linguistics.* Edinburgh: Edinburgh University Press.

Medina A., Sierra G., Garduño G., Méndez C. y Saldaña R. 2004. "CLI. An Open Linguistic Corpus for Enineering". *IX Congreso Iberoamericano de Inteligencia Artificial (IBERAMIA), Actas de congreso*. Tonantzintla, Puebla, pp. 203-208.

Plante, P., Dumas, L. 1998. "Le Dépoulliment terminologique assisté par ordinateur". *Terminogramme, 46*, pp. 24-28.

Reformatskii A. A. 1961. "¿Qué es el término y qué es la terminología?*", Problemas de la terminología*, Academia de Ciencias de la URSS, Moscú, pp. 46-54.

Russel S., Norvig P. 2003. *Artificial Intelligence: A Modern Approach, Second Edition*, Prentice Hall, pp. 1132.

Sager J. C. 1978. "Commentary by Prof. Juan Carlos Sager. In Guy Rondeau". Actes Table Ronde sur les Problèmes du Découpage du Terms. Montréal.

Schmid H. 1994. "Probailistic Part-of-Speech Tagging Using Decision Trees". International Conference on New Methods in Language Processing, Manchester, UK.

Sierra G., McNaught J. 2000. **"**Design of an onomasiological search system: A concept-oriented tool for terminology**".** *Terminology, Num. 1, Volumen 6*.

Sierra G. 2005. *Lingüística de corpus, curso de la Maestría en Ciencia e Ingeniería de la Computación*, UNAM, página web: http://www.iling.unam.mx/CursoCorpus/default.html.

Turing A. M. 1950. "Computing Machinery and Intelligence". *Mind 49:* 433-460.

Vivaldi J. 1995. *Proyectos del IULA: El corpus técnico.* Simposium español de Lingüística. Instituto Cervantes y Universidad de Manchester, Manchester.

Wüster E. *Introducción a la teoría general de la terminología y a la lexicografía terminológica.* IULA, Barcelona, pp. 227, 2003. Impresión en español

Apéndice 1. Etiquetas de las herramientas para el etiquetado POS

Donde se presentan las etiquetas utilizadas por las herramientas de etiquetado morfosintáctico contempladas en este trabajo: TreeTagger y Freeling.

El conjunto de etiquetas de TreeTagger sigue el estándar del Pen-Treebank[117] y se muestra en la siguiente tabla:

etiqueta	*significado*	*etiqueta*	*significado*	*etiqueta*	*significado*
FS	Marcas de puntuación	INT	Pronombre interrogativo (quién, cuántas)	VEinf	Verbo estar en infinitivo
SYM	Símbolos	ITJN	Intersección (oh, ja)	VE	Verbo estar en pasado participio (estado)
ART	Artículos (un, las, la, unas)	NC	Nombre común (mesa, libro)	VHfin	Verbo haber conjugado (has, hemos)
ADJ	Adjetivos (mayores, grande)	NP	Nombre propio	VHger	Verbo haber en gerundio (habiendo)
ADV	Adverbios (muy, demasiado)	NEG	Negación	VHinf	Verbo haber en infinitivo
ALFP	Letra del alfabeto en plural (As, bes)	ORD	Ordinal (primeras, segundo)	VHadj	Verbo haber en pasado participio (habido)
ALFS	Letra del alfabeto en singular (A, b)	PAL	Contracción “al”	VLfin	Verbo conjugado (voy, cuentas)
CARD	Cardinales (1,3 2.5)	PDEL	Contracción “del”	VLger	Verbo en gerundio (mirando, programando)
CC	Conjunción (y, o)	PE	Palabra extranjera	VLinf	Verbo en infinitivo (hacer, caminar)
CCAD	Conjunción adversativa (pero)	PNC	Palabra no clasificada	VLadj	Verbo en pasado participio (hecho, roto)
CCNEG	Conjunción negativa (ni)	PPX	Clíticos y pronombres personales (nos, me, ustedes)	VMfin[118]	Verbo modal finito
CODE	Código alfanumérico (RS232)	PPO	Pronombres posesivos (mía, tuyos)	VMger	Verbo modal en gerundio
CQUE	Conjunción “que”	PREP	Preposiciones (de, a, sin)	VMinf	Verbo modal infinitivo
CSUBF	Conjunción subordinada que introduce cláusulas finitas (apenas)	QU	Cuantificadores (sendas, cada)	VM	Verbo modal en pasado participio
CSUBI	Conjunción subordinada que introduce cláusulas infinitas (al)	REL	Pronombres relativos (cuyo)	VSfin	Verbo ser conjugado (somos, eres)
CSUBX	Conjunción subordinada subespecificada para tipo subordinado (aunque)	SE	Participio “se”	VSger	Verbo ser en gerundio (siendo)
DM	Pronombres demostrativos (ésas, esta)	VEfin	Verbo estar conjugado (estoy, estás)	VSinf	Verbo ser en infinitivo
FO	Fórmula	VEger	Verbo estar en gerundio (estando)	VS	Verbo sen en pasado participio (sido)

[117] Tomada de http://www.ims.uni-stuttgart.de/ftp/pub/corpora/spanish-tagset.txt en marzo de 2007

[118] En realidad en el español no existen los verbos modales, sin embargo, han sido incluidos en el conjunto de etiquetas por los desarrolladores de esta herramienta.

Por otro lado, Freeling utiliza el formato de etiquetado EAGLES. Éste se caracteriza porque cada uno de los seis carácteres de la etiqueta tiene un significado. Por ejemplo, en el caso de la palabra *alegres*, cuya etiqueta es *AQ0CP0*, A significa adjetivo, Q calificativo, 0 sin información, C género común, P plural y 0 sin información.

A continuación, se muestran sólo las categorías principales consideradas en este esquema de etiquetado y que de hecho son las que importan para la extracción de términos[119].

etiqueta	***significado***	***etiqueta***	***significado***
A	Adjetivo	I	Interjecciones
R	Adverbio	S	Preposiciones
D	Determinante	F	Signo de puntuación
N	Nombres	C	Conjunción
V	Verbos	Z	Cifras
P	Pronombre	W	Fecha u hora

[119] Para profundizar en las demás características gramaticales consideradas en este esquema de etiquetado, consúltese http://garraf.epsevg.upc.es/freeling/doc/userman/parole-es.html

Apéndice 2. Lista de paro

Donde está la lista de paro que fue generada a partir del Corpus de Informática en Español y contiene las palabras (nombres y adjetivos) que más frecuencia presentan en este corpus y que no se espera que aparezcan en los términos del área.

accesorio	actual	adicional	administración	aero	aibo
ajuste	américa	amplio	anterior	anuncio	año
aparato	aplicación	área	aspecto	atractivo	automático
avance	ayuda	básico	batería	beneficio	blanco
brillante	bueno	caja	cálculo	calidad	cambio
campo	cantidad	característica	cintiq	color	combinación
compañía	completo	común	condición	confiabilidad	confiable
conjunto	consumidor	contenido	continuo	corporation	corporativo
costo	cuenta	definición	delgado	desempeño	detalle
día	diferente	diseño	disponible	distancia	eco
económico	efecto	ejemplo	elegante	elemento	embargo
emc	empresa	empresarial	energía	entretenimiento	equipo
espacio	estado	estrategia	etc	excelente	exclusivo
existente	experiencia	exposición	fabricante	fácil	facilidad
factor	familia	fin	flexibilidad	forma	fotográfico
fotógrafo	funcionamiento	futuro	gama	garantía	gestión
gracia	grande	grupo	herramienta	hogar	ideal
igual	importante	infraestructura	innovación	investigación	kodak
lanzamiento	lápiz	largo	latina	lente	líder
línea	liviano	llamada	lugar	luminosidad	luz
manejo	manera	mantenimiento	manual	marca	mayoría
mecanismo	medida	mejora	mercado	metálico	mismo
modalidad	modelo	moderno	momento	msi	mundial
mundo	necesario	necesidad	nuevo	olympus	opción
opcional	p2	panasonic	papel	parte	paso
película	pequeño	perfecto	peso	plano	poderoso
popular	posibilidad	posible	posición	potencia	precio
preciso	presentación	prestación	problema	productividad	producto
profesional	propiedad	propio	protección	proveedor	próximo
prueba	rango	reciente	recurso	respuesta	resultado
revolucionario	ruido	sala	segmento	sencillo	sensible
series	server	siguiente	similar	solo	solución
soporte	superior	symmetrix	tamaño	technology	técnico
tecnológico	tecra	televisión	televisor	temperatura	tiempo
tinta	tipo	toma	total	tradicional	través
tv	último	único	uso	útil	valor
variedad	ventaja	versión	vez	vida	viewsonic
wacom					

Apéndice 3. Salida proporcionada por el prototipo

En el que se muestran los distintos formatos de salida que el extractor puede de proporcionar.

Se ha considerado que en aras de que la explotación de la aplicación sea sencilla, es necesario que sea capaz de proporcionar la salida del proceso en distintos formatos.

La primer salida consiste en un archivo HTML que contiene la lista completa de los candidatos a término, obtenidos del corpus de entrada, ordenados por su NC-value. Incluye el valor de NC-value, el candidato en su versión lematizada, el candidato original que se halló en el corpus y el patrón sintáctico del candidato:

nc-value	lemmatized	candidate	POS
37.147	computadora	computadora	NC
24.186	usuario	usuario	NC
23.75	sistema operativo	sistema operativo	NC ADJ
22.667	sistema	sistemas	NC
17.316	dispositivo	dispositivo	NC
15.817	instrucción	instrucciones	NC
14.128	memoria	memorias	NC

Otra opción es proporcionar exactamente la misma información de la tabla anterior pero en un archivo de texto con extensión .txt en el que los campos están separados por ";":

NC-value;lemmatized;candidate;POS
37.147;computadora;computadora;NC
24.186;usuario;usuario;NC
23.75;sistema operativo;sistema operativo;NC ADJ
22.667;sistema;sistemas;NC
17.316;dispositivo;dispositivo;NC
15.817;instrucción;instrucciones;NC

Este archivo puede ser importado en una hoja de cálculo para realizar las manipulaciones que se consideren convenientes, como ordenar a los candidatos alfabéticamente o por su patrón sintáctico, como se muestra a continuación:

37.15	computadora	computadora	NC
3.23	computadora actual	computadoras actuales	NC ADJ
1.62	computadora Aunque	computadoras Aunque	NC NP
3.24	computadora central	computadoras centrales	NC NC
4.89	computadora de uso general	computadoras de uso general	NC PDE NC ADJ
3.29	computadora digital	computadoras digitales	NC ADJ
1.6	computadora electrónico	computadoras electrónicas	NC ADJ
2.16	computadora La denominación	computadora La denominación	NC NP NC
3.29	computadora moderno	computadoras modernas	NC ADJ
6.58	computadora personal	computadoras personales	NC ADJ
1.6	computadora personales»	computadoras personales»	NC ADJ
1.61	computadora programables	computadoras programables	NC ADJ

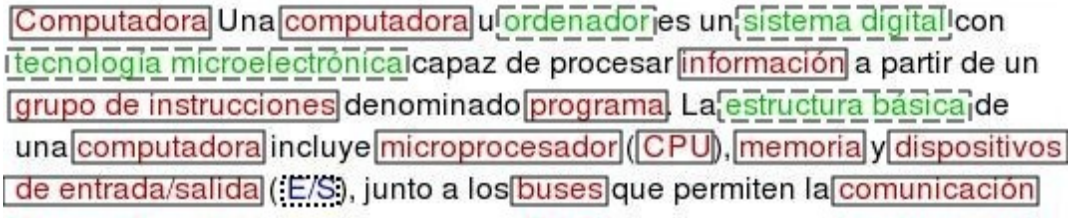
La última opción es generar un archivo XML con los candidatos a término marcados en el mismo texto que fue proporcionado en la entrada. Un ejemplo de la etiqueta que se asigna a cada candidato se muestra a continuación:

```
<term type="1" lemma="computadora" NCvalue="37.147" Cvalue="44.56" freq="46" lenght="1"
POS="NC"> Computadora</term>
```

El parámetro *type* puede valer 1, 2 o 3. Simplemente señala si el candiato se encuentra en el primer, segundo o tercer tercio de la lista ordenada por medio del NC-value. El parámetro *lemma* contiene el lema del candidato. *NCvalue*, *Cvalue* y *freq* contienen estos valores del candidato mientras que *lenght* tiene su longitud en palabras. Finalmente, *POS* almacena el patrón sintáctico del candidato. Por medio de estos parámetros se pueden hacer búsquedas sobre categorías gramaticales, longitud de los términos, etc.

Además, se ha creado una hoja de estilo para desplegar este archivo XML en un navegador de internet. Los candidatos a término en esta hoja pueden ser de color rojo, verde o azul dependiendo del valor que tome el parámetro type. Si el tipo de término es 1 (se encuentra en el primer tercio de la lista), es más probable que sea un verdadero término, por lo que se despliega en color rojo. Si su valor es 2 (segundo tercio), se despliega en color verde. Finalmente, si se encuentra en el último tercio de la lista, con valor 3, el color que toma es el azul.

A continuación se ofrece una muestra del archivo XML generado para cada uno de los archivos del corpus de entrada. Debido a que este documento está en blanco y negro, los candidatos se identifican encerrados: línea sólida para los del grupo 1, línea de guiones para los del grupo 2 y línea punteada para los del grupo 3.



Apéndice 4. Lista completa de los verdaderos términos extraídos

Donde se ofrece una lista con los verdaderos términos hallados en la lista de candidatos a término proporcionada por el extractor en el experimento A realizado en una muestra del Corpus de Informática en Español (apartado 4.1).

disco duro
memoria
computadora
sistema operativo
memoria flash
dato
dispositivo
instrucción
sistema
procesador
unidad de disco rígido
ordenador
programa
partición
USB
información
microprocesador
unidad de control
servidor
capacidad de almacenamiento
sistema de archivos
usuario
llavero USB
computadora personal
CPU
memoria principal
tabla de particiones
ancho de banda
fuente de alimentación
tarjeta de memoria
unidad de disco duro
registro de arranque maestro
reproductor portátil de MP3
dispositivo de almacenamiento
aplicación
memoria RAM
placa base
computadora portátil
puerto de red
dispositivo E/S
mouse óptico
sistema de ficheros
RAM
sector de arranque
particionamiento de disco
puerto
lenguaje de máquina
SCSI
cargador de arranque
software
disponibilidad
función
cámara digital
byte
bus
computadora notebook
unidad central de proceso
puerto hardware
dispositivo hardware
DRAM
dispositivo de celda multi-nivel
computadora de uso general
unidad lógica y aritmética
partición de disco duro
puerto USB
Memoria de acceso Aleatorio
corriente eléctrica
sistema de archivos FAT
tecnología DVD+RW
teclado
máquina
almacenamiento
interfaz
voltaje
dirección de memoria
Firewire
velocidad de transferencia
canal de fibra
reproductor de CD-ROM
lenguaje de programación
grabador-lector de DVDs
compatibilidad
computadora de escritorio
arquitectura von Neumann
Mbit/s
ordenador de sobremesa
programador
computador
CDs
UPS
ALU
DVDs
proceso
PDA
PCs
unidad de disco
puerto máquina
impresor
unidad aritmético-lógica
memoria EEPROM
controlador SCSI
PC Card
computadora digital
acceso secuencial
puerto paralelo
computadora central
servidor SMP
módem
terminal sumidero
máquina de Turing
ordenador personal
sistema NAND
MB
computación
GB
PC
chip
circuito
programación
IDE
EEPROM
KB
memoria de sobretodo lectura
tarjeta de memoria flash
carga de trabajo high-end
dispositivos de almacenamiento desinstalables
reproductor de sonido portátil
reproductor de MP3 portátil
memoria de lectura escritura
lectura de acceso aleatorio
standard de bus serie
PCs portátiles
unidad de disco interno
Bus de Serie Universal
unidad de disco flexible
informática de alto rendimiento
memoria de acceso secuencial
dispositivo de almacenamiento masivo
máquina de Turing universal
cargador de arranque LILO
lenguaje de alto nivel
máquina diferencial de Babbage
lenguaje de bajo nivel
dirección de volumen lógico
reproductor de DVDs portátil
máquina tabuladora de Hollerith
tarjeta de memoria extraíble
red
acceso
registro de particiones
almacenamiento de datos
procesamiento de información
instrucción de salto
corrupción de datos
ancho de bus
secuencia de arranque
reproductor de audio
memoria de dato
unidad de información
densidad de almacenamiento

unidad de ejecución
transmision de datos
dispositivo de E/S
memoria de instrucción
memoria Tag RAM
navegador de internet
dispositivo de entrada/salida
servicio de correo
unidad de DVD
interfaz de dispositivo
tarjeta de expansión
escala de integración
dispositivo de mano
reproductor de DVD-ROM
transistor
registro de almacenamiento
componente de red
barra de tarea
concentrador
grabadora de CD
unidad de DVD-ROM
rutina de computadora
interfaz de UBS
sistema de wireless
procesador de texto
modo de direccionamientos
procesamiento de datos
sistema de particionamiento
dirección de red
lector de CDROM
reproductor de películas
RAID de paridad
banco de memoria
slot de sistema
torre de servidores
interfaz de red
transferencia de dato
base de datos
path de datos
puerto de E/S
controlador de dispositivo
ciclo de CPU
escalabilidad
puerto bus ISA
juego de instrucciones
hoja de cálculo
paths de datos
asistente personal digital
descriptor de partición
formato
informática de consumo
conexión de red
operación de comparación
lector de CD-ROM
arquitectura de computadoras
navegación
almacenamiento high-end
tecla
performance
transistor
SRAM
memoria volátil
CD
conector
conexión
reproductor MP3
conexion USB
monitor
soporte CD-R
capacidad plug-and-play
sistema digital
memoria VRAM
standard USB
controlador gráfico
transistor NMOS
dispositivo SCSI
lector USB
EPROM
sistema informático
Async SRAM
Sync SRAM
capacidad crítica
soporte DVD+RW
USB-On-The-Go
transistor FAMOS
Fetch
video consola
formato lógico
tarjeta gráfica
sóquet
minicomputadora
sistema auxiliar
E/S mapeada
escáner
lector óptico
puerto serie
dato crítico
cabeza lectograbadora
disco SATA
enlace USB
switch
información visual
acceso directo
disco DVD
código abierto
red inalámbrica
protocolo XML
microprocesador gráfico
sistema USB
unidad CD-ROM
pin
censor óptico
mouse
sistema NOR
megabyte
entorno informático
circuito impreso
capa mecánico
memoria temporal
conector USB
sistema microordenador
memoria NAND
software HTML
tecla programable
partición primaria
teclado AT
memoria secundaria
placa madre
localidad temporal
trabajo computacional
conductor
computadora ultraportátil
carga computacional
cable USB
ATA
UDP
número IP
impulso magnético
equipo informático
conexión USB
control computacional
video DVD
dispositivo digital
relé electromecánico
TCP
pantalla
host externo
código máquina
memoria DRAM
módulo SIMM
CD audio
particionamiento imaginario
terminal fuente
módulo RIMM
ratón PS
reconfiguración dinámica
procesador integrado
operación aritmético
disquete
standard SCSI
software digital
módulo DIMM
información digital
dispositivo portátil
unidad DVD
fotografía digital
Pipelined SRAM
E/S

SATA
Rom
BIOS
DVD
semiconductor
chipset
ratón
YAFFS
MPEG
recolector de basura
grabación
tarea
XML
tasa de transferencia
medio de arranque
cargador de arranque
almacenamiento de información
reproductor de sonido
bus de datos
memoria de lectura
esquema de particionamiento
protocolo
Descodificación
caché
microcontrolador
microchip
IDE/ATA
multitarea
Mbytes/s
supercomputadora
poliactividad
disquetera
Memoria
configuración
impresora
foto digital
kilobyte
ATA/IDE
UCP
MB/s
ensamblador
disquete
LCD
PCI
kilobit
tarjeta madre
UPSs
Prolog
LiveCDs
Tag RAM
videojuego
reprogramación
cursor
watts
MBs
hercios
disquete-
dispositivo electrónico
PROM
MBR
socket
circuito electrónico
macrocomputadoras
joysticks
GRUB
XOSL
web
navegador
bits
correo electrónico
Zip
CDVivo
especificación USB
operación lógica
Mhz
Gigaherzios
Ghz
Particionamiento
notebook
JFFS
arranque
RIMM
DIMM
botón
hardware
kernel
mininotebook
ensamblador
wireless
refresco
Internet
MB/seg
RAID
control
informática
rutina
ejecución
archivo
procesamiento
funcionalidad
FAT
flash
disco rígido