Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Departamento de Computación

Nuevas fuentes de información para entrenamiento de etiquetadores gramaticales

Tesis presentada para optar al título de
Licenciado en Ciencias de la Computación

Fernando Jorge Rodriguez

Director: Dr. José Castaño
Buenos Aires, 2013

Índice general

Capítulo 1

Introducción

El etiquetado gramatical, también conocido como Part-of-speech tagging, POS tagging o simplemente POST, es el proceso de asignar una etiqueta a cada una de las palabras de un texto según su categoría léxica. Por ejemplo tomemos la oración siguiente:

There is no asbestos in our products now.

El resultado de etiquetarla gramaticalmente es:

There/EX is/VBZ no/DT asbestos/NN in/IN our/PRP products/NNS now/RB ./.

donde cada palabra está sucedida por una barra oblicua seguida de la etiqueta gramatical asignada.

Se puede apreciar por ejemplo que la palabra *is* fué etiquetada como VBZ (verbo de tiempo presente en tercera persona singular), *products* fué etiquetada como NNS (sustantivo plural), etc. Es decir que a cada palabra se le asignó un código que se corresponde con una función gramatical.

El etiquetado gramatical brinda una gran cantidad de información sobre una palabra y sus vecinas. Una etiqueta gramatical puede ofrecer información relacionada con la pronunciación: en inglés la palabra *content* puede ser un sustantivo o un adjetivo y su pronunciación varía dependiendo de este hecho. Utilizando estas ideas se pueden producir pronunciaciones más naturales en un sistema de síntesis del habla (texto a voz) y también se puede obtener más exactitud en un sistema de reconocimiento del habla (voz a texto).

Otra aplicación importante del etiquetado gramatical en sistemas de recuperación de la información es el reconocimiento de sustantivos u otro tipo de palabras importantes dentro de un documento, para guardar y utilizar esta información en búsquedas posteriores.

La complejidad del etiquetado gramatical reside en la ambigüedad gramatical inherente al lenguaje. Por ejemplo, la palabra *premio* puede funcionar como sustantivo o como verbo:

1) *Gané un premio*
2) *Por tu esfuerzo te premio*

En 1), *premio* tendría que recibir la etiqueta gramatical NN (sustantivo común) mientras que en 2) tendría que recibir la etiqueta gramatical VB (verbo). El sentido gramatical de una palabra se obtiene en base a la definición de la misma y el contexto en que ésta aparece (las palabras y signos de puntuación circundantes).

El etiquetado gramatical es realizado manualmente por lingüistas o automáticamente por programas conocidos como etiquetadores gramaticales. La mayoría de las implementaciones actuales de estos programas están basadas en el aprendizaje; toman un corpus [1] anotado correctamente con el cual se entrenan y luego emplean el conocimiento adquirido para etiquetar el corpus objetivo.

En esa primer etapa conocida como entrenamiento, el etiquetador gramatical obtiene y preserva información sobre cada palabra, su etiqueta asignada y su contexto. Posteriormente dado un corpus objetivo como entrada el etiquetador determina una etiqueta para cada palabra.

Los resultados del etiquetado dependen en gran medida de la calidad, cantidad y representatividad sobre el dominio abordado de los datos de entrenamiento. Uno de los grandes problemas de este proceso reside en la falta de corpus anotados para utilizar como datos de entrenamiento.

Un corpus anotado es un conjunto de palabras con su correspondiente etiqueta gramatical, como se muestra a continuación:

Areas /NNS of/IN the/DT factory/NN were/VBD particularly/RB dusty/JJ where/WRB the/DT crocidolite/NN was/VBD used/VBN . /.

La idea de este trabajo consiste en generar un nuevo conjunto de datos que se utilizará como corpus de entrenamiento empleando la información gramatical contenida en los ejemplos del diccionario *Cobuild*. Este diccionario fué elegido ya que utiliza ejemplos reales y posee información gramatical sobre el uso de la palabra definida para cada ejemplo.

Cobuild presenta su información en un archivo de texto difícilmente legible y carente de formato conocido. En la primer etapa de este trabajo fué necesario identificar y comprender las entradas del diccionario.

Una vez realizada esta tarea la próxima etapa consistió en extraer los ejemplos junto con su información gramatical. Se realizó una conversión de etiquetas *Cobuild* en etiquetas *Penn Treebank*[2] ya que las primeras no son standard y no poseen documentación, lo cual dificulta ampliamente la tarea de análisis y medición de resultados. Se procesaron y reprocesaron los archivos cuidadosamente para perder la menor cantidad de información gramatical posible.

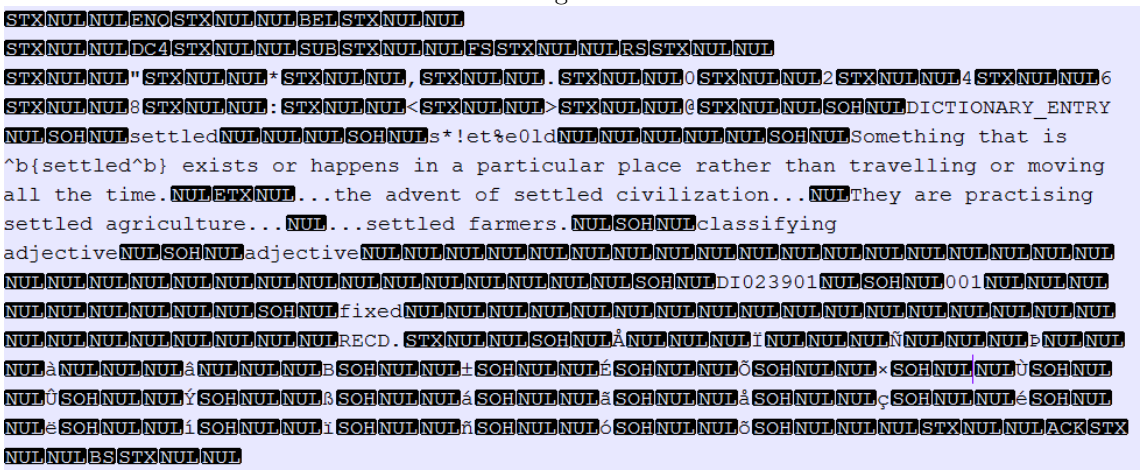
Una vez realizado esto, se generaron etiquetas gramaticales para las palabras de los ejemplos que no las poseían utilizando un etiquetador automático.

Por último se unieron los ejemplos y sus etiquetas dando lugar al nuevo conjunto de datos mencionado. Una vez obtenida esta nueva fuente de información, fué utilizada como corpus de entrenamiento para distintos etiquetadores sobre distintos corpora, analizando y midiendo los resultados obtenidos.

[1] Colección de textos escritos y/o transcripciones del lenguaje oral para cierto idioma
[2] Penn Treebank es un conjunto de etiquetas standard bien documentado

A modo de ejemplo se presenta un extracto de cada una de las etapas:

Archivo original Cobuild:



Se puede observar que el archivo original de *Cobuild* es de dificultosa lectura y no hay documentación conocida para su formato. A continuación se presenta la entrada extraída para *settled*, correspondiente al extracto del archivo anterior:

`DICTIONARY_ENTRY`
`settled` ⟶ *palabra*
`s*!et%e0ld` ⟶ *pronunciación*
`Something that is settled exists or happens in a particular place rather than travelling or moving all the time.` ⟶ *definición*
`...the advent of settled civilization... They are practising settled agriculture... ...settled farmers.` ⟶ *ejemplos*
`classifying adjective` ⟶ *etiqueta específica*
`adjective` ⟶ *etiqueta general*

A partir de esta entrada se extraen y se unen los ejemplos de *Cobuild* junto con su información gramatical, la cual es traducida en etiquetas *Penn Treebank*. El resultado de este proceso puede verse en *1)*. Luego se corre un etiquetador automático sobre los ejemplos para obtener las etiquetas que no están presentes en *1)*, obteniendo así el fragmento *2)*. Por último se unen *1)* y *2)* preservando las etiquetas extraídas de *Cobuild*.

1) Ejemplos extraídos de Cobuild	
the	
advent	
of	
settled	***JJ***
civilization	
They	
are	
practising	
settled	***JJ***
agriculture	
settled	***JJ***
farmers	
.	

⟶

2) Etiquetado automático	
the	*DT*
advent	*NN*
of	*IN*
settled	***VBN***
civilization	*NN*
They	*PRP*
are	*VBP*
practising	*VBG*
settled	***VBN***
agriculture	*NN*
settled	***VBN***
farmers	*NNS*
.	.

⟶

3) Nueva fuente de información generada	
the	*DT*
advent	*NN*
of	*IN*
settled	***JJ***
civilization	*NN*
They	*PRP*
are	*VBP*
practising	*VBG*
settled	***JJ***
agriculture	*NN*
settled	***JJ***
farmers	*NNS*
.	.

Una vez realizado este proceso se utiliza la nueva fuente de información generada para entrenar etiquetadores atomáticos y analizar sus resultados sobre distintos corpora.

Capítulo 2

Definiciones y marco teórico

A continuación se presentan definiciones y teorías sobre las que se basa el trabajo realizado. Se presenta el concepto de etiqueta gramatical, es decir, un código que identifica el rol que cumple una palabra dentro de cierto contexto. Se muestran los conjuntos de etiquetas que han sido utilizados intentando abarcar los distintos significados que pueden tener las palabras. Se explica el concepto de etiquetado gramatical, es decir, la tarea de asignar a cada palabra una etiqueta gramatical adecuada según el contexto en donde ésta aparece. Se muestran ejemplos de que esta tarea está muy lejos de ser trivial, introduciendo el concepto de ambigüedad gramatical.

Se exhibe la importancia del etiquetado gramatical dentro de distintas áreas como la computación lingüistica, reconocimiento y síntesis del habla. Se muestra como se maneja este proceso utilizando programas que lo realizan automáticamente; los etiquetadores gramaticales automáticos. Se decriben implementaciones actuales que utilizan información estadística que el etiquetador emplea para reproducir el etiquetado.

Se presenta el concepto de corpus y corpus anotado gramaticalmente como conjuntos de información extremadamente valiosos para todas estas tareas. Se muestra la forma de medir, evaluar y comparar el rendimiento de los etiquetadores gramaticales, introduciendo los conceptos de corpus de entrenamiento y corpus de verificación. Se muestran técnicas de análisis de error para la etiquetación automática. Se exhibe también el manejo de ciertos casos especiales dentro del proceso de etiquetación automático; las palabras desconocidas. Y por último se explican en detalle los etiquetadores automáticos utilizados en el presente trabajo.

2.1. Etiquetas

Tradicionalmente la deficinición de POS o etiqueta gramatical se ha basadado en funciones sintácticas y morfológicas, es decir que se agrupan en clases las palabras que funcionan similarmente con respecto a lo que puede ocurrir a su alrededor (sus propiedades de distribución sintáctica) o con respecto a los afijos que poseen (sus propiedades morfológicas). Mientras que las clases de palabras tienen tendencia hacia la coherencia semántica (por ejemplo los sustantivos generalmente describen gente, lugares o cosas y los adjetivos generalmente

describen propiedades), este no es necesariamente el caso y en general no se utiliza coherencia semántica como criterio para la definición de POS o etiqueta gramatical.

Las etiquetas gramaticales pueden ser divididas en dos grandes categorías: clases cerradas y clases abiertas. Las clases cerradas son aquellas que tienen miembros relativamente fijos. Por ejemplo, las preposiciones son una clase cerrada porque hay un conjunto cerrado de ellas, es decir que son un grupo de palabras que raramente varía ya que raramente se agregan nuevas preposiciones. En contraste, los sustantivos y los verbos son clases abiertas ya que continuamente se introducen y eliminan nuevos verbos y sustantivos al lenguaje. Es probable que cualquier hablante o corpus tenga una clase abierta de palabras diferente, pero todos los hablantes de un lenguaje y corpora suficientemente grandes, seguramente van a compartir el conjunto de clases de palabras cerradas. Las clases de palabras cerradas también son generalmente palabras funcionales como *de*, *y* o *tu*, que tienden a ser muy cortas, ocurrir frecuentemente y generalmente tienen usos estructurales en gramática.

Hay cuatro clases abiertas principales:

- **Sustantivos** Es el nombre dado a la clase sintáctica que denota personas, lugares o cosas. Pero desde que las clases sintácticas como sustantivos son definidas sintáctica y morfológicamente en lugar de semánticamente, algunas palabras para personas, lugares y cosas pueden no ser sustantivos y a la inversa, algunos sustantivos pueden no ser palabras para personas, lugares o cosas. Por lo tanto los sustantivos incluyen términos concretos como *barco* y *silla*, abstracciones como *banda ancha* y *relación*. Se puede definir a una palabra como sustantivo basándose en características como la capacidad de ocurrir con determinantes (una *cabra*, su *banda ancha*), tomar posesivos (los ingresos anuales de *IBM*) y para la mayoría pero no todos los sustantivos, ocurrir en la forma plural (*cabras*, *teléfonos*). Los sustantivos tradicionalmente son agrupados en sustantivos propios y sustantivos comunes.
  - **Sustantivos propios:** Son nombres de personas específicas o entidades y usualmente son escritos en mayúscula.
  - **Sustantivos comunes:** En algunos lenguajes se dividen en sustantivos contables e incontables.
    - **Sustantivos contables:** Son aquellos que permiten establecer su número en unidades. En general esta clase posee forma singular y plural (silla/s, dedo/s).
    - **Sustantivos incontables:** Se refieren a sustantivos para los cuales no se puede determinar su número en unidades (harina, nieve, azúcar).
- **Verbos:** Los verbos son una clase de palabras que incluye a la mayoría de las palabras referidas a acciones y procesos. Tienen ciertas formas morfológicas como tiempo, modo, persona, regularidad, etc. Además, el verbo puede concordar en género, persona y número con algunos de sus argumentos o complementos (a los que normalmente se conoce como sujeto, objeto, étc.). En español concuerda con el sujeto siempre en número y casi siempre en persona (la excepción es el caso del llamado sujeto inclusivo: *Los españoles somos así*).

Algunos ejemplos:

Marisol *canta* una ópera.
La comida *está* caliente.

- **Adjetivos:** Las palabras pertenecientes a esta clase expresan propiedades o cualidades. Por ejemplo *Ese hombre es* ***alto***. Los adjetivos tienen género y número al igual que los sustantivos. El género y el número de los adjetivos depende del sustantivo al que acompañan. Hay adjetivos que presentan una sola forma para el masculino y para el femenino. Son adjetivos de una sola terminación (verde, especial, amable, grande, etc.). Por el otro lado, los adjetivos de dos terminaciones presentan distintas formas para el masculino y el femenino (feo-fea, pequeño-pequeña, blanco-blanca, etc.) Se clasifican en:
  - **Determinativos:** Preceden al sustantivo, lo concretan y lo presentan
    - **Demostrativos:** *Esta* niña
    - **Posesivos:** *Mi* niña
    - **Numerales:** *Tres* niñas
    - **Indefinidos:** *Algunas* niñas
    - **Exclamativos:** ¡*Qué* niña!
    - **Interrogativos:** ¿*Qué* niña?
  - **Calificativos:** Califican al sustantivo, es decir, añaden cualidades al sustantivo. Los adejtivos calificativos se dividen en especificativos y explicativos o epítetos.
    - **Adjetivos calificativos especificativos:** Son aquellos que concretan el significado del sustantivo. Suelen aparecer detrás del sustantivo.
      Ej: Quiero una corbata *azul*.
    - **Adjetivos calificativos explicativos o epítetos:** Indican cualidades que ya de por sí lleva el sustantivo. Suelen ir delante del sustantivo.
      Ej: *Blanca* nieve, *Verde* hierba.

- **Adverbios:** Los adverbios son otro ejemplo de clase abierta de palabras: se definen como modificadores del verbo, adjetivo o de otro adverbio. Tradicionalmente se dividen en:
  - **Adverbios de lugar**: aquí, allí, ahí, allá, acá, arriba, abajo, cerca, lejos, delante, detrás, encima, debajo, enfrente, atrás, alrededor, etc.
  - **Adverbios de tiempo absoluto**: pronto, tarde, temprano, todavía, aún, ya, ayer, hoy, mañana, siempre, nunca, jamás, próximamente, prontamente, anoche, enseguida, ahora, mientras.
  - **Adverbios de modo**: bien, mal, regular, despacio, deprisa, así, tal, como, aprisa, adrede, peor, mejor, fielmente, estupendamente, fácilmente - todas las que se formen con las terminaciones "mente".

- **Adverbios de cantidad o grado**: muy, poco, muy poco, mucho, bastante, más, menos, algo, demasiado, casi, sólo, solamente, tan, tanto, todo, nada, aproximadamente.

Por otro lado tenemos las clases cerradas de palabras que detallamos a continuación:

- **Preposiciones:** Las preposiciones son enlaces que relacionan los componentes de una oración para brindarles sentido. La unión se lleva a cabo con una o varias palabras. La significación que dan las preposiciones responde a circunstancias de movimiento, lugar, tiempo, modo, causa, posesión, pertenencia, materia y procedencia.
  Algunos ejemplos:

  Me levanté de la cama ***a*** *las ocho de la mañana.*
  Dejé mis cuadernos ***sobre*** *el sillón.*
  Corrí apresurado ***hacia*** *la calle pero no logré divisarte.*
  Lucía se divierte ***con*** *sus muñecas.*

- **Determinantes:** Los determinantes son clases cerradas de palabras que ocurren con sustantivos, generalmente marcando el principio de una frase sustantiva. Un pequeño subtipo de determinantes es el artículo: *a*, *el*. Otros determinantes incluyen *ese* (como en *el libro ese*).

- **Pronombres:** Los pronombres son formas que generalmente actúan como una clase de atajo para referirse a alguna frase sustantiva, entidad o evento. Se dividen en:

  - **Pronombres personales:** Hacen referencia a personas o entidades (yo, tú, él, ella, nosotros, ellos, etc.)
  - **Pronombres posesivos:** Son formas de pronombres personales que indican una posesión actual o mas generalmente solo una relacion abstracta entre la persona y algun objeto (mío, tuyo, suyo, mi, nuestro, etc.)

- **Conjunciones:** Las conjunciones son utilizadas para unir dos frases, cláusulas o sentencias. Las conjunciones coordinantes como *y*, *o* unen dos elementos de igual estado. Las conjunciones subordinativas son utilizadas cuando uno de los elementos es de algún tipo de estado integrado.

  Ej.: *Me molestó* ***que*** *no me lo dijeras.*

- **Verbos auxiliares:** Los verbos auxiliares son verbos que proporcionan información gramatical y semántica adicional a un verbo de significado completo. Dichos verbos auxiliares brindan la información gramatical de modo, tiempo, persona y número y las formas no personales.

  Ej.: *¿Por qué no* ***has*** *llegado a la hora prevista?*

  o también

  La avenida principal de la ciudad ***fué*** *clausurada por obras de refacción.*

- **Numerales:** Los determinantes numerales o simplemente numerales son los que expresan de modo preciso y exacto la cantidad de objetos designados por el sustantivo al que acompañan, delimitan o designan. Limitan el significado general del sustantivo, precisando con exactitud la cantidad de objetos que aquel designa o el lugar de orden que ocupan. Los numerales pueden ser de varias clases. Los más importantes son:

  - **Cardinales**: Informan una cantidad exacta:
    Quiero *cuatro* libros.
  - **Ordinales**: Informan del orden de colocación:
    Quiero el *cuarto* libro.
  - **Fraccionarios**: Informan de particiones de la unidad:
    Quiero la *cuarta* parte.
  - **Multiplicativos**: Informan de múltiplos:
    Quiero *doble* ración.

2.2. Conjuntos de etiquetas

La sección anterior dió una descripción general de los tipos de clases sintácticas a las que pertenecen las palabras. Esta sección presenta los conjuntos de códigos que se corresponden con cada una de estas clases sintácticas, también llamados *tagsets* o conjuntos de etiquetas.

Todavía no existe un consenso sobre el conjunto de etiquetas o *tagset* más adecuado. Generalmente los conjuntos de etiquetas grandes ofrecen una decripción sintáctica más específica mientras que los conjuntos de etiquetas más pequeños usualmente brindan una información lingüística más acotada. Una de las características clave para decidir que conjunto de etiquetas es el más adecuado justamente depende del nivel de detalle lingüístico que se esté buscando.

También cabe destacar que los conjuntos de etiquetas más pequeños generalmente están contenidos en los conjuntos mayores. Debido a que las etiquetas más específicas que se encuentran en los conjuntos mayores pueden ser convertidas en etiquetas de menor especificidad con la consecuente pérdida de detalle lingüístico.

Por el otro lado, también se pueden convertir las etiquetas pertenecientes a conjuntos pequeños en etiquetas de mayor especificidad que pertenecen a conjuntos más grandes, ya que generalmente existen etiquetas equivalentes en estos últimos.

A continuación se muestra como ejemplo el conjunto de etiquetas *Penn Treebank*:

Cuadro 2.1: *Conjunto de Etiquetas Penn Tree Bank*

Etiqueta	**Descripción**	**Ejemplo**
CC	Coordinating conjunction	*and*
CD	Cardinal number	*1, third*
DT	Determiner	*the*
EX	Existential	*there there is*
FW	Foreign word	*d'hoevre*

Cuadro 2.1: *Conjunto de Etiquetas Penn Tree Bank*

Etiqueta	**Descripción**	**Ejemplo**
IN	Preposition/subordinating conjunction	*in, of, like*
JJ	Adjective	*green*
JJR	Adjective, comparative	*greener*
JJS	Adjective, superlative	*greenest*
LS	List marker	*1)*
MD	Modal	*could, will*
NN	Noun, singular or mass	*table*
NNS	Noun plural	*tables*
NNP	Proper noun, singular	*John*
NNPS	Proper noun, plural	*Vikings*
PDT	Predeterminer both	*the boys*
POS	Possessive ending	*friend's*
PRP	Personal pronoun	*I, he, it*
PRP$	Possessive pronoun	*my, his*
RB	Adverb	*however, usually, naturally, here, good*
RBR	Adverb, comparative	*better*
RBS	Adverb, superlative	*best*
RP	Particle	*give up*
SYM	Symbol	*+, %,&*
TO	To	*to go, to him*
UH	Interjection	*uhhuhhuhh*
VB	Verb, base form	*take*
VBD	Verb, past tense	*took*
VBG	Verb, gerund/present participle	*taking*
VBN	Verb, past participle	*taken*
VBP	Verb, sing. present, non-3d	*take*
VBZ	Verb, 3rd person sing. present	*takes*
WDT	Wh-determiner	*which*
WP	Wh-pronoun	*who, what*
WP$	Possessive wh-pronoun	*whose*
WRB	Wh-abverb	*where, when*
$	Dollar sign	*$*
#	Pound sign	*#*
”	Left quote	(’ or ”)
”	Right quote	(’ or ”)
(	Left parenthesis	([, (, {, ¡)
)	Right parenthesis	( ],), }, ¿)
,	Comma	,
.	Sentence-final punc	(. ! ?)
:	Mid-sentence punc	(: ; ... -)

Aunque no exista un consenso sobre que conjunto de etiquetas utilizar, hay un pequeño número de conjuntos de etiquetas o *tagsets* populares para el idioma inglés, muchos de los cuales evolucionaron a partir del conjunto de etiquetas utilizado para etiquetar el corpus *Brown*. Este conjunto de etiquetas se conoció co-

mo el *Brown Corpus Tag-set*, un conjunto de 87 etiquetas que se utilizó para etiquetar el corpus *Brown*.

Junto al *Brown Corpus Tag-set* se encuentran dos de los conjuntos de etiquetas más utilizados: el conjunto de etiquetas reducido *Pen Treebank* de 45 etiquetas y el conjunto de etiquetas *CLAWS C5* de tamaño medio (62 etiquetas) que fué utilizado para etiquetar el *British National Corpus* (BNC).

El conjunto de etiquetas *Penn Treebank* mostrado en la tabla anterior fué utilizado para etiquetar el corpus *Brown*, el corpus *Wall Street Journal* y el corpus *Switchboard* entre otros. En realidad, quizás en parte por su pequeño tamaño es uno de los conjuntos de etiquetas más utilizado.

A continuación se exhiben algunos ejemplos de oraciones del corpus *Brown* etiquetadas con el conjunto de etiquetas *Penn Treebank*. Se representará una palabra etiquetada mediante la colocación de una barra oblicua seguida de su etiqueta:

1. The/DT grand/JJ jury/NN commented/VBD on/IN a/DT number/NN of/IN other/JJ topics/NNS ./.
2. **There/EX** are/VBP 70/CD children/NNS **there/RB**
3. Although/IN preliminary/JJ findings/NNS were/VBD **reported/VBN** more/RBR than/IN a/DT year/NN aho/IN ,/, the/DT latest/JJS results/NNS appear/VBP in/IN today/NN **'s/POS** New/NNP England/NNP Journal/NNP of/IN Medicine/NNP ./.

El primer ejemplo exhibe los determinantes *the* y *a*, los adjetivos *grand* y *other*, los sustantivos comunes *jury*, *number* y *topics* y el verbo en tiempo pasado *commented*.

El segundo ejemplo muestra el uso de la etiqueta ET para marcar la construcción existencial *there* y otro uso de *there* que es etiquetado como un adverbio (RB).

El tercer ejemplo muestra la segmentación del morfema posesivo *'s* y un ejemplo de la construcción pasiva 'were reported', en la cual el verbo *reported* está marcado como un pasado participio (VBN) en lugar de pasado simple (VBD). También es interesante notar que el sustantivo propio *New England* está etiquetado como NNP. Finalmente, se puede observar que como *New England Journal of Medicine* es un sustantivo propio, el etiquetado de *Treebank* elige marcar cada sustantivo separado como NNP, incluyendo *journal* y *medicine*, que en otro caso hubieran sido etiquetados como sustantivos comunes (NN).

2.2.1. Especificidad de etiquetas: Treebank vs C5 y Brown

El conjunto de etiquetas *Penn Treebank* es una selección de 45 etiquetas del conjunto de etiquetas *Brown* (de 87 etiquetas). Este conjunto reducido excluye cierta información lingüística. Por ejemplo los conjuntos de etiquetas *Brown* y *C5* incluyen una etiqueta para cada una de las diferentes formas de los verbos *do*, *be* y *have*(C5 propone la etiqueta VDD para *did* y VDG para *doing*). Estas etiquetas fueron omitidas en el conjunto *Penn Treebank*.

Ciertas distinciones sintácticas no fueron marcadas en el conjunto de etiquetas *Penn Treebank*. Por ejempo, la etiqueta IN es utilizada para preposiciones para conjunciones subordinadas. El conjunto del *Penn Treebank* no es suficientemente específico en ciertos casos. Los conjuntos de etiquetas de *Brown* y *C5*,

por ejemplo, distinguen preposiciones (IN) de conjunciones subordinadas (CS), como en los siguiente ejemplos:

1. **after/CS** spending/VBG a/AT few/AP days/NNS at/IN the/AT Brown/NP Palace/NN Hotel/NN
2. **after/IN** a/AT wedding/NN trip/NN to/IN Corpus/NP Christi/NP ./.

También tienen dos etiquetas para la palabra *to*; en *Brown* el uso del infinitivo es etiquetado como TO, mientras que las preposiciones son etiquetadas como IN:

1. **to/TO** give/VB priority/NN **to/IN** teacher/NN pay/NN raises/NNS

El conjunto de etiquetas *Brown* también posee la etiqueta NR para sustantivos adverbiales como *home*, *west*, *Monday* y *tomorrow*. Como *Penn Treebank* carece de esta etiqueta; *Monday*, *Tuesday* y otros días de la semana son marcados como NNP, *tomorrow*, *west* y *home* son marcados algunas veces como NN y algunas veces como RB. Esto hace al conjunto de etiquetas *Penn Treebank* menos útil para tareas de alto nivel lingüístico como la detección del tiempo de frases.

Sin embargo, el conjunto de etiquetas *Penn Treebank* ha sido el más utilizado para la evaluación de algoritmos de etiquetado automático. Esta es la razón por la cual elegimos este conjunto de etiquetas para utilizar en el desarrollo del presente trabajo.

La información presentada en este subcapítulo está basada en [1, sub chapter 5.2]

2.3. Corpus

Un corpus es una colección de textos escritos y/o transcripciones del lenguaje oral y/o lenguaje oral para cierto idioma que generalmente se utiliza para el estudio del lenguaje. La palabra corpus significa cuerpo en latín, su plural es corpora. Habitualmente el tamaño de un corpus es superior al millón de palabras. Para construir un corpus se reúne una cantidad considerable de textos escritos y/o transcripciones orales y/o lenguaje oral para luego ser preservado en algún formato (generalmente electrónico).

Los corpora son utilizados por lingüistas para describir naturalmente el lenguaje basados en la evidencia obtenida de sus observaciones. En su trabajo generalmente utilizan operaciones estadísticas sobre los corpora para medir la frecuencia de algún aspecto léxico. Los corpora, grandes cantidades de ocurrencia natural del lenguaje, han ayudado a realizar progresos en diferentes campos del lenguaje como el estudio de fraseología, análisis crítico del discurso, estilismos, lingüística forense, traducciones y enseñanza del lenguaje entre otros.

Diferentes tipos de corpora permiten el análisis de distintas clases de discuros para hallar evidencia cuantitativa sobre la existencia de patrones en el lenguaje o para verificar teorías. Los primeros estudios sobre un corpus se enfocaron en palabras; su frecuencia y ocurencia. Con el desarrollo de la tecnología y de motores de búsqueda más precisos y eficientes, las posibilidades crecieron ampliamente.[17]

Cuando corpora escritos y hablados se hicieron disponibles, los lingüistas comenzaron a analizarlos para verificar patrones o diferencias entre el lenguaje hablado y el lenguaje escrito. Parece que aparte de algunas características

obvias como salidas en falso y vacilaciones que se producen en el habla, la utilización de un gran número de expresiones deícticas es más frecuente en los discursos orales. Probablemente esto es debido a los signos lingüísticos extra en los que el lenguaje hablado es más vago. Adicionalmente ciertas características gramaticales manifestadas en el habla deben ser consideradas agramaticales en la escritura.

Otra área importante de estudio lingüístico de corpora es el cambio histórico de los significados de las palabras y la gramática. Y aunque la cantidad de textos viejos disponibles en formato electrónico es mucho más pequeña que la cantidad de textos contemporáneos se han establecido las diferencias.

Por otro lado, en las traducciones es habitual utilizar corpora paralelos que permiten una mejor elección de equivalencias y estructuras gramaticales que podrían reflejar el significado deseado. Estudios adicionales sobre corpora revelaron que los traductores no traducen palabra por palabra sino unidades más grandes (cláusulas o sentencias).

Los estudios de corpora probablemente han tenido una gran influencia en la enseñanza del lenguaje. Primero que nada, han influído en la forma en que se hacen los diccionarios. Segundo los aprendices del lenguaje han sido estudiados para mejorar el conocimiento de los maestros.

Los lingüistas creen que un análisis confiable del lenguaje ocurre mejor en ejemplos recolectados de campo; contextos naturales y con interferencia experimental mínima. Dentro del corpus lingüístico existen visiones divergentes en torno al nivel de las anotaciones. Algunos abogando anotaciones mínimas[18] y permitiendo a los textos «hablar por ellos mismos» mientras que otros se inclinan a favor de las anotaciones como un camino hacia un riguroso entendimiento lingüístico[19].

2.3.1. Un poco de historia

El punto de inflexión en corpus lingüístico moderno seguramente fué la publicación de *Henry Kucera* y *W. Nelson Francis: Computational Analysis of Present-Day American English* en 1967. Un trabajo basado en el análisis del corpus Brown, una compilación cuidadosamente seleccionada de inglés americano de ese entonces, contabilizando alrededor de 1 millón de palabras. *Kucera* y *Francis* sometieron este corpus a una gran variedad de análisis computacionales desde el cual compilaron un rico y nutrido corpus combinando elementos de lingüística, enseñanza de lenguaje, psicología, estadística y sociología. Una publicación adicional clave fué «*Towards a description of English Usage*» de *Randolph Quirk (1960)[21]* en la que introdujo *Survey of English Usage*.

Poco después el editor *Houghton-Mifflin* se acercó a *Kucera* para suministrarle el material base de 1 millón de palabras para su nuevo diccionario *American Heritage Dictionary (AHD)*; el primero en ser compilado utilizando corpus lingüístico. El AHD dió el paso innovador de combinar elementos prescriptivos (como debe utilizarse el lenguaje) con información descriptiva (como se usa actualmente).

Otros editores siguieron el ejemplo. El editor inglés *Collins* creó y compiló el diccionario *Cobuild* utilizando el corpus *Bank of English*. Fué diseñado para usuarios que están aprendiendo inglés como lengua extranjera.

El corpus *Brown* también dió lugar a un número de corpora similarmente estructurada: el corpus *LOB* (1960, inglés británico), *Kolhapur* (inglés indio),

Wellington (inglés de Nueva Zelanda), *Australian Corpus of English* (inglés australiano) y el *Flob* (1990, inglés británico).

Existen también otros corpora que representan más lenguajes, variedades y modos: *International Corpus of English*, el *British National Corpus* que es una colección de 100 millones de palabras provenientes de textos escritos e inglés hablado creado en los 1990s por un consorcio de editores, universidades (*Oxford* y *Lancaster*) y la *British Library*. Para inglés americano contemporáneo, el trabajo se ha centrado en el *American National Corpus* (más de 400 millones de palabras de inglés americano contemporáneo).

El primer corpus computarizado de lenguaje hablado transcripto fué construído en 1971 por el *Montreal French Project[22]*, conteniendo 1 millón de palabras que inspiró a *Shana Poplack* a crear luego un corpus de Francés hablado mucho más grande[23].

Además de estos corpora de lenguaje vivo se encuentra corpora computarizado que fué construído a partir de colecciones de textos de lenguajes antiguos. Como ejemplo tenemos la base de datos *Andersen-Forbes* de la biblia hebrea, desarrollada desde los años 1970[24, 25]. El *Quaric Arabic Corpus*, que es un corpus anotado para el lenguaje árabe clásico del corán. Este proyecto cuenta con múltiples capas de anotación incluyendo segmentación morfológica, etiquetado gramatical y análisis sintáctico utilizando dependencia gramatical[26].

2.3.2. Métodos

Los corpora lingüísticos han generado una cantidad de métodos de investigación intentando trazar un camino desde los datos hacia la teoría. *Wallib* y *Nelson* (2001)[27] introdujeron lo que ellos llamaron la perspectiva 3A: anotación, abstracción y análisis.

- **Anotación:** La anotación consiste en la aplicación de un esquema a los textos. Las anotaciones incluyen marcado estructural, etiquetado gramatical, parsing y varias representaciones más.
- **Abstracción:** La abstracción consiste en la traducción (mapeo) de términos del esquema a términos en el modelo teórico. Típicamente incluye búsqueda lingüística directa y también puede incluir aprendizaje por reglas para parsers.
- **Análisis:** El análisis consiste de exploración estadística, manipulación y generalización desde los datos. También podría incluir evaluaciones estadísticas, optimización basada en reglas o métodos de descubrimiento del conocimiento. La mayoría de los corpora de hoy en día están anotados gramaticalmente y aplican algún método para aislar términos que pueden ser interesantes en las palabras circundantes.

2.4. Etiquetadores gramaticales automáticos

Como se mencionó anteriormente, el etiquetado gramatical es el proceso que asigna a una secuencia de palabras una secuencia de etiquetas gramaticales para las mismas. Generalmente las etiquetas gramaticales también son aplicadas a los signos de puntuación, por lo tanto el etiquetado requiere que los signos de

puntuación sean separados de las palabras. Este proceso se realiza previamente o como parte del etiquetado y es conocido como *tokenización*. El proceso de *tokenización* es el encargado de separar puntos, comas, paréntesis y otros caracteres de las palabras así como también desambigüar el fin de oración (por ejemplo un punto o signo de pregunta) de un signo de puntuación (como en una abreviación, por ejemplo *'etc.'*)

La entrada para un algoritmo de etiquetación automática es una cadena de palabras y un conjunto de etiquetas. La salida es la mejor etiqueta encontrada para cada palabra. Consideremos las siguientes oraciones etiquetadas gramaticalmente:

Book/VB that/DT flight/NN ./.
Does/VBZ that/DT flight/NN serve/VB dinner/NN ?/.

Asignar una etiqueta gramatical a una palabra no es una tarea trivial incluso en estos sencillos ejemplos. Por ejemplo, la palabra *book* es ambigüa. Es decir que tiene más de un uso posible y por lo tanto más de una etiqueta gramatical posible. Puede ser un verbo (como en ***book*** *that flight* o *to* ***book*** *the suspect*) o un sustantivo (como en *hand me that* ***book*** o *a* ***book*** *of matches*). Análogamante *that* puede ser un determinante (como en *Does* ***that*** *flight serve dinner*) o un complementador (como en *I thought* ***that*** *your flight was earlier*).

El problema del etiquetado gramatical reside en resolver estas ambigüedades, eligiendo la etiqueta adecuada según el contexto. ¿Pero qué magnitud tiene el problema de la ambigüedad de las palabras? Podemos apreciar que la mayoría de las palabras en inglés no son ambigüas, o lo que es lo mismo, tienen una única etiqueta posible. Sin embargo, muchas de las palabras más comunes del inglés son ambigüas, es decir que las palabras más utilizadas, las que se emplean con mayor frecuencia, pueden tener más de una etiqueta. Por ejemplo *can* puede ser un auxiliar (puede), un sustantivo (lata o contenedor de metal) o un verbo (poner algo en la lata).

Afortunadamente muchas de las palabras ambigüas son fácilmente desambigüables. Esto sucede porque las etiquetas asociadas a una palabra no suelen ocurrir con la misma frecuencia. Por ejemplo *a* puede ser un determinante o la letra *a* (quizás como parte de un acrónimo o una inicial), pero es preciso notar que el sentido de *a* es mucho más frecuente como determinante que como letra. Es decir que es mucho más frecuente encontrar *a* en oraciones como *My father bought* ***a*** *new car* o *There is* ***a*** *hair in my soup* que en oraciones como *Written by* ***A***. *Kamio* o *The letter* ***a*** *is the first letter of the alphabet.*

Existen distintos métodos computacionales para asignar una etiqueta gramatical a una palabra. La mayoría de los algoritmos de etiquetado automático pertenecen a una de dos clases: etiquetadores basados en reglas o etiquetadores estocásticos.

Los etiquetadores basados en reglas generalmente incluyen una gran cantidad de reglas de desambigüación escritas a mano que especifican, por ejemplo, que una palabra ambigüa es un sustantivo antes que un verbo si es seguida por un determinante.

Los etiquetadores estocásticos generalmente resuelven la ambigüedad de etiquetas utilizando un corpus de entrenamiento del cual "aprenden" como etiquetar. Este aprendizaje se realiza extrayendo información sobre la probabilidad de que una palabra dada tenga cierta etiqueta en cierto contexto.

Adicionalmente existe una tercera clase de etiquetadores que es una mezcla de estos dos: etiquetadores basados en la transformación. Como los etiquetadores basados en reglas, están basados en reglas que determinan cuando una palabra ambigüa debe tener cierta etiqueta. Y como los etiquetadores estocásticos tienen un componente de aprendizaje automático; las reglas son inducidas automáticamente a partir de un corpus de entrenamiento previamente etiquetado.

La información presentada en este subcapítulo está basada en [1, sub chapter 5.3]

2.4.1. Etiquetadores gramaticales basados en reglas

Los primeros algoritmos de asignación de etiquetas gramaticales estaban basados en un proceso de dos etapas. En la primer etapa utilizaban un diccionario para asignar a cada palabra una lista de potenciales etiquetas gramaticales. En la segunda etapa utilizaban grandes listas de reglas de desambigüación escritas a mano para reducir la lista de etiquetas hasta llegar a una para cada palabra. De esta manera eliminaban las etiquetas inconsistentes con el contexto.

Las versiones actuales de los etiquetadores gramaticales basados en reglas mantienen los principios originales teniendo en cuenta que los diccionarios y el conjunto de reglas han adquirido un tamaño considerablemente mayor: manejan alrededor de 3800 reglas y un diccionario de etiquetas del órden de las 56.000 entradas para el idioma inglés.

2.4.2. Etiquetadores gramaticales de aprendizaje automático

La inclusión de probabilidades en el proceso de etiquetación gramatical no es una idea nueva. Surge como una consecuencia natural a partir del hecho de que una palabra es empleada con un sentido gramatical mucho más frecuentemente que con otro. Como se mencionó anteriormente, *a* es mucho más frecuentemente utilizada como determinante que como letra.

La inclusión de probabilidades también responde a otro factor importante: la construcción gramatical; cierta etiqueta es precedida frecuentemente por ciertas otra/s. Por ejemplo, como se mencionó anteriormente, los pronombres posesivos generalmente son sucedidos por sustantivos. Es decir que es más probable encontrar oraciones cuyas palabras estén etiquetadas con PRP$ sucedida por NN que PRP$ sucedida por otra etiqueta.

A continuación vamos a presentar 2 tipos de etiquetadores gramaticales: etiquetadores basados en el modelo oculto de *Markov* o simplemente etiquetadores HMM[1] y etiquetadores basados en el modelo de máxima entropía.

Etiquetadores gramaticales basados en HMM

El uso del modelo oculto de Markov para realizar etiquetado gramatical es un caso especial de la inferencia bayesiana, un paradigma que fué conocido a partir del trabajo de Bayes (1763). La inferencia bayesiana o clasificación bayesiana fue aplicada exitosamente a problemas del lenguaje a partir de 1950.

[1] Por las siglas en inglés de Hidden Markov Model

La clasificación bayesiana puede apreciarse como una tarea para la cual contamos con un conjunto de observaciones y el trabajo consiste en determinar a que conjunto de clases pertenece. En lo que respecta al etiquetado gramatical, se puede utilizar este mismo concepto para tratarlo como una tarea de clasificación de secuencia. En ese caso, la observación será una secuencia de palabras (digamos una oración) para la cual el trabajo reside en asignar una secuencia de etiquetas gramaticales. Como ejemplo tomemos la oración que aparece a continuación:

Secretariat is expected to race tomorrow

En este caso la observación es la secuencia de palabras (es decir la oración misma) y nuestro objetivo es asignarle las etiquetas correspondientes. Ya que una palabra puede ser ambigüa y tener más de una etiqueta posible, hay una pregunta clave que debemos hacernos: ¿Cuál es la mejor secuencia de etiquetas que corresponden a esta secuencia de palabras?

La interpretación bayesiana comienza considerando todas las posibles secuencias de clases –en nuestro caso, todas las posibles secuencias de etiquetas gramaticales. El objetivo aquí es elegir la secuencia de etiquetas que es más probable dada la secuencia de observaciones de n palabras w_1^n. En otras palabras, queremos obtener, de todas las secuencias de n etiquetas t_1^n la secuencia de etiquetas tal que $P(t_1^n|w_1^n)$ sea mayor. Se utilizará la notación $\hat{}$ para decir "nuestra estimación de la secuencia de etiquetas correcta".

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(t_1^n|w_1^n) \tag{2.1}$$

La ecuación anterior se lee así: de todas las secuencias de etiquetas de longitud n, queremos la secuencia particular t_1^n que maximiza el lado derecho.

Mientras que esta ecuación nos garantiza obtener la secuencia de etiquetas óptima, todavía no queda del todo claro como utilizarla. Es decir, para una secuencia de etiquetas dada t_1^n y una secuencia de palabras w_1^n, no sabemos cómo computar directamente $P(t_1^n|w_1^n)$. Aquí entra en juego la clasificación Bayesiana, ofreciendo una forma de transformar la ecuación en un conjunto de otras probabilidades más sencillas de computar. Las reglas de Bayes reemplazan la probabilidad condicional $P(x|y)$ por otras tres probabilidades:

$$P(x|y) \;=\; \frac{P(y|x)P(x)}{P(y)} \tag{2.2}$$

Podemos sustituir (2.2) en (2.1) para obtener (2.3):

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} \frac{P(w_1^n|t_1^n)P(t_1^n)}{P(w_1^n)} \tag{2.3}$$

Convenientemente podemos simplificar (3) eliminando el denominador $P(w_1^n)$. Esto sucede ya que estamos eligiendo una de todas las secuencias de etiquetas, computando $\frac{P(w_1^n|t_1^n)P(t_1^n)}{P(w_1^n)}$ en cada una de ellas. Pero $P(w_1^n)$ no cambia en ninguna secuencia de etiquetas, entonces estamos preguntando siempre por la misma observación w_1^n, que tiene la misma probabilidad $P(w_1^n)$. Por lo tanto

podemos quitar el denominador con la garantía de que el máximo sea el mismo:

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(w_1^n|t_1^n)P(t_1^n)$$

En resumen, la secuencia de etiquetas más probable $\hat{t}_1^n$ dada alguna palabra w_1^n puede ser computada tomando el producto de dos probabilidades para cada secuencia de etiquetas y eligiendo la secuencia que lo maximiza.

Desafortunadamente todavía sigue siendo muy difícil computar esta ecuación directamente. Los etiquetadores gramaticales basados en HMM realizan dos suposiciones simplificadoras. La primera es que la probabilidad de aparición de una palabra depende solo de su etiqueta gramatical, es decir que es independiente de las palabras y etiquetas que tiene alrededor. Más técnicamente:

$$P(w_1^n|t_1^n) \approx \prod_{i=1}^{n} P(w_i|t_i)$$

La segunda suposición es que la probabilidad de aparición de una etiqueta gramatical depende solo de la etiqueta previa (sin tener en cuenta las etiquetas anteriores a la etiquetea previa), esto es la suposición de bigrama.

$$P(t_1^n) \approx \prod_{i=1}^{n} P(t_i|t_{i-1})$$

Utilizando estas suposiciones obtenemos esta nueva ecuación, la cual es utilizada por los etiquetadores gramaticales basados en bigramas para estimar la secuencia de etiquetas gramaticales más probable.

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(t_1^n|w_1^n)P(t_1^n) \approx \underset{t_1^n}{\operatorname{argmax}} \prod_{i=1}^{n} P(w_i|t_i)P(t_i|t_{i-1})$$

La ecuación anterior contiene dos clases de probabilidades, probabilidades de transición de etiquetas y probabilidades de palabras. Tomemos un momento para ver que es lo que representan estas probabilidades.

- **Probabilidades de transición de etiquetas**: Las probabilidades de transición de etiquetas, $P(t_i|t_{i-1})$, representan la probabilidad de que ocurra una etiqueta dada la etiqueta previa. Por ejemplo, es muy probable que un determinantes preceda a un adjetivos o a un sustantivo, como *that/DD flight/NN y the/DT yellow/JJ hat/NN*. Por lo tanto esperamos que las probabilidades $P(NN|DT)$ y $P(JJ|DT)$ sean altas.

  Por otro lado, es infrecuente que los adjetivos precedan a los determinantes, entonces la probabilidad $P(DT|JJ)$ será pequeña. Podemos computar el estimador de máxima verosimilitud o MLE[2] de una probabilidad de transición de etiquetas $P(NN|DT)$ etiquetando y contando las etiquetas gramaticales en un corpus. Esto es: de todas las veces que vemos DT, cuántas de esas veces vemos NN después de DT. Lo expresamos más formalmente con el siguiente cociente:

[2]Por sus siglas en inglés Maximum Likelihood Estimated

$$P(t_i|t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_i)}$$

Elijamos un corpus específico para examinar, por ejemplo el corpus Brown.

En el corpus Brown etiquetado con el conjunto de etiquetas Treebank, la etiqueta DT ocurre 116.454 veces. De esas veces, DT es seguido por NN 56.509 veces. Por lo tanto esta probabilidad de transición se calcula como sigue:

$$P(NN|DT) = \frac{C(DT, NN)}{C(DT)} = \frac{56509}{116454} = 0{,}49$$

Claramente la probabilidad de obtener un sustantivo común después de un determinante es .49 y de hecho alta como sospechábamos.

- **Probabilidades de la palabra**: Por otro lado las probabilidades de la palabra, $P(w_i|t_i)$, representan la probabilidad de que dada una etiqueta esta esté asociada con cierta palabra. Por ejemplo si tenemos la etiqueta VBZ (verbo singular de tiempo presente en tercera persona) y quisiéramos adivinar el verbo asociado a esa etiqueta, probablemente elegiríamos el verbo *is*[3], debido a que el verbo *to be* es muy común en inglés.

  Podemos computar $P(is|VBZ)$ de nuevo contando de cuántas veces que vemos VBZ en un corpus cuántas de esas veces VBZ está etiquetando la palabra *is*. Esto es computar el siguiente cociente:

$$P(w_i|t_i) = \frac{C(t_i, w_i)}{C(t_i)}$$

  En el corpus Brown etiquetado con Treebank, la etiqueta VBZ ocurre 21.627 veces y VBZ es la etiquetra para *is* 10.073 veces. Entonces:

$$P(is|VBZ) = \frac{C(VBZ, is)}{C(VBZ)} = \frac{10{,}073}{21{,}627} = 0{,}47$$

Resumiendo, el etiquetado HMM es la tarea de elegir con la mayor probabilidad una secuencia de etiquetas para una secuencia de palabras dada. HMM incluye la suposición de ciertos hechos para simplificar las ecuaciones originales mejorando así la eficiencia de los cómputos.

La información presentada en este subcapítulo está basada en [1, sub chapter 5.5]

Etiquetadores gramaticales de máxima entropía

El principio de máxima entropía observa que la correcta distribución de la probabilidad de etiquetar la palabra w con una etiqueta t, $p(w, t)$, es aquella que maximiza la incertidumbre o entropía sujeta a restricciones que representan la evidencia; los hechos conocidos. Estas restricciones son llamadas características o *features* y se expresan mediante funciones.

[3] *is* es el presente en tercera persona del verbo *to be*

Dicho de otra manera, dada una sucesión de palabras que se quieren etiquetar con un conjunto de etiquetas (por ejemplo NN, VB, JJ), la asignación correcta de etiquetas es aquella que resulte más uniforme, es decir, la que asigne cada etiqueta a un número parecido de palabras. Penalizando además aquellas distribuciones de probabilidades con poca entropía.

Por ejemplo, una distribución de etiquetas poco uniforme sería asignar a todas las palabras la etiqueta NN, por lo que esta distribución de etiquetas se consideraría poco probable bajo un modelo de máxima entropía.

La idea es encontrar la distribución de probabilidades de etiquetas que mejor modele la sucesión de palabras de entrada. Para lograr este objetivo, lo primero es entrenar el modelo. En este caso, al igual que en anteriores ocasiones, el entrenamiento consiste en la observación de palabras y etiquetas asociadas. Para cada par de palabra y etiqueta $[w, t]$ observado, se calcula:

$$p(w,t) = \frac{OcurrenciasDe(w,t)}{OcurrenciasDe(w)}$$

Además de estas estadísticas, también se pueden considerar diferentes características que no tienen referencia a la frecuencia de ocurrencia de las palabras, sino a aspectos dependientes del contexto de la palabra; que la palabra esté o no en mayúscula, o que sea principio de frase, etc. En general se pueden definir ciertos aspectos siempre y cuando se puedan expresar como una función binaria:

$$f(w,t) = \begin{cases} 1 & \text{si se cumple la condición deseada} \\ 0 & \text{en otro caso} \end{cases}$$

A estas funciones, o a los aspectos que representan se las denomina características o *features* y son utilizadas como restricciones en el modelo:

$$p(f) = \sum_{w,t} p(w,t) f(w,t)$$

2.5. Corpora de entrenamiento y corpora de verificación

Los etiquetadores gramaticales que se basan en modelos de aprendizaje poseen un proceso de entrenamiento sobre un corpus etiquetado previamente en el cual se generan las probabilidades que se utilizan para tomar decisiones frente a palabras ambigüas.

Dicho corpus de entrenamiento necesita ser cuidadosamente considerado: si es muy específico al dominio, corpora pertenecientes a ese dominio serán etiquetados con precisión, pero corpora de diferente dominio serán etiquetados con errores. Por otro lado, si el corpus de entrenamiento es muy general las probabilidades no alcanzarán a reflejar el dominio.

Supongamos que estamos intentando etiquetar una oración particular. Si nuestra oración es parte del corpus de entrenamiento, las probabilidades de las etiquetas para esa oración van a ser extraordinariamente precisas y vamos a sobreestimar la precisión de nuestro etiquetador. Se desprende como conclusión que el corpus de entrenamiento no debe ser parcial incluyendo esa oración. Por lo tanto al trabajar con etiquetadores basados en modelos estocásticos, dado un

corpus de datos relevante, es una tarea habitual dividir los datos en un corpus de entrenamiento y un corpus de verificación.

Una vez realizada esta división se entrena el etiquetador con el corpus de entrenamiento, se ejecuta el proceso de etiquetación y luego se comparan los resultados con el corpus de verificación.

En general existen dos métodos para entrenar y verificar un etiquetador gramatical. En el primer método, se divide el corpus disponible en tres partes: un corpus de entrenamiento, un corpus de verificación y un corpus de test de desarrollo[4]. Se entrena el etiquetador con el corpus de entrenamiento. Entonces se utiliza el corpus de test de desarrollo para eventualmente afinar o ajustar algunos parámetros y en general decidir cual es el mejor modelo. Una vez que se elige el supuesto mejor modelo, se corre contra el corpus de verificación para analizar su rendimiento.

En el segundo método de entrenamiento y verificación, se elige aleatoreamente una división de corpus de entrenamiento y verificación para nuestros datos. Se entrena el etiquetador y luego se calcula el error en el corpus de verificación. A continuación se repite con un corpus de entrenamiento y de verificacion diferente seleccionado aleatoreamente. La repetición de este proceso, llamado validación cruzada, generalmente es realizada 10 veces. Luego se promedian esas 10 corridas para obtener un promedio en la proporción del error.

La información presentada en este subcapítulo está basada en [1, sub chapter 5.5]

2.6. Evaluación de etiquetadores gramaticales

Los etiquetadores gramaticales generalmente son evaluados comparando su *accuracy* contra un corpus de verificación[5] etiquetado por humanos. Definimos *accuracy* como el porcentaje de todas las etiquetas en el corpus de verificación donde el etiquetador y el *Gold Standard* concuerdan. Los algoritmos actuales de etiquetado gramatical tienen un *accuracy* del 96 %-97 % para conjuntos de etiquetas simples como el *Penn Treebank*. Estos valores son para palabras y puntuaciones, el valor para palabras solas es menor.

Naturalmente uno tiende a preguntarse qué tan bueno es un 97 %. El rendimiento de un proceso de etiquetado puede ser comparado contra un límite inferior y un límite superior. Una manera de establecer un límite superior es ver que tan bien realizan la tarea los humanos.

Marcus[13], por ejemplo, encontró que los etiquetadores humanos concuerdan en el 96 %-97 % de las etiquetas en el corpus *Brown* etiquetado con etiquetas *Penn Treebank*. Esto sugiere que el *Gold Standard* debe tener un 3 %-4 % de margen de error, y por lo tanto no tiene sentido obtener un *accuracy* del 100 %. *Ratnaparkhi*[14] mostró que en las palabras donde su etiquetador ha tenido problemas de ambigüedad de etiquetación fueron exactamente las mismas en donde los humanos han etiquetado inconsistentemente el corpus de entrenamiento. Dos experimientos realizados por *Voutilainen*[15] encontraron que cuando a los humanos se les permitió discutir etiquetas, alcanzaron un consenso en el 100 % de las etiquetas.

[4] También llamado *devtest*

[5] También llamado *Gold Standard*

Por otro lado el límite inferior sugerido por *Gale*[16] es elegir la etiqueta más probable aplicando el modelo de unigrama para cada palabra ambigüa. La etiqueta más probable para cada palabra puede ser computada desde un corpus etiquetado a mano (que puede ser el mismo que el corpus de entrenamiento para el etiquetador que está siendo evaluado).

La información presentada en este subcapítulo está basada en [1, sub chapter 5.5]

2.7. Análisis de error

Para mejorar el rendimiento de un etiquetador gramatical necesitamos entender donde está funcionando mal. Por eso el análisis de error tiene un papel preponderante. Esta tarea se realiza construyendo una matriz de confusión o tabla de contingencia. Una matriz de confusión es una matriz de n x n donde la celda (x, y) contiene el número de veces que una palabra con correcta etiqueta x fué etiquetada por el modelo como y. Por ejemplo, la siguiente tabla muestra una porción de la matriz de confusión para los experimentos de etiquetado con HMM.

Cuadro 2.2: *Ejemplo de matriz de confusión*

	IN	**JJ**	**NN**	**NNP**	**RB**	**VBD**	**VBN**
IN	-	.2			.7		
JJ	.2	-	3.3	2.1	1.7	.2	2.7
NN		8.7	-				.2
NNP	.2	3.3	4.1	-	.2		
RB	2.2	2.0	.5		-		
VBD		.3	.5			-	4.4
VBN		2.8				2.6	-

Las etiquetas de la fila indican las etiquetas correctas, las etiquetas de las columnas indican las etiquetas asignadas por el etiquetador, y cada celda indica el porcentaje del error de etiquetado general. Por lo tanto 4.4 % del total de errores fueron causados por fallida etiquetación de VBD como VBN. La matriz anterior y el análisis de error relacionado en *Franz*[12], *Kupiec*[11] y *Ratnaparkhi*[14] sugieren que algunos de los mayores problemas que encaran los etiquetadores actuales son:

1. **NN contra NNP contra JJ:** Estas etiquetas son difíciles de distinguir. Es especialmente importante distinguir entre sustantivos propios para extracción de la información y traducción automática.

2. **RP contra RB contra IN:** Todas estas etiquetas pueden aparecer inmediatamente después del verbo.

3. **VBD contra VBN contra JJ:** Distinguir estas etiquetas es importante

para el *parsing* parcial y para etiquetar correctamente los bordes de las frases nominales.

El análisis de error es una parte crucial de cualquier aplicación lingüística computacional. Puede ayudar a encontrar *bugs*, encontrar problemas en los datos de entrenamiento y lo más importante, ayuda en el desarrollo de conocimiento y/o algoritmos para utilizar en la solución de problemas.

La información presentada en este subcapítulo está basada en [1, sub chapter 5.7]

2.8. Palabras desconocidas

Todos los algoritmos de etiquetado gramatical presentados anteriormente requieren un diccionario que liste las posibles etiquetas de cada palabra para que posteriormente el proceso de etiquetado se encargue de identificar la etiqueta correcta. Pero claro, hay un problema: ningún diccionario, derivado o no de un corpus, es capaz de contener todas las palabras. Los sustantivos propios y los acrónimos son creados muy frecuentemente, de hecho ingresan al lenguaje nuevos sustantivos comunes y verbos en una proporción sorprendente. Por lo tanto, para construir un etiquetador completo no podemos utilizar siempre un diccionario para obtener $P(w_i|t_i)$. Necesitamos algún método para adivinar la etiqueta de una palabra desconocida.

El algoritmo más básico para manejar palabras desconocidas es suponer que cada palabra desconocida es ambigua entre todas las posibles etiquetas, con igual probabilidad. Entonces el etiquetador debe confiar únicamente en etiquetas contextuales para sugerir la etiqueta adecuada. Un algoritmo ligeramente más complejo está basado en la idea de que la distribución de probabilidad de las etiquetas sobre las palabras desconocidas es muy similar a la distribución de las etiquetas sobre palabras que ocurren solo una vez en un corpus de entrenamiento, una idea sugerida por *Baayen y Sproat (1996)*[7] y *Dermatas y Kokkinakis (1995)*[8]. Estas palabras que ocurren solo una vez son conocidas como *hapax legomena*.

Por ejemplo, las palabras desconocidas y *hapax legomena* son similares en el hecho de que son más probables de ser sustantivos, seguidas por verbos, pero infrecuentemente suelen ser determinantes. Entonces la probabilidad $P(w_i|t_i)$ para una palabra desconocida es determinada por el promedio de la distribución sobre todos los conjuntos de palabras de una sola ocurrencia en el corpus de entrenamiento. En resúmen, la idea es utilizar "cosas que hemos visto una vez" como un estimador para "cosas que nunca hemos visto".

De todas maneras, la mayoría de los algoritmos para palabras desconocidas hace uso de una fuente de información mucho más poderosa: la morfología de las palabras. Para el inglés, por ejemplo, palabras terminadas en *s* tienden a ser sustantivos plurales (NNS), palabras terminadas en *ed* tienden a ser pasado participio (VBN), palabras terminadas en *able* tienden a ser adjetivos (JJ), y así. Incluso si nunca vimos una palabra, podemos utilizar hechos sobre su forma morfológica para adivinar su etiqueta. Además la información ortográfica puede ser de mucha ayuda. Por ejemplo, palabras que comienzan con letras mayúsculas generalmente son sustantivos propios (NNP). La presencia de un guión es también una característica útil; las palabras con guión tienen más probabilidad de ser adjetivos (JJ).

¿Cómo son combinadas y utilizadas estas características en los etiquetadores gramaticales? Un método es entrenar por separado estimadores de probabilidad para cada característica, asumiendo independencia, y multiplicando las probabilidades. *Weischedel (1993)*[9] construyó un modelo así, basado en cuatro clases específicas. Utilizaron 3 terminaciones infleccionales (*ed*, *s*, *ing*), 32 terminaciones derivacionales (como *ion*, *al*, *ive* y *ly*), 4 valores de mayúscula dependiendo si una palabra es inicio de oración (+/- mayúscula, +/- inicio) y donde una palabra fué guionada. Para cada característica, entrenaron estimadores de máxima verosimilitud de la característica dada una etiqueta desde un corpus de entrenamiento etiquetado. Entonces combinaron las características para estimar la probabilidad de una palabra desconocida asumiendo independencia y multiplicando:

$$P(w_i|t_i) = p(\text{palabra desconocida}|t_i)p(\text{mayúscula}|t_i)p(\text{final/guión}|t_i) \quad (2.4)$$

Otro acercamiento basado en HMM, proveniente del trabajo realizado por *Samuelsson (1993)*[10] y *Brants (2000)*[3], generaliza el uso de morfología en una manera basada en datos. En este acercamiento, en lugar de preseleccionar ciertos sufijos a mano, son consideradas todas las secuencias finales de letras de todas las palabras. Consideran sufijos menores a diez letras, computando para cada sufijo de longitud i la probabilidad de la etiqueta t_i:

$$P(t_i|l_{n-i+1}, \ldots, l_n) \quad (2.5)$$

Estas probabilidades son suavizadas utilizando sucesivamente menores y menores sufijos. Esta información de sufijos se mantiene por separado para palabras en mayúscula y minúscula.

En general, la mayoría de los modelos de palabras desconocidas intentan capturar el hecho de que las palabras desconocidas improbablemente pertenecen a clases cerradas de palabras. *Brants* modela este hecho computando solamente las probabilidades de sufijos desde el corpus de entrenamiento para palabras cuya frecuencia en el corpus de entrenamiento es ≤ 10.

La información presentada en este subcapítulo está basada en [1, sub chapter 5.7.2]

2.9. Etiquetador Gramatical TnT

TnT(Trigrams' n' Tags) es un etiquetador gramatical estocástico basado en HMM. Según Brants este etiquetador tiene un rendimiento mejor o igual a otros etiquetadores de diferentes bases teóricas, incluyendo etiquetadores basados en máxima entropía.

2.9.1. Modelo teórico

TnT utiliza modelos de Markov de segundo órden para la etiquetación gramatical. Técnicamente calcula, dada una secuencia de T palabras $w_1, \ldots, w_T$

$$argmax_{t_1,\ldots,t_T} \left[\prod_{i=1}^{T} P(t_i|t_{i-1}, t_{i-2})P(w_i|t_i) \right] P(t_{T+1}|t_T)$$

para hallar las etiquetas $t_1, \ldots, t_T$. Las etiquetas adicionales t_{-1}, t_0 y t_T son delimitadores del principio y del final de la secuencia. Estas etiquetas adicionales mejoran levemente los resultados del etiquetado marcando una particularidad de TnT con respecto a otros etiquetadores.

Las probabilidades son estimadas a partir de un corpus etiquetado previamente (el ya mencionado corpus de entrenamiento). Para ello TnT utiliza probabilidades de máxima verosimilitud $\hat{P}$ obtenidas a partir de la frecuencia relativa y luego aplica una técnica de suavizado

$$\text{Unigramas:} \hat{P}(t_3) = \frac{f(t_3)}{N}$$

$$\text{Bigramas:} \hat{P}(t_3|t_2) = \frac{f(t_2, t_3)}{f(t_2)}$$

$$\text{Trigramas:} \hat{P}(t_3|t_1, t_2) = \frac{f(t_1, t_2, t_3)}{f(t_1, t_2)}$$

$$\text{Léxico:} \hat{P}(w_3|t_3) = \frac{f(w_3, t_3)}{f(t_3)}$$

donde t_1, t_2 y t_3 pertenecen al conjunto de etiquetas y w_3 pertenece al lexicon. N es el número de *tokens* del corpus de entrenamiento. La probabilidad de máxima verosimilitud se calcula como cero si el denominador o el nominador son cero.

2.9.2. Suavizado

TnT aplica una técnica de suavizado sobre las frecuencias contextuales. Esto tiene lugar debido al problema de los datos esparsos en las probabilidades de los trigramas. Es decir, no hay suficientes instancias de cada trigrama para calcular confiablemente su probabilidad asociada. Incluso establecer a cero la probabilidad de un trigrama que no aparece en el corpus genera el efecto indeseado de convertir la probabilidad de una secuencia completa en cero. TnT utiliza interpolación lineal de unigramas, bigramas y trigramas para realizar este proceso de suavizado. Es decir que se estima la probabilidad de un trigrama como sigue

$$P(t_3|t_1, t_2) = \lambda_1 \hat{P}(t_3) + \lambda_2 \hat{P}(t_3|t_2) + \lambda_3 \hat{P}(t_3|t_1, t_2)$$

donde $\hat{P}$ son los estimadores de máxima verosimilitud presentados anteriormente y λ_1, λ_2 y λ_3 son los pesos asociados a estos estimadores, tales que $\lambda_1 + \lambda_2 + \lambda_3 = 1$. TnT utiliza interpolación lineal con independencia de contexto. Es decir que λ_1, λ_2 y λ_3 tienen el mismo valor para todos los trigramas, o lo que es lo mismo, λ_1, λ_2 y λ_3 son independientes del trigrama que se está calculando. Los valores λ_1, λ_2 y λ_3 son estimados por interpolación de borrado. La idea es que se dará mayor peso a la información de unigrama, bigrama o trigrama más abundante. A continuación se presenta el algoritmo utilizado para realizar esta tarea

Algoritmo 1 Cálculo de λ_1, λ_2 y $\lambda_3 = 0$

Establecer $\lambda_1 = \lambda_2 = \lambda_3 = 0$
  por cada trigrama t_1, t_2, t_3 con $f(t_1, t_2, t_3) > 0$
    según el máximo de los tres valores siguientes:
      caso $\frac{f(t_1,t_2,t_3)-1}{f(t_1,t_2)-1}$: incrementar λ_1 en $f(t_1, t_2, t_3)$
      caso $\frac{f(t_2,t_3)-1}{f(t_2)-1}$: incrementar λ_2 en $f(t_1, t_2, t_3)$
      caso $\frac{f(t_3)-1}{N-1}$: incrementar λ_3 en $f(t_1, t_2, t_3)$
    fin
  fin
normalizar λ_1, λ_2 y λ_3

2.9.3. Manejo de palabras desconocidas

TnT, al igual que muchos otros etiquetadores gramaticales, maneja las palabras desconocidas mediantes análisis de sufijos. Los sufijos son fuertes predictores del tipo de palabra. Por ejemplo las palabras terminadas en *able* en el corpus *Wall Street Journal* son adjetivos (JJ) en el 98 % de los casos (ej.:*fashionable*, *variable*) y sustantivos (NN) en el 2 % restante.

La distribución de probabilidades para un sufijo particular es generada a partir de todas las palabras en el corpus de entrenamiento que comparten el mismo sufijo (de alguna longitud máxima predefinida). El término sufijo se entiende en este contexto como la secuencia final de letras de una palabra, que no coincide necesariamente con el significado lingüístico de sufijo.

La fórmula utilizada para calcular la probabilidad de que una etiqueta pertenezca a cierto sufijo es $P(t|l_{n-m+1}, \ldots, l_n)$, es decir, la probabilidad de una etiqueta t dadas las últimas letras l_i de una palabra de n letras. TnT aplica una técnica de suavizado utilizando sufijos cada vez más pequeños aplicando un peso θ_i a cada uno:

$$P(t|l_{n-m+1}, \ldots, l_n) = \frac{\hat{P}(t|l_{n-i+1}, \ldots, l_n) + \theta_i P(t|l_{n-i}, \ldots, l_n)}{1 + \theta_i}$$

para $i = m, \ldots, 0$, utilizando el estimador de máxima verosimilitud $\hat{P}$ para las frecuencias en el lexicon, los pesos θ_i y el caso base

$$P(t) = \hat{P}(t)$$

El estimador de máxima verosimilitud para un sufijo de longitud i es

$$\hat{P}(t|l_{n-i+1}, \ldots, l_n) = \frac{f(t, l_{n-i+1}, \ldots, l_n)}{f(l_{n-i+1}, \ldots, l_n)}$$

TnT utiliza desvío estándard del estimador de máxima verosimilitud para calcular los pesos θ_i.

Decisiones de diseño:

1. La primer decisión de diseño que afronta TnT es encontrar un buen valor para n, la longitud máxima de sufijo utilizada. TnT elige tomar la longitud del mayor sufijo encontrado en el corpus de entrenamiento, con la restricción de que sea menor o igual a 10.

2. Se utiliza independencia de contexto para calcular θ_i, la misma idea que se utilizó para calcular λ_i.

3. Se utilizan estimadores distintos para mayúsculas y minúsculas. Es decir, se mantienen dos árboles de sufijos distintos, uno para mayúsculas y otro para minúsculas.

4. La otra decisión relevante es: ¿Qué palabras del lexicon deben ser utilizadas para el manejo de sufijos? Basándose en el hecho de que las palabras desconocidas son infrecuentes, TnT utiliza sufijos de palabras infrecuentes. Por lo tanto, restringe el procedimiento de cálculo de probabilidades de sufijos a palabras con una frecuencia menor o igual a 10.

Adicionalmente, TnT discrimina la información sobre mayúsculas y minúsculas. Esto es debido a que las probabilidades de las etiquetas de palabras con mayúsuculas son distintas a las de las palabras con minúsculas. Para llevar esto a cabo se utilizan *flags* en las probabilidades contextuales. En lugar de

$$P(t_3|t_1, t_2)$$

se utiliza

$$P(t_3, c_3|t_1, c_1, t_2, c_2)$$

donde c_1, c_2 y c_3 son 1 si la palabra contiene mayúsculas y 0 en otro caso. Esto es equivalente a doblar el conjunto de etiquetas y utilizar etiquetas diferentes según si la palabra aparece en mayúscula o no.

La información presentada en este subcapítulo está basada en [3]

2.10. Etiquetador Gramatical Stanford Tagger

Stanford Tagger es un etiquetador gramatical estocástico basado en el modelo de máxima entropía.

Al igual que otros etiquetadores estocásticos, Stanford Tagger aprende de texto etiquetado: analiza y preserva información estadística sobre las palabras y las etiquetas asignadas. Dada una palabra w y un contexto h[6] el modelo asigna una probabilidad a cada etiqueta t perteneciente al conjunto de todas las etiquetas posibles T.

Como vimos anteriormente, la idea del modelo de máxima entropía es elegir la distribución de probabilidades p que tiene la mayor entropía entre todas las distribuciones que satisfacen ciertas restricciones.

Las restricciones obligan al modelo a comportarse de acuerdo a un conjunto de estadísticas obtenidas del corpus de entrenamiento. Estas estadísticas son expresadas como los valores esperados de funciones definidas sobre los conjuntos h y las etiquetas t.

Por ejemplo si se quiere restringir el modelo a etiquetar la palabra *make* como verbo o sustantivo con la misma frecuencia del corpus de entrenamiento, se pueden definir las características:

[6]El contexto generalmente se define como una secuencia de varias palabras y etiquetas precediendo a la palabra actual

$$f_1(h,t) = 1 \Leftrightarrow w_i = \text{make} \wedge t = \text{NN}$$
$$f_2(h,t) = 1 \Leftrightarrow w_i = \text{make} \wedge t = \text{VB}$$

A diferencia del modelo oculto de Markov, máxima entropía permite definir e incorporar información estadística más compleja que información de frecuencia, bigramas y/o trigramas.

Stanford Tagger define características generales clásicas (bigramas, trigramas y frecuencia de etiquetas) y también características especiales para palabras raras, con el objetivo de mejorar la capacidad de predicción del modelo en palabras desconocidas.

- **Características generales**

$$w_i = X \wedge t_i = T$$
$$t_{i-1} = T_1 \wedge t_i = T$$
$$t_{i-1} = T_1 \wedge t_{i-2} = T_2 \wedge t_i = T$$
$$w_{i+1} = X \wedge t_i = T$$

- **Características para palabras raras**

$$\text{El sufijo de } w_i = S \wedge |S| < 5 \wedge t_i = T$$
$$\text{El prefijo de } w_i = P \wedge 1 < |P| < 5 \wedge t_i = T$$
$$w_i \text{ contiene un número} \wedge t_i = T$$
$$w_i \text{ contiene una mayúscula} \wedge t_i = T$$
$$w_i \text{ contiene un guión} \wedge t_i = T$$

Las palabras raras son aquellas que aparecen pocas veces en el corpus de entrenamiento[7].

El rendimiento reportado para Stanford Tagger se encuentra dentro de los mismos parámetros de rendimiento de otros etiquetadores estocásticos. La ventaja es la capacidad de experimentar nuevas características o *features* que ayudan a mejorar su rendimiento.

La información presentada en este subcapítulo está basada en [4]

2.11. Diccionario Cobuild

Cobuild es una fuente de información que contiene un conjunto de entradas donde cada una está asociada a una palabra; posee una explicación de su significado, algunas características como su pronunciación, uno o más ejemplos que muestran su uso y clase gramatical y en algunos casos sinónimos.

A continuación se muestra un ejemplo de una entrada para la palabra *acid*.

[7]Stanford Tagger toma como palabras raras aquellas que aparecen menos de 7 veces en el corpus

`DICTIONARY_ENTRY`
`acid` ⟶ *palabra*
`acids` ⟶ *formas flexionadas*
`*!as!id` ⟶ *pronunciación*
`An acid fruit or drink has a sour or sharp taste.` ⟶ *definición*
`These oranges are very acid.` ⟶ *ejemplo*
`qualitative adjective` ⟶ *etiqueta específica*
`adjective` ⟶ *etiqueta general*

Podemos apreciar que en esta entrada Cobuild asigna la etiqueta *qualitative adjective* a la palabra *acid* que aparece en el ejemplo. Notemos que también aparece la etiqueta *adjective*: *qualitative adjective* es una etiqueta específica que brinda mayor información gramatical y sintáctica y *adjective* es una etiqueta general. Cada etiqueta general posee muchas etiquetas específicas.

Veamos ejemplos para la etiqueta general *adjective*:

Palabra: *abdominal*
Ejemplo: *They suffered abdominal pains.*
Etiqueta específica: ***classifying adjective: usually attributive***
Etiqueta general: ***adjective***

Palabra: *accessible*
Ejemplo: *...computers cheap enough to be accessible to virtually everyone.*
Etiqueta específica: ***qualitative adjective: predicative + to***
Etiqueta general: ***adjective***

Palabra: *acid*
Ejemplo: *These oranges are very acid.*
Etiqueta específica: ***qualitative adjective***
Etiqueta general: ***adjective***

Palabra: *abbreviated*
Ejemplo: *Her lecture was an abbreviated version of a talk she had given the previous year.*
Etiqueta específica: ***classifying adjective***
Etiqueta general: ***adjective***

Cobuild posee información gramatical sobre la palabra definida para cada uno de los ejemplos en donde ésta aparece. Con dicha información alcanza para construir un corpus parcialmente anotado. El proceso consiste en pegar o concatenar las palabras de los ejemplos y asignar la etiqueta de Coubild para la palabra asociada.

Entonces, para estas entradas del diccionario:

```
siren
sirens
s*a*!i*\%er\%e0n
A woman is described as a siren when she is attractive and dangerous to men.
One of the women, another of those sirens, haughtily regarded us as we talked.
countable noun
noun
```

```
sirloin
sirloins
s*\$e*:l!o!in
A sirloin is a piece of beef which is cut from the lower part of a cows back.
... a sirloin of Scotch beef.
mass noun
noun

sissy
sissies
s*!isi1
A boy is described as a sissy, especially by other boys, if he does not like
sport and is afraid to do things that are slightly dangerous.
Youre a lot of cry-babies and sissies ... ... Mummys little sissy boy.
countable noun: also vocative
noun
```

Se pueden concatenar sus ejemplos, traducir la información gramatical en etiquetas Penn Treebank y verse como:

```
One of the women, another of those sirens/NNS, haughtily regarded us as we talked.
... a sirloin/NN of Scotch beef.
Youre a lot of cry-babies and sissies/NNS ...
... Mummys little sissy/NN boy.
```

Esta última información conforma un corpus parcialmente anotado, es decir, un conjunto de oraciones donde alguna/s de las palabras que comprenden cada oración posee/n una etiqueta gramatical. Este corpus se utilizará como base para construir un nuevo corpus completamente anotado que servirá como una nueva fuente de información para entrenar etiquetadores gramaticales.

Claramente el primer paso para llevar a cabo esta tarea es elegir un diccionario y extraer la información mencionada anteriormente. El diccionario elegido fué Cobuild. A continuación se detallan las características que lo hicieron distintivo frente a otros diccionarios.

2.11.1. Características

Cobuild es un diccionario basado en la información del corpus *Bank of English*[8] . Su siglas significan: *Collins Birmingham University International Language Database*.

Las palabras incluídas en el diccionario fueron elegidas utilizando información sobre la frecuencia de ocurrencia de las mismas. Cada entrada posee una definición, ejemplos típicos de uso e información sobre la gramática, semántica y pronunciación.

Los ejemplos y la información gramatical asociada a la palabra definida son de particular interés para este trabajo y conforman la información de base que se utilizará para confeccionar el nuevo corpus.

[8]El corpus *Bank of English* contiene 650 millones de palabras seleccionadas del corpus *Collins* (compuesto por alrededor de 2.5 billones de palabras en inglés provenientes de websites, diarios, revistas, libros, material hablado de radio, TV y conversaciones diarias) para dar reflejo preciso y balanceado del inglés que se usa día a día.

2.11.2. Método de construcción

Cobuild se caracteriza por utilizar ejemplos reales obtenidos de un corpus en lugar de crearlos. Dicho corpus se compone de inglés hablado y escrito, americano y británico, etc. Incluso contiene transcripciones de conversaciones informales.

Cobuild se especializa en presentar las palabras y frases que son frecuentes en el uso diario. Lejos de ser un registro histórico del lenguaje es más bien una muestra del lenguaje contamporáneo.

2.11.3. Definiciones

Cobuild se distingue en el uso de frases completas en las definiciones. El significado de una palabra es establecido de la forma en que una persona ordinaria podría explicárselo a otra.

Generalmente los diccionarios ofrecen definiciones breves y tradicionales, mientras que Cobuild expone definiciones realmente amplias y ricas. Si se observan detenidamente las definiciones particulares se puede apreciar que cada palabra es elegida para ilustrar ciertos aspectos del significado. Y en la medida en que es posible, las palabras utilizadas en una definición son más frecuentes que la palabra que está siendo definida.

2.11.4. Ejemplos

Cobuild fué concebido teniendo especial atención en los ejemplos expuestos. Todos los ejemplos muestran patrones gramaticales, vocabulario y contextos típicos para cada palabra. Son piezas de texto genuinas elegidas en base al uso de la palabra que se está definiendo.

En consecuencia Cobuild presenta una cantidad exhaustiva del vocabulario inglés derivado de observaciones directas del lenguaje.

2.11.5. Información gramatical

Casi cada sentido de cada entrada en el diccionario Cobuild tiene junto a esta una clasificación gramatical, usualmente una clase de palabra y a menudo también una nota estructural. Esta es la información sobre la que se sustenta este trabajo, ya que en base a ella se construirá el nuevo corpus de entrenamiento.

2.12. Corpus BNC

El *British National Corpus*[9] (BNC) es un corpus de inglés británico cuyo tamaño es de alrededor de 100 millones de palabras. Está compuesto de una amplia gama de muestras de diferentes textos. La mayoría de estas muestras tienen un tamaño de entre 40 y 50 mil palabras; los textos publicados raramente aparecen completos.

El BNC fué diseñado para reflejar el uso del inglés británico contemporáneo. Está compuesto en un 90 % de inglés escrito y en un 10 % de transcripciones de

[9]Desde aquí en adelante se utilizarán indistintamente *British National Corpus* y su abreviación BNC

inglés hablado. El inglés escrito está compuesto a su vez por muestras tomadas de las siguientes fuentes:

- 60 % de libros
- 30 % de periódicos
- 10 % de misceláneos, textos publicados y textos no publicados

El 75 % de los textos de BNC está categorizado como informativo y el 25 % restante como imaginativo. La fecha de publicación de los textos informativos es de 1975 en adelante mientras que la fecha de publicación de los textos imaginativos data de 1960 en adelante.

El lenguaje hablado transcripto representa el 10 % del BNC, aportando alrededor de 10 millones de palabras. Las fuentes principales de este componente pueden clasificarse en:

- Encuentros informales grabados por individuos seleccionados por sexo, edad, clase social y región geográfica
- Encuentros más formales: Debates, lecturas, seminarios, programas de radio, etc.

Este segmento está compuesto en un 19 % por monólogos, en un 75 % por diálogos y un 6 % de material no clasificado.

Para obtener las grabaciones de encuentros informales se reclutaron 124 adultos, con aproximadamente la misma cantidad de hombres y mujeres, perteneciendo a una de 4 clases sociales distintas y a uno de 5 grupos de edades diferentes. Cada individuo utilizó un grabador portátil para grabar su propio discurso y las conversaciones que tuvieron con otras personas durante más de una semana.

BNC posee información gramatical (POS) para cada una de sus palabras. Las 100 millones de palabras de BNC fueron etiquetadas automáticamente por CLAWS4, un etiquetador automático desarrollado en la universidad de Lancaster. El conjunto de etiquetas utilizado para dicha tarea fué C5 (58 etiquetas gramaticales).

2.13. Corpus WSJ

El *Wall Street Journal*[10] (WSJ) es un corpus de inglés americano cuyo tamaño es de alrededor de 1 millón de palabras. Forma parte del proyecto *Peen Treebank*.

Como parte de este proyecto, WSJ fué etiquetado utilizando un proceso de 2 etapas: en la primer etapa se utilizó un etiquetador gramatical para asignar etiquetas automáticamente mientras que en la segunda se corrigieron los errores de etiquetado manualmente.

El etiquetador gramatical utilizado fué PARTS, un etiquetador estocástico desarrollado en los laboratorios de AT&T. Este etiquetador asigna etiquetas con un porcentaje de error de 3-5 %. PARTS genera etiquetas pertenecientes a un

[10] Desde aquí en adelante se utilizarán indistintamente *Wall Street Journal* y su abreviación WSJ

conjunto de etiquetas similar al conjunto *Brown* (levemente modificado). Por lo tanto, la salida de PARTS debe convertirse en etiquetas de *Penn Treebank*. Esta tarea introduce un error del 4 % ya que las etiquetas de *Penn Treebank* hacen ciertas distinciones que el conjunto de etiquetas PARTS no posee. Entonces el texto etiquetado posee un porcentaje de error total de 7-9 %.

Una vez finalizada esta etapa de etiquetación y conversión automática, las etiquetas son corregidas manualmente.

Capítulo 3

Desarrollo

Las entradas que conforman el diccionario Cobuild y que constituyen el conjunto de datos principal sobre el cual se basa este trabajo fueron cuidadosamente procesadas y refinadas intentando mantener toda la información disponible, explícita e implícita.

Se desarrollaron algoritmos para extraer de estas entradas los ejemplos junto a sus rótulos asociados. Se tradujeron las etiquetas gramaticales provistas por Cobuild en etiquetas gramaticales standard. Se analizaron todos estos procesos y se ajustaron hasta aplacar sus fallas.

El resultado obtenido fué un corpus parcialmente anotado correspondiente a la concatenación de los ejemplos extraídos.

Luego se completó la anotación utilizando etiquetación automática, obteniendo como resultado final un corpus completamente anotado.

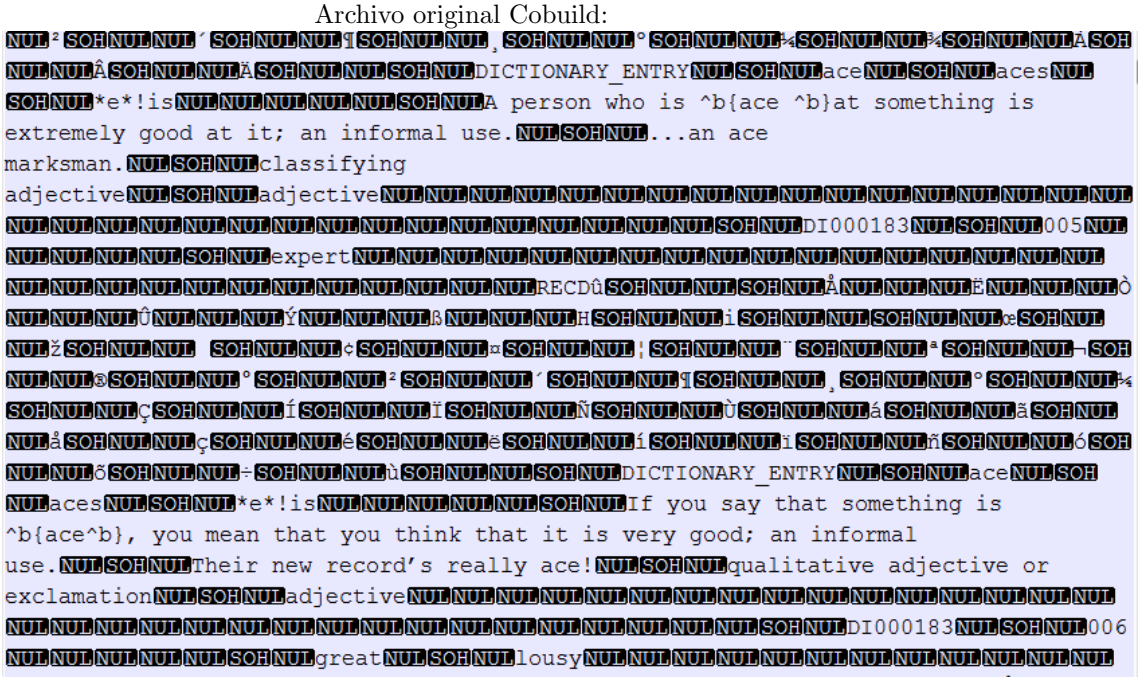
3.1. Extracción de la información

Cobuild guarda su información en un archivo de texto difícilmente legible con un formato carente de documentación conocida. El primer desafío de este trabajo consistió en identificar y obtener las entradas del archivo mencionado.

En ese sentido se crearon los algoritmos de extracción necesarios que consistieron en la eliminación de caracteres no ascii para poder obtener un archivo legible y en identificar y separar cada entrada.

A continuación se presentan extractos a modo de ejemplo:

Archivo original Cobuild:



Entradas extraídas correspondientes al fragmento anterior:

```
DICTIONARY_ENTRY
ace
aces
*e*!is
A person who is ace at something is extremely good at it; an informal use.
...an ace marksman.
classifying adjective
adjective

DICTIONARY_ENTRY
ace
aces
*e*!is
If you say that something is ace, you mean that you think that it is very good;
an informal use.
Their new records really ace!
qualitative adjective or exclamation
adjective
```

Las entradas de *Cobuild* se caracterizan por poseer una cantidad variable de campos difícilmente identificables. Sin embargo contienen algunos rasgos comu-

nes: la palabra, sus formas, la pronunciación, su definición y uno o más ejemplos donde se indica como se emplea (mediante una etiqueta gramatical).

Por ejemplo, en la primer entrada se pueden distinguir estos campos:

`DICTIONARY_ENTRY`
`ace` ⟶ *palabra*
`aces` ⟶ *formas flexionadas*
`*e*!is` ⟶ *pronunciación*
`A person who is ace at something is extremely good at it; an informal use.` ⟶ *definición*
`...an ace marksman.` ⟶ *ejemplo*
`classifying adjective` ⟶ *etiqueta específica*
`adjective` ⟶ *etiqueta general*

Se procesó cada entrada identificando la palabra que se está definiendo, las formas flexionadas de la misma y los ejemplos junto a su etiqueta gramatical asociada.

Como consecuencia se determinaron algunas características particulares para las entradas de Cobuild.

1. Algunas entradas presentan la pronunciación mientras que otras presentan detalles de la misma descriptos en lenguaje natural. Ejemplo:

```
DICTIONARY_ENTRY
abstract
abstracts, abstracting, abstracted
An idea, argument, or way of thinking that is abstract is based on general
ideas and principles rather than on particular things and events.
The arguments of contemporary science are so abstract that they are no longer intelligible...
...our capacity for abstract reasoning.
qualitative adjective
adjective
The word abstract is pronounced /*!abstr!akt/ when it is an adjective or a
noun, and /%e3bstr*!akt/ when it is a verb.
```

2. En la mayoría de los casos las palabras se definen con una oración, sin embargo existen entradas que presentan definiciones utilizando más de una oración como se muestra abajo:

```
DICTIONARY_ENTRY
account
accounts, accounting, accounted
%ek*a*!unt
The word account is also used in the following expressions. If you say that
something is the case by all accounts or from all accounts, you mean that everyone
you talk to about it, or everyone who writes about it, says that it is so.
From all accounts she was a clever girl.
phrase: used as an adjunct
phrase
```

Para el caso 1, las frases explicativas sobre pronunciación de las palabras fueron identificadas y descartadas, para no confundirlas con ejemplos.

Ante la imposibilidad de distinguir si una oración es parte de la definición o es parte de un ejemplo (caso 2), se asumió que la primera oración de la entrada corresponde a la definición (esto es cierto en la mayoría de los casos).

De esta manera se evita la pérdida de ejemplos por confundirlos con la definición. No obstante puede introducirse falsa información al identificar una oración perteneciente a la definición como un ejemplo.

Sin embargo este hecho no es tan grave: debido a las características de Cobuild, la palabra es generalmente definida utilizando el mismo sentido que exhibe el ejemplo.

Por lo tanto se puede concluir que inclusive para los pocos casos en que las oraciones pertenecientes a una definición se identifican como un ejemplo, éstas aportan información gramatical válida.

3.2. Traducción de etiquetas

La información gramatical que Cobuild presenta en cada uno de sus ejemplos no posee un formato conocido ni pertenece a ningún conjunto de etiquetas documentado.

Por ejemplo en la siguiente entrada:

`DICTIONARY_ENTRY`
`acid` ⟶ *palabra*
`acids` ⟶ *formas flexionadas*
`*!as!id` ⟶ *pronunciación*
`An acid fruit or drink has a sour or sharp taste.` ⟶ *definición*
`These oranges are very acid.` ⟶ *ejemplo*
`qualitative adjective` ⟶ *etiqueta específica*
`adjective` ⟶ *etiqueta general*

Podemos apreciar que la palabra *acid* está anotada como *qualitative adjective*. Notemos también la presencia de la anotación *adjective*. Ésta es una etiqueta general mientras que *qualitative adjective* es una etiqueta específica que brinda mayor información gramatical y sintáctica.

Como la idea de este trabajo es producir un corpus anotado a partir de este diccionario para utilizar como fuente de entrenamiento de etiquetadores gramaticales, es necesario que el conjunto de etiquetas empleado sea el mismo que emplea el *Gold Standard* para posteriormente poder medir los resultados. En ese sentido se realizó la traducción de etiquetas *Cobuild* en *Penn Treebank*.

Para llevar a cabo este proceso se construyó una tabla de conversión. Se realizó un análisis sobre las entradas de *Cobuild* arrojando como resultado la existencia de más de 4000 etiquetas diferentes.

A partir de este hecho se identificaron las etiquetas *Cobuild* que ocurren con mayor frecuencia y se seleccionó la etiqueta *Penn Treebank* equivalente para cada una de ellas, obteniendo una tabla de traducción ad hoc que se presenta a continuación:

Cuadro 3.1: *Tabla de traducción de etiquetas*

Etiqueta Cobuild	**Etiqueta Penn Treebank**
coordinating conjunction	CC
number	CD
determiner	DT
determiner + countable noun in singular	DT

Cuadro 3.1: *Tabla de traducción de etiquetas*

Etiqueta Cobuild	**Etiqueta Penn Treebank**
preposition	IN
subordinating conjunction	IN
preposition, or adverb after verb	IN
preposition after noun	IN
adjective	JJ
classifying adjective	JJ
qualitative adjective	JJ
adjective colour	JJ
ordinal	JJ
adjective after noun	JJ
modal	MD
adverb	RB
noun	NN
uncountable noun	NN
noun singular	NN
countable or uncountable noun	NN
countable noun with supporter	NN
uncountable or countable noun	NN
noun singular with determiner	NN
mass noun	NN
uncountable noun with supporter	NN
partitive noun	NN
noun singular with determiner with supporter	NN
countable noun + of	NN
countable noun, or by + noun	NN
countable noun or partitive noun	NN
count or uncountable noun	NN
countable noun or vocative	NN
partitive noun + uncountable noun	NN
noun singular with determiner + of	NN
noun in titles	NN
noun vocative	NN
uncountable noun + of	NN
indefinite pronoun	NN
uncountable noun, or noun singular	NN
countable noun, or in + noun	NN
partitive noun + noun in plural	NN
countable or uncountable noun with supporter	NN
uncountable noun, or noun before noun	NN
uncountable or countable noun with supporter	NN
noun before noun	NN
noun plural with supporter	NNP
noun in names	NNP
proper noun or vocative	NNP
proper noun	NNP
noun plural	NNS
predeterminer	PDT

Cuadro 3.1: *Tabla de traducción de etiquetas*

Etiqueta Cobuild	**Etiqueta Penn Treebank**
pronoun	PP
possessive	PPS
adverb with verb	RB
adverb after verb	RB
sentence adverb	RB
adverb + adjective or adverb	RB
adverb + adjective	RB
preposition or adverb	RB
adverb after verb, or classifying adjective	RB
adverb or sentence adverb	RB
adverb with verb, or sentence adverb	RB
exclamation	UH
exclam	UH
verb	VB
verb + object	VB
verb or verb + object	VB
ergative verb	VB
verb + adjunct	VB
verb + object + adjunct	VB
verb + object (noun group or reflexive)	VB
verb + object or reporting clause	VB
verb + object (reflexive)	VB
verb + object, or phrasal verb	VB
verb + to-infinitive	VB
ergative verb + adjunct	VB
verb + object + adjunct (to)	VB
verb + object, or verb + adjunct	VB
verb + object + adjunct (with)	VB
verb + adjunct (with)	VB
verb + complement	VB
verb + object, or verb	VB
verb + object + to-infinitive	VB
verb + reporting clause	VB
verb or ergative verb	VB
verb + adjunct (from)	VB
wh: used as determiner	WDT
wh: used as relative pronoun	WP
wh: used as pronoun	WP
wh: used as adverb	WRB

El proceso de traducción de etiquetas consiste en intentar encontrar en la tabla la etiqueta *Penn Treebank* correspondiente a la etiqueta específica de *Cobuild* (en el ejemplo anterior*qualitative adjective*), si no fuera el caso se busca la etiqueta *Penn Treebank* correspondiente a la etiqueta general *Cobuild*(*adjective* para el ejemplo).

Utilizando este método se logró traducir aproximadamente el 99.26 % de las etiquetas.

Finalizado este proceso se verificó que el etiquetado haya sido correcto: se realizó una rutina que etiquetara automáticamente todos los ejemplos de Cobuild (utilizando el etiquetador TnT) y se generó una matriz de confusión comparando las etiquetas extraídas y traducidas provenientes de Cobuild contra las asignadas por TnT.

El resultado fué de 71 % de aciertos. Se analizaron las etiquetas que diferían, luego se corrigieron las traducciones y se ajustaron los algoritmos de extracción y traducción de etiqueteas hasta alcanzar un grado de error mínimo.

Los mayores focos de error que no pudieron ser aplacados corresponden a palabras etiquetadas como VBD cuando son VBN y viceversa. Estas etiquetas son difícilmente desambigüables automáticamente.

3.2.1. Recuperación de precisión gramatical

Una vez obtenidos los ejemplos con las etiquetas traducidas a Penn Treebank, se realiza un nuevo proceso para aportar o recuperar precisión gramatical perdida en la traducción: se comparan las etiquetas traducidas contra las etiquetas asignadas automáticamente por TnT. En caso de coincidir y en caso de que la etiqueta TnT aporte mayor precisión, se asigna esta última. Se utilizó el siguiente algoritmo:

Algoritmo 2 Obtener la etiqueta de mayor detalle gramatical

Entrada: Etiqueta obtenida de Cobuild, etiqueta asignada por TnT

Si se obtuvo de Cobuild la etiqueta:

NN y TnT asignó **NNS, NNP o NNPS**: Asignar la etiqueta TnT

NNS y TnT asignó **NNPS**: Asignar la etiqueta NNPS

VBN—VBD:

Si TnT asignó **VBN o VBD**, asignar la etiqueta TnT
Si no asignar la etiqueta VB

VB y TnT asignó **VBN, VBD, VBZ, VBP o VBG**: Asignar la etiqueta TnT

JJ y TnT asignó **JJR o JJS**: Asignar la etiqueta TnT

RB y TnT asignó **RBR o RBS**: Asignar la etiqueta TnT

WP y TnT asignó **WP$**: Asignar la etiqueta WP$

PRP y TnT asignó **PRP$**: Asignar la etiqueta PRP$

A continuación se exhiben ejemplos de este proceso:

Etiqueta extraída de Cobuild	
It	
cannot	
produce	
enough	
heat	
to	
activate	***VB***
the	
electrons	
.	

$\longrightarrow$

Etiquetado automático	
It	
cannot	
produce	
enough	
heat	
to	
activate	***VBP***
the	
electrons	
.	

Etiqueta extraída de Cobuild	
Armed	
with	
this	
information	
,	
parents	
will	
be	
better	***RB***
able	
to	
cater	
for	
their	
childrens	
needs	
...	

$\longrightarrow$

Etiquetado automático	
Armed	
with	
this	
information	
,	
parents	
will	
be	
better	***RBR***
able	
to	
cater	
for	
their	
childrens	
needs	
...	

Etiqueta extraída de Cobuild	
They	
have	
their	
first	
degrees	
and	
are	
studying	
for	
higher	***JJ***
degrees	
...	

$\longrightarrow$

Etiquetado automático	
They	
have	
their	
first	
degrees	
and	
are	
studying	
for	
higher	***JJR***
degrees	
...	

En los ejemplos presentados se puede observar como el etiquetado automático aporta información gramatical a las palabras anotadas durante el proceso de extracción y traducción.

Luego de ejecutado este proceso el análisis de comparación contra las etiquetas asignadas por TnT dió un 87.5 % de aciertos .

Cabe destacar que más allá de los esfuerzos por preservar la información gramatical, inevitablemente se generó una pérdida de información semántica durante el proceso de traducción ya que las etiquetas *Penn Treebank* son menos específicas que las etiquetas de Cobuild. En otras palabras, las etiquetas *Penn Treebank* carecen del rico detalle lingüístico que las etiquetas Cobuild poseen, originando una pérdida natural de información en la traducción.

3.2.2. Reconocimiento de formas flexionadas

Las entradas de Cobuild exponen formas flexionadas de la palabra: plurales, pasados, etc. En muchas de ellas ocurre la palabra que se está definiendo y uno o más ejemplos en donde aparecen formas flexionadas de la misma junto con sus rótulos.

El objetivo entonces es reconocer y registrar esta información gramatical implícita.

Tomemos la siguiente entrada:

`DICTIONARY_ENTRY`
`celebrate` ⟶ *palabra*
`celebrates, celebrating, celebrated` ⟶ *formas flexionadas*
`s*!el%e1bre!it` ⟶ *pronunciación*
`If you celebrate someone or something, you praise them for their good qualities;`
`a fairly formal use.` ⟶ *definición*
`People were celebrating him as a bright alternative to Nixon.` ⟶ *ejemplo*
`verb + object, or verb + object + adjunct (as/for)` ⟶ *etiqueta específica*
`verb` ⟶ *etiqueta general*

Aquí arriba se puede observar una entrada del diccionario para la palabra *celebrate*, que contiene la definición y un ejemplo en donde aparece la forma derivada *celebrating* junto a una etiqueta gramatical asociada:

People were ***celebrating*** *him as a bright alternative to Nixon.*

En la entrada presentada la palabra definida es *celebrate* y las formas derivadas son *celebrates, celebrating* y *celebrated*. Con esta información y la etiqueta asignada por *Cobuild* (*verb + object, or verb + object + adjunct (as/for)*) se pueden inferir y generar etiquetas *Penn Treebank* para dichas formas.

En lugar de guardar la etiqueta *Penn Treebank* VB correspondiente a *verb* para la palabra *celebrating*, guardaríamos la etiqueta VBG[1] que contiene más información gramatical.

La tarea aquí será reconocer *celebrating* como verbo gerundio o presente participio a partir de que está etiquetada como verbo y que deriva de *celebrate*. Es decir, inferir el tipo de la forma derivada a partir de la palabra y la etiqueta asignada por Cobuild.

En este sentido se desarrollaron reglas y métodos para aprovechar toda la información presente. Entonces, a partir de la palabra, la forma en que ocurre

[1] verbo gerundio o presente participio

y la etiqueta asignada se aplican las siguientes reglas para reconocer etiquetas gramaticales para formas flexionadas:

Algoritmo 3 Reconocimiento de formas flexionadas

Entrada: Etiqueta asignada por Cobuild, forma flexionada

Traducir la etiqueta asignada por Cobuild a PenTreeBank
Si la etiqueta obtenida es

JJ:
Si la forma termina en *er* o empieza en *more* o *less* aplicar **JJR**
Si la forma termina en *est* o empieza en *most* o *least* aplicar **JJS**

RB:
Si la forma termina en *er* o empieza en *more* o *less* aplicar **RBR**
Si la forma termina en *est* o empieza en *most* o *least* aplicar **RBS**

NN:
Si la forma termina en *s* aplicar **NNS**

VB:
Si la forma termina en *ed* aplicar **VBD|VBN**
Si la forma termina en *ing* aplicar **VBG**
Si la forma termina en *s* aplicar **VBZ**

A continuación se presentan algunos ejemplos del resultado de aplicar este proceso:

1. Forma derivada *celebrating* inferida como VBG a partir de la palabra *celebrate* y la etiqueta VB:

```
DICTIONARY_ENTRY
celebrate
celebrates, celebrating, celebrated
s*!el%e1bre!it
If you celebrate someone or something, you praise them for their good qualities;
a fairly formal use.
People were celebrating him as a bright alternative to Nixon.
verb + object, or verb + object + adjunct (as/for)
verb
```

Resultado: *People were celebrating/**VBG** him as a bright alternative to Nixon.*

2. Forma derivada *faults* inferida como NNS a partir de la palabra *fault* y la etiqueta NN:

```
DICTIONARY_ENTRY
fault
faults, faulting, faulted
f*!o*:lt
A fault on a machine or in a structure is a broken part or a mistake in
the way it was made.
Send it back to the manufacturer if the machine develops the same fault...
Technicians laboriously tried to find and remedy faults.
countable noun
noun
```

Resultado: *Technicians laboriously tried to find and remedy faults/**NNS**.*

3. Forma derivada *larger* inferida como JJR a partir de la palabra *large* y la etiqueta JJ:

```
DICTIONARY_ENTRY
large
larger, largest
l*%a*:d!z
If you say that someone or something is larger than life, you mean that they
appear or behave in a way that seems more important or exaggerated than usual.
The central character is a larger than life, cantankerous New Englander...
...a larger-than-life version of our present society.
classifying adjective
adjective
```

Resultado: *The central character is a larger/**JJR** than life, cantankerous New Englander...*

3.3. Nuevo Corpus generado

A partir del corpus parcialmente anotado generado en la etapa anterior, se completarán las anotaciones con un etiquetador automático (TnT) preservando las etiquetas obtenidas a partir de la información gramatical proveniente del diccionario *Cobuild*.

A continuación se exhibe un ejemplo de este proceso:

Entrada *Cobuild* para la palabra *abide*

```
DICTIONARY_ENTRY
abide
abides, abiding, abided
%eb*a*!id
If something abides, it continues to happen or exist for a long time.
We feel the need to lean on something that abides.
verb
verb
```

Resultado de extracción, reconocimiento y traducción de etiquetas y formas flexionadas correspondiente a la entrada anterior:

We
feel
the
need
to
lean
on
something
that
abides *VBZ*
.

Se puede apreciar que en el ejemplo se ha reconocido *abides* como verbo en tercera persona a partir de *abide* y el rótulo *verb*, asignando la etiqueta gramatical traducida correspondiente: VBZ (obtenida por inferencia).

El próximo paso será el de completar las anotaciones gramaticales para las palabras restantes. Este proceso se realiza anotando el corpus con el etiquetador gramatical automático TnT, como puede verse en 2). Luego se une el corpus anotado parcialmente procedente de Cobuild 1) con 2), preservando todas las etiquetas del diccionario.

El resultado es un nuevo corpus obtenido a partir de Cobuild, con las anotaciones que este provee y completado con anotaciones automáticas 3).

1) Ejemplo extraído	
We	
feel	
the	
need	
to	
lean	
on	
something	
that	
abides	***VBZ***
.	

.

⟶

2) Etiquetado automático	
We	*PRP*
feel	*VBP*
the	*DT*
need	*NN*
to	*TO*
lean	*VB*
on	*IN*
something	*NN*
that	*IN*
abides	***NNS***
.	.

⟶

3) Nuevo corpus	
We	*PRP*
feel	*VBP*
the	*DT*
need	*NN*
to	*TO*
lean	*VB*
on	*IN*
something	*NN*
that	*IN*
abides	***VBZ***
.	.

Capítulo 4

Experimentación

4.1. Primer experimento

El primer experimento consiste en medir (generando una matriz de confusión) la información extraída de Cobuild contra la misma información generada a partir de un etiquetador automático (TnT). De esta manera podremos observar la diferencia entre la información gramatical de Cobuild y la información que se podría generar automáticamente.

Como se mencionó anteriormente la información extraída de Cobuild, es la unión de ejemplos con la información gramatical correspondiente a la palabra definida. A continuación se presenta un pequeño extracto:

She
put
out
a
hand
and
stroked
the
cat ***NN***
softly
...

...
domestic
animals
such
as
dogs
and
cats ***NNS***
.

Esta es la información extraída de Cobuild para la palabra *cat*; la unión de los ejemplos

She put out a hand and stroked the cat softly...
...domestic animals such as dogs and cats.

Se puede notar la información gramatical expresada mediante las etiquetas NN y NNS para las palabras *cat* y *cats* respectivamente. La idea de este experimento será comparar estas etiquetas contra las etiquetas asignadas por el etiquetador automático TnT. Entonces se tomará este corpus plano (sin etiquetas), se lo etiquetará utilizando TnT entrenado con el corpus de entrenamiento WSJ y luego se realizará la comparación.
La matriz de confusión[1] generada a partir de dicha comparación es la siguiente:

Cuadro 4.1: *Diferencias entre etiquetas generadas por TnT vs extraÃdas de Cobuild*

Cobuild / TnT	**NN**	**JJ**	**VB**	**VBD—VBN**	**RB**	**VBG**	**VBZ**	**NNS**	**IN**	**PPS**
JJ	**771**	-	141	**294**	**234**	97	2	83	38	-
NN	-	**699**	**653**	1	99	183	-	47	111	-
VBN	10	**669**	-	-	6	-	-	10	1	-
NNP	**403**	74	49	1	14	1	-	31	2	-
VB	**250**	61	-	2	9	-	-	1	5	-
RB	28	**238**	7	2	-	1	-	3	60	-
VBG	113	**238**	-	-	6	-	-	2	9	-
NNS	174	8	1	-	7	1	146	-	5	2
VBZ	-	-	-	-	-	-	-	138	3	-
IN	7	20	9	-	132	-	-	1	-	-

Aciertos: 49.468 (87,58 %)
Errores: 7.018 (12,42 %)

Se puede apreciar un alto porcentaje de aciertos entre las etiquetas extraídas de Cobuild (87,58 %) y las etiquetas asignadas por TnT. Este porcentaje indica que la información de etiquetas extraídas de Cobuild es consistente con las producidas por TnT. La mayoría de los errores se da en etiquetas NN, JJ y VB de Cobuild cuando son etiquetadas como JJ, NN y VB y VBN por TnT respectivamente. A continuación se muestran algunas ejemplos de los errores:

Etiquetado por TnT como NN pero extraído como VB de Cobuild

- **share**: Lets share the petrol costs...
- **name**: Name the place, well be there...

Etiquetado por TnT como JJ pero extraído como NN de Cobuild

- **flat**: A flat usually includes a kitchen and bathroom.

[1]Las matrices de confusión presentadas de aquí en adelante contienen las primeras 10 etiquetas de mayor error

- **wireless**: messages sent by cable or wireless

Etiquetado por TnT como NN pero extraído como JJ de Cobuild

- **firm**: Bake the cake for about an hour until it is firm and brown
- **kind**: I find them all very pleasant and extremely kind and helpful

Etiquetado por TnT como VBN pero extraído como JJ de Cobuild

- **settled**: They are practising settled agriculture

4.2. Segundo experimento: Etiquetar el corpus WSJ

El segundo experimento realizado tiene como objetivo evaluar la nueva fuente de información obtenida (NFI) como corpus de entrenamiento. Para esto se entrenará el etiquetador gramatical TnT y se etiquetará con él el corpus Wall Street Journal (WSJ). Posteriormente se realizarán mediciones de desempeño pertinentes.

4.2.1. Etiquetar el corpus WSJ con TnT

Este experimento consiste en entrenar TnT con WSJ y con WSJ + NFI. Luego se procede a etiquetar el WSJ plano (sin etiquetas gramaticales) con estos dos modelos. Por último se contruye la matriz de confusión:

Cuadro 4.2: *Matriz de confusion para*
WSJ_1 = WSJ original contra
WSJ_2 = WSJ etiquetado con TnT (entrenado con WSJ)

WSJ_1 \ WSJ_2	**JJ**	**NNP**	**VBN**	**RB**	**IN**	**RP**	**NNPS**	**VBD**	**NN**	**VB**
NN	**2534**	**1661**	30	73	18	7	-	37	-	295
VBD	60	6	**1492**	-	-	-	-	-	40	38
IN	107	29	-	**1461**	-	**1363**	-	-	4	2
RB	738	44	4	-	**1460**	854	-	1	202	40
NNP	449	-	4	46	22	1	**1261**	3	370	7
VBN	**1241**	22	-	-	-	-	-	**1137**	37	52
JJ	-	730	**916**	572	67	23	-	49	898	68
VBG	422	18	-	-	-	-	-	-	882	-
VBP	26	10	29	11	9	1	1	52	292	715
VB	67	28	46	14	13	-	-	29	362	-

Aciertos: 1.230.151 (97,39 %)
Errores: 32.969 (2,61 %)

Cuadro 4.3: *Matriz de confusion para*
WSJ_1 = WSJ original contra
NFI_1 = WSJ etiquetado con TnT (entrenado con WSJ + NFI)

WSJ_1 \ NFI_1	**JJ**	**NNP**	**IN**	**VBN**	**RP**	**RB**	**NNPS**	**VBD**	**NN**	**VB**
NN	**2589**	**1919**	19	32	6	81	-	42	-	286
RB	774	55	**1610**	4	**1030**	-	-	1	205	39
VBD	71	9	-	**1593**	-	-	-	-	39	28
IN	104	42	-	-	**1513**	**1329**	-	-	4	3
VBN	**1322**	25	-	-	-	-	-	**1235**	36	42
NNP	422	-	21	4	1	50	**1257**	4	344	11
JJ	-	849	79	897	25	621	-	46	902	66
VBP	31	11	10	35	1	7	1	58	339	864
VBG	491	23	1	-	-	-	-	-	839	-
VB	77	32	15	47	1	15	-	43	379	-

Aciertos: 1.228.312 (97,24 %)
Errores: 34.808 (2,76 %)

Se puede observar que el rendimiento del etiquetador TnT entrenado con WSJ apenas mejor (97,39 %) que el rendimiento de TnT entrenado con WSJ+NFI (97,24 %). La mayoría de los errores para TnT entrenado con WSJ se da en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP por TnT. Para TnT entrenado con WSJ + NFI la mayoría de los errores se da en las mismas etiquetas, pero con cantidad de errores mayor, sobre todo para NN etiquetado como NNP.

La segunda evaluación de este experimento consiste en entrenar TnT con la mitad de WSJ y con la mitad de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta la mitad restante de WSJ y se construye la matriz de confusión. Se realiza la misma operación para cada mitad:

Cuadro 4.4: *Matriz de confusion para*
WSJ_3 = 1 mitad WSJ original contra
TnT_1 = 1 mitad WSJ etiquetado con TnT (entrenado con 2 mitad de WSJ)

TnT_1 / WSJ_3	**JJ**	**NNP**	**VBN**	**NN**	**VBD**	**IN**	**RB**	**RP**	**VB**	**NNPS**
NN	**1959**	**1154**	26	-	24	5	60	2	269	2
VBD	76	12	**1129**	19	-	-	1	-	29	-
JJ	-	545	**801**	**1039**	52	19	313	9	62	1
VBN	**617**	23	-	24	**819**	-	-	-	36	-
RB	432	25	3	91	2	**808**	-	318	19	-
IN	71	24	1	3	-	-	**634**	**615**	1	-
VBP	26	19	19	285	33	6	4	-	613	1
NNP	419	-	8	534	11	19	43	-	20	600
VBG	276	22	-	577	-	-	-	-	-	-
NNPS	26	549	-	-	-	-	-	-	-	-

Aciertos: 607.876 (96,25 %)
Errores: 23.695 (3,75 %)

Cuadro 4.5: *Matriz de confusion para*
WSJ_3 = 1 mitad WSJ original contra
NFI_2 = 1 mitad WSJ etiquetado con TnT (entrenado con 2 mitad de WSJ + NFI)

NFI_2 / WSJ_3	**JJ**	**NNP**	**VBN**	**NN**	**IN**	**VBD**	**RP**	**VB**	**NNPS**	**RB**
NN	**1751**	**1210**	26	-	9	22	1	207	1	59
VBD	60	15	**980**	14	-	-	-	19	-	-
JJ	-	586	**634**	**859**	33	30	5	43	-	341
RB	433	26	2	88	**854**	1	459	19	-	-
VBN	**612**	26	-	20	-	**760**	-	22	-	-
IN	66	26	-	4	-	-	**724**	1	-	557
VBP	21	18	21	253	4	30	-	**612**	1	3
NNP	362	-	7	532	19	5	-	18	571	44
NNPS	20	513	-	2	-	-	-	-	-	-
VBG	311	17	-	476	2	-	-	-	-	-

Aciertos: 609.270 (96,47 %)
Errores: 22.301 (3,53 %)

Cuadro 4.6: *Matriz de confusion para*
WSJ_4 = 2 mitad WSJ original contra
TnT_2 = 2 mitad WSJ etiquetado con TnT (entrenado con 1 mitad de WSJ)

TnT_2 / WSJ_4	**JJ**	**VBN**	**NNP**	**NN**	**RB**	**VBD**	**NNPS**	**IN**	**RP**	**VB**
NN	**1826**	35	**1089**	-	51	27	-	11	6	256
VBD	73	**1097**	12	37	-	-	-	-	-	19
JJ	-	609	559	**1085**	360	69	2	44	18	78
IN	44	-	19	6	**881**	-	-	-	**693**	4
VBN	**838**	-	30	22	-	**859**	-	-	-	33
NNP	457	17	-	458	40	6	**855**	20	2	18
RB	384	3	45	163	-	-	-	**741**	517	19
VBP	35	17	14	187	8	31	-	7	1	560
VBG	294	-	21	552	1	-	-	-	-	1
VB	74	25	49	405	9	23	-	7	-	-

Aciertos: 607.593 (96,21 %)
Errores: 23.956 (3,79 %)

Cuadro 4.7: *Matriz de confusion para*
WSJ_4 = 2 mitad WSJ original contra
NFI_3 = 2 mitad WSJ etiquetado con TnT (entrenado con 1 mitad de WSJ + NFI)

NFI_3 / WSJ_4	**JJ**	**NNP**	**VBN**	**NN**	**RP**	**IN**	**NNPS**	**VBD**	**RB**	**VB**
NN	**1725**	**1194**	27	-	4	11	-	21	47	199
VBD	49	9	**1009**	18	-	-	-	-	-	12
JJ	-	574	502	**876**	20	52	1	41	380	60
IN	41	25	-	4	**827**	-	-	-	**745**	3
VBN	**822**	26	-	20	-	-	-	**782**	-	25
RB	418	42	2	142	616	**817**	-	-	-	18
NNP	401	-	13	452	1	12	**813**	4	29	18
VBP	25	7	21	205	1	8	-	29	4	585
VBG	299	24	-	480	-	1	-	-	1	-
VBZ	2	10	-	6	-	-	7	-	-	-

Aciertos: 608.663 (96,38 %)
Errores: 22.886 (3,62 %)

Se puede apreciar una leve mejora en el porcentaje de etiquetas acertadas para el modelo que incorpora NFI; 96,25 % contra 96,47 % y 96,21 % contra 96,38 % para cada mitad respectivamente. Los errores más comunes son producidos en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP por TnT, para las dos mitades entrenadas tanto con WSJ como con WSJ

+ NFI. Se puede notar que el porcentaje de error al etiquetar JJ cuando era NN es menor en la evaluación realizada sobre TnT entrenado con el modelo que incorpora NFI.

A continuación se presenta la diferencia entre las etiquetas generadas a partir de WSJ vs WSJ + NFI.

Específicamente se muestran las matrices de confusión entre las mitades de WSJ etiquetado con TnT entrenado con la mitad restante con y sin NFI.

Cuadro 4.8: *Matriz de confusion para*
TnT_1 = 1 mitad WSJ etiquetado por TnT (entrenado con 2 mitad WSJ) vs
NFI_2 = 1 mitad WSJ etiquetado con TnT (entrenado con 2 mitad de WSJ + NFI)

NFI_2 / TnT_1	**NN**	**VBD**	**JJ**	**VBN**	**NNP**	**VB**	**RP**	**VBG**	**VBP**	**IN**
JJ	**453**	39	-	132	208	43	1	61	6	21
VBN	15	**438**	**308**	-	5	6	-	-	1	-
NN	-	8	**434**	22	**307**	**240**	-	**222**	65	5
VBD	7	-	36	**361**	9	7	-	-	6	-
NNP	**240**	2	116	1	-	15	-	11	1	-
RB	30	1	118	-	11	2	**227**	-	-	174
VBP	71	13	6	1	-	192	-	-	-	1
VB	172	24	57	22	6	-	1	-	184	2
VBG	118	1	174	-	15	1	-	-	-	3
IN	1	-	4	-	4	-	152	-	1	-

Aciertos: 623.785 (98,77 %)
Errores: 7.789 (1,23 %)

Cuadro 4.9: *Matriz de confusion para*
TnT_2 = 2 mitad WSJ etiquetado por TnT (entrenado con 1 mitad WSJ) vs
TnT_3 = 2 mitad WSJ etiquetado con TnT (entrenado con 1 mitad de WSJ + NFI)

NFI_3 / TnT_2	**JJ**	**VBD**	**VBN**	**NN**	**NNP**	**RP**	**IN**	**VB**	**VBG**	**VBP**
NN	**508**	18	19	-	**281**	-	6	**195**	**188**	29
VBN	**235**	**434**	-	21	4	-	-	10	-	2
VBD	51	-	**407**	12	2	-	-	5	-	3
JJ	-	39	153	**397**	170	-	13	37	78	13
RB	141	-	-	30	21	**245**	**225**	10	-	1
VBP	13	4	6	76	3	-	-	187	-	-
NNP	109	3	5	175	-	-	7	24	4	6
VB	43	16	21	154	14	1	1	-	1	148
VBZ	1	-	-	6	-	-	-	1	-	-
VBG	137	2	1	98	14	-	-	6	-	2

Aciertos: 624.128 (98,82 %)
Errores: 7.422 (1,18 %)

La tercer evaluación de este experimento consiste en entrenar TnT con un cuarto de WSJ y con un cuarto de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta los 3/4 restantes de WSJ y se construye la matriz de confusión. Se realiza la misma operación para cada uno de los cuartos.

En promedio de la cantidad de aciertos para el modelo entrenado con WSJ es 95.91 %, mientras que para el modelo entrenado con WSJ+NFI es 96.26 %

Cuadro 4.10: *Rendimiento de TnT entrenado con cuartos de WSJ con y sin NFI*

Evaluación	**Porcentaje de aciertos**
TnT entrenado con el primer 1/4 de WSJ	95.93 %
TnT entrenado con el primer 1/4 de WSJ + NFI	96.25 %
TnT entrenado con el segundo 1/4 de WSJ	95.89 %
TnT entrenado con el segundo 1/4 de WSJ + NFI	96.24 %
TnT entrenado con el tercer 1/4 de WSJ	95.92 %
TnT entrenado con el tercer 1/4 de WSJ + NFI	96.27 %
TnT entrenado con el cuarto 1/4 de WSJ	95.9 %
TnT entrenado con el cuarto 1/4 de WSJ + NFI	96.28 %

En todos los casos se puede apreciar una mejora en el acierto de etiquetas para el modelo que incorpora NFI.

La cuarta evaluación de este experimento consiste en entrenar TnT con un décimo de WSJ y con un décimo de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta los 9/10 restantes de WSJ y se presentan los resultados:

- 95.32 % de acierto de etiquetas para el etiquetado de 9/10 de WSJ con TnT entrenado con 1/10 WSJ
- 96.02 % de acierto de etiquetas para el etiquetado de 9/10 de WSJ con TnT entrenado con 1/10 WSJ+NFI

Se puede apreciar un aumento del porcentaje de aciertos en el modelo que incorpora NFI.

4.3. Experimentos adicionales

Se realizaron experimentos similares al descripto en la sección anterior para explorar la existencia de variaciones en los resultados a partir de la utilización de otro corpus u otro etiquetador gramatical. En tal sentido se empleó el etiquetador gramatical automático de máxima entropía Stanford Tagger. También se repitieron los mismos experimentos utilizando ambos etiquetadores gramaticales (Stanford Tagger y TnT) sobre el corpus BNC.

Los experimentos consistieron en comparar el resultado de ejecutar el etiquetador gramatical entrenado con WSJ vs WSJ más la incorporación de la nueva

fuente de información generada (NFI) sobre el corpus. Se realizó este mismo experimento sobre particiones; mitades, cuartos y décimos del corpus, intentando encontrar variaciones.

Para todos los casos se observaron resultados similares; un leve aumento del porcentaje de aciertos en el modelo que incorpora NFI. El porcentaje de aciertos aumenta a medida que la partición es menor.

Los datos y el detalle de estos experimentos se puede consultar en el apéndice.

Capítulo 5

Conclusiones

Se realizó un gran esfuerzo en el preprocesamiento del diccionario Cobuild. A partir de un archivo con formato desconocido la primera etapa consistió en identificar cada entrada y hacer que el archivo fuera legible. En las etapas siguientes se enfocó el esfuerzo en descifrar el formato de cada entrada y en entender la información gramatical obtenida.

Se crearon algoritmos para poder distinguir ejemplos de definiciones y se verificó el correcto funcionamiento de los mismos. Se detectaron errores de extracción de etiquetas para ciertos casos particulares, por consiguiente se realizaron ajustes en los algoritmos hasta obtener resultados satisfactorios.

Partiendo del hecho de que las etiquetas de Cobuild no poseen un formato conocido ni pertenecen a ningún conjunto de etiquetas documentado, hubo que decidir como realizar la conversión a etiquetas Penn Treebank. El primer análisis determinó que Cobuild posee más de 4000 etiquetas, por lo tanto se realizó un relevamiento de las etiquetas de mayor ocurrencia, que fueron incluídas en la tabla de conversión junto con la etiqueta Penn Treebank equivalente.

Cabe aclarar que las etiquetas Penn Treebank poseen un nivel de detalle gramatical menor a las etiquetas de Cobuild (que son muy ricas gramaticalmente), en consecuencia se produjo una pérdida natural de información en la tarea de mapeo.

Se tuvieron en cuenta palabras derivadas y cuando fué posible (cuando no hubo ambiguedad) se infirieron etiquetas para las mismas. Se realizaron trabajos de verificación de etiquetado, etiquetando automáticamente Cobuild para luego analizar las diferencias contra el resultado del proceso de traducción de etiquetas.

En muchos casos se analizaron palabras particulares donde las etiquetas no coincidían. Se corrigieron las traducciones y se ajustaron los algoritmos hasta alcanzar un grado de error mínimo.

Como consecuencia de este trabajo de tesis y con el objetivo de medir los resultados obtenidos se ha desarrollado un generador de matrices de confusión con salida opcional para LaTeX. Esta herramienta es configurable y puede mostrar una cantidad arbitraria de etiquetas de mayor error dentro de la matriz. También posee la capacidad de exhibir las palabras (y la cantidad de veces que ocurren) para cada par de etiquetas contenidas en la matriz. Ninguna de estas características fue encontrada en generadores de matrices de confusión clásicos.

Utilizar un diccionario como nueva fuente de información, convirtiéndolo en un corpus de entrenamiento para etiquetadores gramaticales aumenta levemente

el rendimiento final del etiquetado. Esto es cierto incluso para etiquetadores de distintas bases teóricas (máxima entropía y modelos ocultos de Markov). Las mejoras no logran ser significativas y aumentan tímidamente los valores del resultado final.

Esto puede suceder ya que la cantidad de información gramatical que agrega un diccionario no es tan considerable; asciende a un valor cercano al 10 % de etiquetas por palabra, es decir que de cada 100 palabras que se extraen de los ejemplos del diccionario solo 10 poseen una etiqueta gramatical.

Este trabajo de tesis deja como aporte una nueva fuente de información semántica producida a partir de Cobuild, la cual puede ser utilizada en trabajos futuros. Dicha fuente de información es pública y se encuentra disponible en COMPLETAR. También todo el código utilizado para realizar las tareas de extracción, traducción de etiquetas, etiquetado, medición y generación de matrices de confusión es público y se encuentra disponible en COMPLETAR

Capítulo 6

Apendice

6.1. Etiquetar el corpus WSJ con Stanford Tagger

La primer evaluación de este experimento consiste en entrenar el etiquetador gramatical Stanford Tagger con WSJ como corpus de entrenamiento y con WSJ + NFI. Luego se procede a etiquetar el WSJ plano (sin etiquetas gramaticales) con estos dos modelos. Por último se contruye la matriz de confusión:

Cuadro 6.1: *Matriz de confusion para*
WSJ_1 = WSJ original contra
WSJ_2 = WSJ etiquetado con MaxEnt (entrenado con WSJ)

WSJ_1 \ WSJ_2	**JJ**	**IN**	**NN**	**NNP**	**VBD**	**RB**	**VBN**	**VBP**	**RP**	**JJR**
NN	**1726**	15	-	**1132**	16	61	18	28	1	2
RB	736	**1593**	189	139	-	-	3	1	293	36
JJ	-	60	**1276**	632	51	515	762	7	-	5
VBN	**894**	-	44	25	**1052**	1	-	5	-	-
NNPS	40	-	-	**997**	-	-	-	-	-	-
IN	87	-	4	22	-	**959**	-	2	527	-
VBG	196	-	**829**	14	-	-	1	1	-	-
VBD	40	-	26	8	-	-	**806**	14	-	-
RP	4	628	-	1	-	230	-	-	-	-
VB	58	8	365	36	37	12	22	544	-	6

Aciertos: 1.236.647 (97,90 %)
Errores: 26.477 (2,10 %)

Cuadro 6.2: *Matriz de confusion para*
WSJ_1 = WSJ original contra
NFI_1 = WSJ etiquetado con MaxEnt (entrenado con WSJ + NFI)

WSJ_1 \ NFI_1	**JJ**	**IN**	**NNP**	**NN**	**VBD**	**RB**	**RP**	**VBN**	**NNPS**	**VBP**
NN	**1880**	15	**1257**	-	20	66	2	19	-	26
RB	764	**1454**	141	180	-	-	482	3	-	2
JJ	-	62	673	**1247**	47	537	2	771	-	4
VBN	**966**	-	26	46	**1104**	-	-	-	-	5
IN	107	-	19	2	-	**973**	**890**	-	-	2
NNPS	41	-	**946**	1	-	-	-	-	-	-
VBD	33	-	8	26	-	-	-	**821**	-	16
VBG	248	-	21	810	-	-	-	1	-	1
NNP	524	31	-	487	4	20	1	8	526	4
VB	64	7	29	374	38	13	-	28	-	524

Aciertos: 1.235.462 (97,81 %)
Errores: 27.662 (2,19 %)

Se puede observar que el rendimiento del etiquetador entrenado con WSJ es un poco mejor (97,9 %) que cuando es entrenado con WSJ + NFI (97,81 %). La mayoría de los errores para Stanford Tagger entrenado con WSJ se da en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP. Para Stanford Tagger entrenado con WSJ + NFI la mayoría de los errores se da en las mismas etiquetas, pero con cantidad de errores mayor, sobre todo para NN etiquetado como JJ.

La segunda evaluación de este experimento consiste en entrenar Stanford Tagger con la mitad de WSJ y con la mitad de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta la mitad restante de WSJ y se construye la matriz de confusión. Se realiza la misma operación para cada mitad:

Cuadro 6.3: *Matriz de confusion para*
WSJ_3 = 1 mitad WSJ original contra
ME_1 = 1 mitad WSJ etiquetado con MaxEnt (entrenado con 2 mitad de WSJ)

ME_1 / WSJ_3	**JJ**	**NN**	**NNP**	**IN**	**VBN**	**VBD**	**RB**	**NNS**	**VBG**	**JJR**
NN	**1558**	-	**1027**	6	22	15	38	133	403	8
JJ	-	**1309**	**606**	32	**746**	39	299	65	263	5
RB	512	104	31	**989**	2	1	-	4	1	14
NNPS	31	-	**943**	-	-	-	-	246	-	-
VBN	545	28	36	-	-	**722**	-	-	-	-
VBG	192	**614**	22	-	1	-	-	-	-	-
VBD	41	26	9	-	**604**	-	-	-	-	-
NNP	401	542	-	23	8	10	38	156	37	-
IN	72	5	26	-	1	-	489	-	2	-
RP	2	3	1	449	-	-	179	-	-	-

Aciertos: 610.045 (96,59 %)
Errores: 21.529 (3,41 %)

Cuadro 6.4: *Matriz de confusion para*
WSJ_3 = 1 mitad WSJ original contra
NFI_2 = 1 mitad WSJ etiquetado con MaxEnt (entrenado con 2 mitad de WSJ + NFI)

NFI_2 / WSJ_3	**JJ**	**NN**	**NNP**	**IN**	**VBD**	**VBN**	**RP**	**RB**	**NNS**	**VBG**
NN	**1509**	-	**1012**	6	12	12	-	38	114	391
JJ	-	**1092**	567	32	27	**617**	1	310	68	209
RB	484	80	82	**886**	1	2	238	-	1	-
NNPS	29	4	**850**	-	-	-	-	-	233	-
VBN	**574**	21	39	-	**695**	-	-	-	1	-
NNP	432	**644**	-	20	6	6	-	23	151	27
VBD	36	18	13	-	-	**615**	-	-	-	-
IN	81	5	27	-	-	-	527	467	-	2
VBG	256	509	26	-	-	1	-	-	-	-
VBZ	-	1	26	-	-	-	-	-	407	-

Aciertos: 611.063 (96,75 %)
Errores: 20.511 (3,25 %)

Cuadro 6.5: *Matriz de confusion para*
WSJ_4 = 2 mitad WSJ original contra
ME_2 = 2 mitad WSJ etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

ME_2 / WSJ_4	**JJ**	**NN**	**IN**	**NNP**	**VBD**	**RB**	**VBN**	**NNPS**	**NNS**	**VBG**
NN	**1604**	-	12	**916**	16	36	21	-	150	381
JJ	-	**1197**	45	522	37	381	483	-	46	202
RB	466	168	**944**	62	1	-	1	-	3	1
VBN	**863**	29	-	32	**779**	1	-	-	-	-
IN	50	3	-	26	-	**698**	-	-	-	2
NNPS	16	-	-	**651**	-	-	-	-	167	-
VBG	198	**572**	-	19	-	-	-	-	-	-
VBD	66	43	2	16	-	-	**570**	-	-	-
NNP	462	503	18	-	2	19	19	518	131	16
RP	3	1	426	-	-	129	-	-	-	-

Aciertos: 610.309 (96,64 %)
Errores: 21.241 (3,36 %)

Cuadro 6.6: *Matriz de confusion para*
WSJ_4 = 2 mitad WSJ original contra
NFI_3 = 2 mitad WSJ etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

NFI_3 / WSJ_4	**JJ**	**NN**	**NNP**	**IN**	**VBD**	**RB**	**VBN**	**RP**	**NNPS**	**NNS**
NN	**1564**	-	**979**	10	9	39	10	1	-	147
JJ	-	**1000**	516	39	35	383	418	5	-	54
VBN	**848**	31	27	-	**767**	-	-	-	-	1
RB	456	151	110	**830**	-	-	1	316	-	2
IN	50	2	32	-	-	**692**	-	554	-	-
NNPS	15	-	**616**	-	-	-	-	-	-	179
VBD	48	27	9	-	-	-	**615**	-	-	1
NNP	482	**589**	-	17	3	15	9	1	492	136
VBG	231	525	24	1	-	-	-	-	-	-
VBZ	-	2	8	-	-	-	-	-	2	380

Aciertos: 610.918 (96,73 %)
Errores: 20.632 (3,27 %)

Se puede apreciar una leve mejora en el porcentaje de etiquetas acertadas; 96,59 % contra 96,75 % y 96,64 % contra 96,73 % para cada mitad respectivamente. Los errores más comunes son producidos en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP por TnT y por etiquetas JJ etiquetadas como NN, para las dos mitades entrenadas tanto con WSJ como con WSJ +

NFI. Se puede notar que el porcentaje de error al etiquetar JJ cuando era NN es menor en la evaluación realizada sobre TnT entrenado con WSJ + NFI.

A continuación se presenta la diferencia entre las etiquetas generadas a partir de WSJ vs WSJ + NFI.

Cuadro 6.7: *Matriz de confusion para*
ME_1 = 1 mitad WSJ etiquetado por MaxEnt (entrenado con 2 mitad WSJ) vs
NFI_2 = 1 mitad WSJ etiquetado con MaxEnt (entrenado con 2 mitad de WSJ + NFI)

ME_1 \ NFI_2	**JJ**	**NN**	**RP**	**VBN**	**VBD**	**NNP**	**VBG**	**VB**	**RB**	**NNPS**
NN	**630**	-	2	15	11	**228**	**224**	198	40	-
JJ	-	**447**	1	135	15	155	54	41	152	2
IN	16	2	**425**	-	-	7	-	1	197	-
NNP	**226**	**353**	-	4	-	-	9	12	3	146
VBN	**290**	13	-	-	**251**	8	-	1	-	-
VBD	30	6	-	**277**	-	10	-	12	-	-
RB	112	19	181	-	-	68	-	3	-	-
VBP	15	103	-	5	11	3	-	170	2	-
VBG	166	134	-	-	-	26	-	-	1	-
VBZ	1	-	-	-	-	5	-	-	-	-

Aciertos: 623.914 (98,79 %)
Errores: 7.660 (1,21 %)

Cuadro 6.8: *Matriz de confusion para*
ME_2 = 2 mitad WSJ etiquetado por MaxEnt (entrenado con 1 mitad WSJ) vs
TnT_3 = 2 mitad WSJ etiquetado con MaxEnt (entrenado con 1 mitad de WSJ + NFI)

ME_2 \ NFI_3	**JJ**	**RP**	**NN**	**VBN**	**VBD**	**RB**	**NNP**	**VB**	**VBG**	**NNS**
NN	**564**	1	-	11	19	22	**260**	175	163	26
IN	9	**422**	3	-	2	**263**	16	1	-	1
JJ	-	-	**385**	178	32	144	157	33	60	28
VBD	31	-	12	**310**	-	1	1	7	-	-
NNP	195	-	**286**	9	4	20	-	20	2	73
VBN	**216**	-	19	-	**267**	-	13	7	-	-
RB	124	**199**	16	1	-	-	75	6	-	-
VBP	20	-	104	3	15	-	6	177	-	1
VBG	151	-	138	-	-	-	13	5	-	-
VBZ	-	-	2	-	-	-	4	-	-	143

Aciertos: 624.007 (98,81 %)
Errores: 7.543 (1,19 %)

La tercer evaluación de este experimento consiste en entrenar Stanford Tagger con un cuarto de WSJ y con un cuarto de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta los 3/4 restantes de WSJ y se construye la matriz de confusión. Se realiza la misma operación para cada uno de los cuartos:

Cuadro 6.9: *Rendimiento de Stanford Tagger entrenado con cuartos de WSJ con y sin NFI*

Evaluación	**Porcentaje de aciertos**
Stanford Tagger entrenado con el primer 1/4 de WSJ	96.30 %
Stanford Tagger entrenado con el primer 1/4 de WSJ + NFI	96.56 %
Stanford Tagger entrenado con el segundo 1/4 de WSJ	96.30 %
Stanford Tagger entrenado con el segundo 1/4 de WSJ + NFI	96.52 %
Stanford Tagger entrenado con el tercer 1/4 de WSJ	96.28 %
Stanford Tagger entrenado con el tercer 1/4 de WSJ + NFI	96.57 %
Stanford Tagger entrenado con el cuarto 1/4 de WSJ	96.25 %
Stanford Tagger entrenado con el cuarto 1/4 de WSJ + NFI	96.52 %

En todos los casos se puede apreciar una mejora en el acierto de etiquetas para el corpus de entrenamiento WSJ + NFI contra WSJ.

La cuarta evaluación de este experimento consiste en entrenar Stanford Tagger con un décimo de WSJ y con un décimo de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta los 9/10 restantes de WSJ y se presentan los resultados:

- 95.67 % de acierto de etiquetas para el etiquetado de 9/10 de WSJ con Stanford Tagger entrenado con 1/10 WSJ
- 96.21 % de acierto de etiquetas para el etiquetado de 9/10 de WSJ con Stanford Tagger entrenado con 1/10 WSJ+NFI

Se puede apreciar un aumento del porcentaje de aciertos en el corpus de entrenamiento que incorpora NFI.

6.2. Etiquetar el corpus BNC con TnT

La primer evaluación de este experimento consiste en entrenar el etiquetador gramatical TnT con WSJ como corpus de entrenamiento y con WSJ + NFI. Luego se procede a etiquetar el BNC plano (sin etiquetas gramaticales) con estos dos modelos. Por último se construye la matriz de confusión:

Cuadro 6.10: *Matriz de confusion para*
BNC_1 = BNC original contra
BNC_2 = BNC etiquetado con TnT (entrenado con WSJ)

BNC_1 \ BNC_2	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**NNPS**	**VBD**	**RB**	**VBG**
NN	**26529**	**5641**	-	140	701	4	19	153	353	1467
JJ	**8562**	-	**2459**	**3754**	49	2	23	338	1091	1913
DT	74	**7773**	215	-	1	-	-	-	914	-
RB	1034	2021	**4350**	9	269	6	1	7	-	41
NN	472	651	-	7	**3319**	-	17	2	8	4
IN	212	253	665	19	85	**2903**	6	43	1178	14
NNS	2355	82	686	1	-	-	**2688**	-	5	-
VBN	50	385	96	-	-	2	-	2390	1	7
RP	23	7	142	-	-	1	-	-	2087	-
VBD	53	227	92	2085	-	-	-	-	6	17

Aciertos: 1.848.844 (92,46 %)
Errores: 150.872 (7,54 %)

Cuadro 6.11: *Matriz de confusion para*
BNC_1 = BNC original contra
NFI_1 = BNC etiquetado con TnT (entrenado con WSJ + NFI)

BNC_1 \ NFI_1	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**VBD**	**NNPS**	**CD**	**VBG**
NN	**25869**	**4786**	-	104	654	3	136	9	59	1402
JJ	**8364**	-	2017	**3471**	68	1	326	18	19	1711
DT	73	**7684**	216	-	1	-	-	-	-	-
RB	1054	2091	**4150**	18	234	3	5	-	107	42
NN	447	648	-	10	**3302**	-	1	14	1896	5
IN	219	151	890	19	90	**2903**	46	5	-	13
VBN	50	436	85	-	3	-	**2635**	-	-	8
NNS	2334	68	827	-	-	-	-	**2524**	47	-
VBD	53	208	69	2086	-	-	-	-	-	8
IN	730	1707	1418	53	490	-	2	2	-	466

Aciertos: 1.854.333 (92,73 %)
Errores: 145.383 (7,27 %)

Se puede observar que el rendimiento del etiquetador TnT entrenado con WSJ+NFI es un poco mejor (92,73 %) que el rendimiento de TnT entrenado con WSJ (92,46 %). La mayoría de los errores para TnT entrenado con WSJ se da en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP por TnT. Para TnT entrenado con WSJ + NFI la mayoría de los errores se da

en las mismas etiquetas, pero con cantidad de errores mayor, sobre todo para NN etiquetado como NNP.

La segunda evaluación de este experimento consiste en entrenar TnT con la mitad de WSJ y con la mitad de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta BNC y se construye la matriz de confusión. Se realiza la misma operación para cada mitad:

Cuadro 6.12: *Matriz de confusion para*
BNC_1 = BNC original contra
TnT_1 = BNC etiquetado con TnT (entrenado con 2 mitad de WSJ)

TnT_1 / BNC_1	**NNP**	**JJ**	**VBN**	**NN**	**NNS**	**WRB**	**VBD**	**RB**	**VBG**	**NNPS**
NN	**27195**	**6142**	180	-	804	5	191	396	1735	9
JJ	**8813**	-	**4421**	**3008**	69	1	371	1106	2191	22
DT	82	**7802**	-	200	1	-	-	903	-	-
RB	1008	2145	15	**4196**	314	5	6	-	41	2
NN	483	650	8	-	**3410**	-	1	9	4	19
NNS	**2962**	76	1	734	-	-	-	19	-	2174
IN	202	314	27	566	86	**2904**	45	1152	15	-
VBN	44	432	-	106	-	-	2707	1	7	-
VBD	45	258	2390	122	-	-	-	7	20	-
RP	18	7	-	144	-	-	-	2335	-	-

Aciertos: 1.841.617 (92,09 %)
Errores: 158.099 (7,91 %)

Cuadro 6.13: *Matriz de confusion para*
BNC_1 = BNC original contra
NFI_2 = BNC etiquetado con TnT (entrenado con 2 mitad de WSJ+NFI)

NFI_2 / BNC_1	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**VBD**	**NNPS**	**CD**	**VBG**
NN	**26040**	**4798**	-	107	656	1	132	12	63	1461
JJ	**8499**	-	2040	**3616**	78	1	341	17	14	1775
DT	82	**7637**	211	-	1	-	-	-	-	-
RB	1036	1956	**4372**	15	223	2	6	-	106	42
NN	452	577	-	13	**3346**	-	-	15	1908	5
IN	232	147	924	14	92	**2903**	50	-	-	13
VBN	43	473	92	-	3	-	**2716**	-	-	8
NNS	**2691**	62	889	-	-	-	-	2251	54	-
VBD	48	218	72	2171	-	-	-	-	-	8
IN	5	-	1	-	-	-	-	-	-	-

Aciertos: 1.852.716 (92,65 %)
Errores: 147.000 (7,35 %)

Cuadro 6.14: *Matriz de confusion para*
BNC_1 = BNC original contra
TnT_2 = BNC etiquetado con TnT (entrenado con 1 mitad de WSJ)

BNC_1 \ TnT_2	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**NNPS**	**VBD**	**VBG**	**RB**
NN	**26287**	**6528**	-	187	783	3	36	169	1614	435
JJ	**8503**	-	**2921**	**3713**	64	1	28	476	2027	1238
DT	84	**7763**	232	-	1	-	-	-	-	976
RB	1157	2031	**4455**	30	515	1	1	8	42	-
NN	478	796	-	6	**3225**	-	21	2	14	4
IN	193	235	626	24	77	**2903**	6	51	14	1165
NNS	2504	87	863	2	-	-	**2714**	-	-	1
VBN	57	615	95	-	-	-	-	2581	13	5
IN	733	1415	2260	57	499	3	2	-	467	614
VBD	67	317	99	2126	1	-	-	-	27	5

Aciertos: 1.842.527 (92,14 %)
Errores: 157.189 (7,86 %)

Cuadro 6.15: *Matriz de confusion para*
BNC_1 = BNC original contra
NFI_3 = BNC etiquetado con TnT (entrenado con 1 mitad de WSJ+NFI)

BNC_1 \ NFI_3	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**VBD**	**NNPS**	**CD**	**WDT**
NN	**25580**	**4844**	-	110	679	3	136	19	63	1
JJ	**8304**	-	2079	**3397**	68	1	357	21	21	1
DT	79	**7643**	225	-	1	-	-	-	-	656
RB	1180	2054	**4224**	20	224	-	6	1	122	4
NN	444	706	-	11	**3228**	-	-	20	1867	-
IN	224	145	894	18	86	**2903**	49	2	-	97
VBN	55	502	79	-	4	-	**2755**	-	-	-
NNS	2380	67	876	-	-	-	-	**2533**	24	-
VBD	58	230	68	2128	-	-	-	-	-	-
IN	728	1818	1893	48	486	3	-	2	-	1

Aciertos: 1.853.464 (92,69 %)
Errores: 146.252 (7,31 %)

Se puede apreciar una leve mejoría en el porcentaje de etiquetas acertadas para el modelo que incorpora NFI; 92,09 % contra 92,65 % y 92,14 % contra 92,69 % para cada mitad respectivamente. Los errores más comunes son producidos en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP por TnT, para las dos mitades entrenadas tanto con WSJ como con WSJ + NFI. Se puede notar que el porcentaje de error al etiquetar JJ cuando era NN es menor en la evaluación realizada sobre TnT entrenado con el modelo que incorpora NFI.

A continuación se presenta la diferencia entre las etiquetas generadas a partir de WSJ vs WSJ + NFI.

Cuadro 6.16: *Matriz de confusion para*
TnT_3 = BNC etiquetado por TnT (entrenado con 1 mitad WSJ) vs
NFI_3 = BNC etiquetado con TnT (entrenado con 1 mitad de WSJ + NFI)

TnT_3 \ NFI_3	**NN**	**JJ**	**VBD**	**VBN**	**VB**	**NNS**	**NNP**	**VBG**	**RB**	**RP**
JJ	**4538**	-	296	1011	393	107	1000	507	792	2
NN	-	**3643**	50	59	**1617**	307	**1440**	1054	509	1
VBN	140	1304	**2210**	-	71	8	50	5	10	-
NNP	**2154**	1088	23	32	268	524	-	89	105	-
VB	**1812**	289	95	138	-	-	159	10	94	1
VBD	88	308	-	**1776**	75	-	41	1	9	-
VBZ	40	37	3	-	6	**1444**	24	-	17	-
VBP	682	151	39	27	**1313**	2	26	1	21	-
RB	437	1214	3	4	123	3	225	-	-	767
VBG	810	914	28	4	19	6	113	-	2	-

Aciertos: 1.948.355 (97,43 %)
Errores: 51.363 (2,57 %)

Cuadro 6.17: *Matriz de confusion para*
TnT_2 = BNC etiquetado por TnT (entrenado con 2 mitad WSJ) vs
NFI_2 = BNC etiquetado con TnT (entrenado con 2 mitad de WSJ + NFI)

TnT_2 \ NFI_2	**NN**	**JJ**	**VBD**	**VB**	**NNS**	**VBN**	**NNP**	**RP**	**VBG**	**VBP**
JJ	**4332**	-	277	314	88	824	900	3	352	56
NN	-	**3805**	76	**1657**	135	58	**1451**	-	895	204
NNP	**2679**	1173	10	203	611	16	-	-	89	19
VBN	147	**1864**	**2131**	46	7	-	19	-	-	5
VB	**2060**	354	197	-	2	113	105	1	6	883
VBZ	25	25	2	6	**1597**	-	15	-	-	5
VBD	109	240	-	34	1	**1586**	35	-	2	28
RB	416	1255	1	133	24	-	183	948	-	9
VBP	591	62	75	1224	1	13	18	-	1	-
VBG	991	914	14	4	5	1	86	-	-	-

Aciertos: 1.947.903 (97,41 %)
Errores: 51.815 (2,59 %)

La tercer evaluación de este experimento consiste en entrenar TnT con un cuarto de WSJ y con un cuarto de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta los 3/4 restantes de WSJ y se construye la matriz de confusión. Se realiza la misma operación para cada uno de los cuartos:

Cuadro 6.18: *Rendimiento de TnT entrenado con cuartos de WSJ con y sin NFI*

Evaluación	**Porcentaje de aciertos**
TnT entrenado con el primer 1/4 de WSJ	91.75 %
TnT entrenado con el primer 1/4 de WSJ + NFI	92.59 %
TnT entrenado con el segundo 1/4 de WSJ	91.74 %
TnT entrenado con el segundo 1/4 de WSJ + NFI	92.6 %
TnT entrenado con el tercer 1/4 de WSJ	91.64 %
TnT entrenado con el tercer 1/4 de WSJ + NFI	92.6 %
TnT entrenado con el cuarto 1/4 de WSJ	91.64 %
TnT entrenado con el cuarto 1/4 de WSJ + NFI	92.55 %

En todos los casos se puede apreciar una mejora en el acierto de etiquetas para el modelo que incorpora NFI.

La cuarta evaluación de este experimento consiste en entrenar TnT con un décimo de WSJ y con un décimo de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta BNC y se presentan los resultados:

- 90.9 % de acierto de etiquetas para el etiquetado de BNC con TnT entrenado con 1/10 WSJ

- 92.49 % de acierto de etiquetas para el etiquetado de BNC con TnT entrenado con 1/10 WSJ+NFI

Se puede apreciar un aumento del porcentaje de aciertos en el modelo que incorpora NFI.

6.3. Etiquetar el corpus BNC con Stanford Tagger

La primer evaluación de este experimento consiste en entrenar el etiquetador gramatical Stanford Tagger con WSJ como corpus de entrenamiento y con WSJ + NFI. Luego se procede a etiquetar el BNC plano (sin etiquetas gramaticales) con estos dos modelos. Por último se contruye la matriz de confusión:

Cuadro 6.19: *Matriz de confusion para*
BNC_1 = BNC original contra
BNC_2 = BNC etiquetado con MaxEnt (entrenado con WSJ)

BNC_1 \ BNC_2	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**RB**	**CD**	**VBG**	**NNPS**
NN	**26141**	**4045**	-	115	533	-	253	16	1143	7
JJ	**8675**	-	**2860**	**3276**	30	-	1033	3	2054	12
DT	119	**8132**	192	-	5	-	443	-	-	-
RB	982	2314	**3753**	159	236	-	-	160	85	4
NN	567	1115	-	19	**3132**	-	10	2605	2	5
NNS	**2982**	90	750	-	-	-	6	47	-	1959
IN	60	334	247	57	104	**2901**	522	1	35	2
RP	43	17	137	-	-	-	2691	-	-	-
FW	1754	255	540	9	199	-	2	426	1	18
IN	-	1	-	-	-	-	-	-	-	-

Aciertos: 1.856.979 (92,86 %)
Errores: 142.739 (7,14 %)

Cuadro 6.20: *Matriz de confusion para*
BNC_1 = BNC original contra
NFI_1 = BNC etiquetado con MaxEnt (entrenado con WSJ + NFI)

BNC_1 \ NFI_1	**NNP**	**JJ**	**NN**	**NNS**	**VBN**	**WRB**	**CD**	**RB**	**VBG**	**NNPS**
NN	**25675**	**3975**	-	542	95	-	27	231	1185	4
JJ	**8397**	-	2258	57	**3146**	-	6	896	1894	10
DT	113	**7880**	194	2	-	-	-	666	-	-
RB	823	2326	**3834**	228	100	-	140	-	89	1
NN	506	1167	-	**3148**	11	-	**2520**	3	5	6
IN	106	232	331	151	34	**2901**	-	718	37	1
NNS	**2778**	82	919	-	1	-	36	6	-	1881
RP	42	8	139	-	-	-	-	2106	-	-
FW	1774	320	491	218	3	-	396	1	3	13
VBN	88	523	87	1	-	-	-	-	7	-

Aciertos: 1.860.683 (93,05 %)
Errores: 139.035 (6,95 %)

Se puede observar que el rendimiento del etiquetador entrenado con WSJ+NFI (93,05 %) es un poco mejor que cuando es entrenado con WSJ (92,86 %). La mayoría de los errores para Stanford Tagger entrenado con WSJ se da en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP. Para Stanford Tagger entrenado con WSJ + NFI la mayoría de los errores se da en las mismas etiquetas, pero con cantidad de errores mayor, sobre todo para NN etiquetado como JJ.

La segunda evaluación de este experimento consiste en entrenar Stanford Tagger con la mitad de WSJ y con la mitad de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta BNC y se construye la matriz de confusión. Se realiza la misma operación para cada mitad:

Cuadro 6.21: *Matriz de confusion para*
BNC_1 = BNC original contra
ME_1 = BNC etiquetado con MaxEnt (entrenado con 2 mitad de WSJ)

BNC_1 \ ME_1	**NNP**	**JJ**	**NN**	**NNS**	**VBN**	**WRB**	**RB**	**CD**	**VBG**	**NNPS**
NN	**26061**	**4689**	-	620	97	-	248	6	1296	8
JJ	**8570**	-	**3574**	49	**3001**	-	1100	6	2190	10
DT	133	**8068**	224	1	-	-	478	1	-	-
RB	985	2428	**3611**	475	143	-	-	164	77	9
NN	554	1404	-	**3128**	7	-	14	2633	7	6
NNS	**2984**	146	855	-	-	-	4	46	-	2054
IN	98	210	237	146	34	**2901**	568	3	21	2
RP	47	6	152	-	-	-	2793	-	-	-
FW	1763	268	454	212	3	-	3	518	1	20
IN	-	1	-	-	-	-	-	-	-	-

Aciertos: 1.851.792 (92,60 %)
Errores: 147.926 (7,40 %)

Cuadro 6.22: *Matriz de confusion para*
BNC_1 = BNC original contra
NFI_2 = BNC etiquetado con MaxEnt (entrenado con 2 mitad de WSJ+NFI)

BNC_1 \ NFI_2	**NNP**	**JJ**	**NN**	**NNS**	**VBN**	**WRB**	**CD**	**RB**	**NNPS**	**VBG**
NN	**25412**	**4306**	-	551	94	-	21	230	5	1193
JJ	**8191**	-	2201	65	**2964**	-	7	931	9	1774
DT	108	**7785**	216	1	-	-	-	754	-	-
RB	823	2471	**3715**	270	109	-	149	-	-	66
NN	492	1306	-	**3125**	5	-	**2500**	3	7	4
IN	125	250	296	155	41	**2901**	1	792	3	19
NNS	**2685**	78	937	-	1	-	30	9	1889	-
RP	50	5	144	-	-	-	-	2019	-	-
FW	1762	381	481	220	2	-	373	3	14	2
IN	524	1638	619	728	105	-	14	662	1	553

Aciertos: 1.860.014 (93,01 %)
Errores: 139.704 (6,99 %)

Cuadro 6.23: *Matriz de confusion para*
BNC_1 = BNC original contra
ME_2 = BNC etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

BNC_1 \ ME_2	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**RB**	**CD**	**VBG**	**VBD**
NN	**27101**	**4519**	-	146	550	-	241	7	1369	116
JJ	**9043**	-	**3838**	**3837**	43	-	1025	3	2238	296
DT	128	**8324**	185	-	1	-	461	-	-	-
RB	1071	2583	**3445**	141	311	-	-	170	71	78
NNS	**3415**	83	885	-	-	-	7	41	-	1
NN	573	955	-	24	**3189**	-	5	2732	3	8
IN	82	412	267	54	106	**2901**	400	1	23	63
RP	21	26	140	-	-	-	2859	-	-	-
VBN	85	464	126	-	1	-	1	-	1	1986
FW	1757	181	509	9	242	-	3	442	1	5

Aciertos: 1.848.799 (92,45 %)
Errores: 150.919 (7,55 %)

Cuadro 6.24: *Matriz de confusion para*
BNC_1 = BNC original contra
NFI_3 = BNC etiquetado con MaxEnt (entrenado con 1 mitad de WSJ+NFI)

BNC_1 \ NFI_3	**NNP**	**JJ**	**NN**	**VBN**	**NNS**	**WRB**	**CD**	**RB**	**VBD**	**VBG**
NN	**26026**	**4058**	-	92	528	-	24	228	99	1259
JJ	**8418**	-	2221	**3338**	68	-	5	876	217	1831
DT	117	**7825**	195	-	1	-	-	741	-	-
RB	1083	2502	**3839**	88	230	-	124	-	37	88
NN	518	1115	-	13	**3154**	-	**2522**	7	3	4
NNS	**2932**	79	967	1	-	-	37	7	-	-
IN	170	288	297	33	159	**2901**	2	702	48	34
RP	37	8	140	-	-	-	-	2073	-	-
VBN	89	567	97	-	1	-	-	-	1855	7
FW	1739	275	470	3	257	-	395	2	2	3

Aciertos: 1.858.602 (92,94 %)
Errores: 141.116 (7,06 %)

Se puede apreciar una leve mejoría en el porcentaje de etiquetas acertadas; 92,6 % contra 93,01 % y 92,45 % contra 92,94 % para cada modelo respectivamente. Los errores más comunes son producidos en etiquetas NN del gold standard cuando son etiquetadas como JJ y NNP por , para las dos mitades entrenadas tanto con WSJ como con WSJ + NFI. Se puede notar que el porcentaje de

error al etiquetar JJ cuando era NN es menor en la evaluación realizada sobre Stanford Tagger entrenado con WSJ + NFI.

A continuación se presentan las matrices de confusión para BNC etiquetado con Stanford Tagger entrenado con la mitad de WSJ con y sin NFI.

Cuadro 6.25: *Matriz de confusion para*
ME_2 = BNC etiquetado por MaxEnt (entrenado con 2 mitad WSJ) vs
NFI_2 = BNC etiquetado con MaxEnt (entrenado con 2 mitad de WSJ + NFI)

ME_2 \ NFI_2	**JJ**	**NN**	**NNP**	**VBN**	**RP**	**NNS**	**VBG**	**VB**	**VBD**	**RB**
NN	**4784**	-	**2095**	120	12	266	**1525**	**1459**	170	425
JJ	-	**3438**	1138	1243	2	113	339	303	199	1006
NNP	**1658**	**3312**	-	54	-	807	115	307	60	224
VBD	299	71	20	**1732**	-	-	5	145	-	7
IN	157	161	196	8	**1675**	16	49	365	124	1010
VBZ	22	29	45	1	-	**1633**	-	6	8	19
VBN	1279	81	58	-	-	16	7	61	1399	27
VBP	173	1030	22	21	-	10	8	1342	109	38
VB	283	1090	271	114	-	6	13	-	64	122
RB	989	526	226	16	943	14	15	104	26	-

Aciertos: 1.941.724 (97,10 %)
Errores: 57.994 (2,90 %)

Cuadro 6.26: *Matriz de confusion para*
ME_3 = BNC etiquetado por MaxEnt (entrenado con 1 mitad WSJ) vs
NFI_3 = BNC etiquetado con MaxEnt (entrenado con 1 mitad de WSJ + NFI)

ME_3 \ NFI_3	**JJ**	**NN**	**NNP**	**NNS**	**VBN**	**RP**	**VB**	**VBD**	**VBG**	**RB**
NN	**4863**	-	**2042**	153	97	12	**1526**	101	1390	266
JJ	-	**3385**	1069	65	972	5	304	117	284	981
NNP	**1962**	**3286**	-	859	35	-	279	17	119	142
VBZ	18	38	48	**1885**	-	-	26	37	1	25
VBN	**1851**	127	56	5	-	-	59	1401	8	31
VBD	362	143	34	2	**1781**	-	187	-	17	12
IN	131	121	236	51	2	**1614**	221	11	21	966
VB	275	1210	265	15	124	-	-	77	16	71
VBP	229	1015	31	9	23	-	1126	132	9	46
RB	1016	603	275	17	6	971	125	18	14	-

Aciertos: 1.941.977 (97,11 %)
Errores: 57.741 (2,89 %)

La tercer evaluación de este experimento consiste en entrenar Stanford Tagger con un cuarto de WSJ y con un cuarto de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta BNC y se construye la matriz de confusión. Se realiza la misma operación para cada uno de los cuartos:

Cuadro 6.27: *Rendimiento de Stanford Tagger entrenado con cuartos de WSJ con y sin NFI*

Evaluación	**Porcentaje de aciertos**
Stanford Tagger entrenado con el primer 1/4 de WSJ	92.09 %
Stanford Tagger entrenado con el primer 1/4 de WSJ + NFI	92.96 %
Stanford Tagger entrenado con el segundo 1/4 de WSJ	92.10 %
Stanford Tagger entrenado con el segundo 1/4 de WSJ + NFI	92.96 %
Stanford Tagger entrenado con el tercer 1/4 de WSJ	92.14 %
Stanford Tagger entrenado con el tercer 1/4 de WSJ + NFI	92.92 %
Stanford Tagger entrenado con el cuarto 1/4 de WSJ	91.98 %
Stanford Tagger entrenado con el cuarto 1/4 de WSJ + NFI	92.87 %

En todos los casos se puede apreciar una mejora en el acierto de etiquetas para el corpus de entrenamiento WSJ + NFI contra WSJ.

La cuarta evaluación de este experimento consiste en entrenar Stanford Tagger con un décimo de WSJ y con un décimo de WSJ + NFI. Posteriormente con estos dos modelos se etiqueta BNC y se presentan los resultados:

- 91.25 % de acierto de etiquetas para el etiquetado de BNC con Stanford Tagger entrenado con 1/10 WSJ
- 92.81 % de acierto de etiquetas para el etiquetado de BNC con Stanford Tagger entrenado con 1/10 WSJ+NFI

Se puede apreciar un aumento del porcentaje de aciertos en el corpus de entrenamiento que incorpora NFI.

Bibliografía

[1] Jurafsky, D. & Martin, J. H., Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition, Second edition, chapter 5, New Jersey: Prentice Hall.

[2] Chris Manning and Hinrich Schütze, Foundations of Statistical Natural Language Processing, MIT Press. Cambridge, MA: May 1999

[3] Brants, T. (2000). TnT: a statistical part-of-speech tagger. In Proceedings of the sixth conference on Applied natural language processing, p.224-231, April 29-May 04, 2000, Seattle, Washington. Morgan Kaufmann Publishers Inc.

[4] Kristina Toutanova and Christopher D. Manning. 2000. Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000), pp. 63-70.

[5] Mitchell P. Marcus , Mary Ann Marcinkiewicz , Beatrice Santorini, Building a large annotated corpus of English: the penn treebank, Computational Linguistics, v.19 n.2, June 1993

[6] Stevenson M., A corpus-based approach to deriving lexical mappings, EACL '99 Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics, Pages 285-286

[7] Baayen, H. and Sproat, R. (1996). Estimating lexical priors for low-frequency morphologically ambiguous forms. Computational Linguistics, 22(2), 155-166.

[8] Dermatas, E. and Kokkinakis, G. (1995). Automatic stochastic tagging of natural language texts. Computational Linguistics, 21(2), 137-164

[9] Weischedel, R., Meteer, M., Schwartz, R., Ramshaw, L. A., and Palmucci, J. (1993). Coping with ambiguity and unknown words through probabilistic models. Computational Linguistics, 19(2), 359-382.

[10] Samuelsson, C. (1993). Morphological tagging based entirely on Bayesian inference. In 9th Nordic Conference on Computational Linguistics NODALIDA-93. Stockholm.

[11] Kupiec, J. (1992). Robust part-of-speech tagging using a hidden Markov model. Computer Speech and Language, 6, 225-242.

[12] Franz, A. (1996). Automatic Ambiguity Resolution in Natural Language Processing. Springer-Verlag, Berlin.

[13] Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn treebank. Computational Linguistics, 19(2), 313-33

[14] Ratnaparkhi, A. (1996). A maximum entropy part-of-speech tagger. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, University of Pennsylvania, pp. 133-142. ACL

[15] Voutilainen, A. (1995). Morphological disambiguation. In Karlsson, F., Voutilainen, A., Heikkilä, J., and Anttila, A.(Eds.), Constraint Grammar: A Language Independent System forParsing Unrestricted Text, pp. 165-284. Mouton deGruyter, Berlin.

[16] Gale, W. A., Church, K. W., and Yarowsky, D. (1992). Estimating upper and lower bounds on the performance of word-sense disambiguation programs. In Proceedings of the 30th ACL, Newark, DE, pp. 249-256. ACL.

[17] Brown K. (Editor) 2005. Encyclopedia of Language and Linguistics - 2nd Edition. Oxford: Elsevier.

[18] Sinclair, J. 'The automatic analysis of corpora', in Svartvik, J. (ed.) Directions in Corpus Linguistics (Proceedings of Nobel Symposium 82). Berlin: Mouton de Gruyter. 1992.

[19] Wallis, S. 'Annotation, Retrieval and Experimentation', in Meurman-Solin, A. & Nurmi, A.A. (ed.) Annotating Variation and Change. Helsinki: Varieng, [University of Helsinki]. 2007

[20] Guy Aston and Lou Burnard, 1998. The BNC Handbook: Exploring the British National Corpus with SARA, Edinburgh University Press

[21] Quirk, R. 'Towards a description of English Usage', Transactions of the Philological Society. 1960. 40-61.

[22] Sankoff, D. & Sankoff, G. Sample survey methods and computer-assisted analysis in the study of grammatical variation. In Darnell R. (ed.) Canadian Languages in their Social Context Edmonton: Linguistic Research Incorporated. 1973. 7-64.

[23] Poplack, S. The care and handling of a mega-corpus. In Fasold, R. & Schiffrin D. (eds.) Language Change and Variation, Amsterdam: Benjamins. 1989. 411-451.

[24] Andersen, Francis I.; Forbes, A. Dean (2003), "Hebrew Grammar Visualized: I. Syntax", Ancient Near Eastern Studies 40: 43-61

[25] Eyland, E. Ann (1987), Revelations from Word Counts", in Newing, Edward G.; Conrad, Edgar W., Perspectives on Language and Text: Essays and Poems in Honor of Francis I. Andersen's Sixtieth Birthday, July 28, 1985, Winona Lake, IN: Eisenbrauns, p. 51, ISBN 0-931464-26-9

[26] Dukes, K., Atwell, E. and Habash, N. 'Supervised Collaboration for Syntactic Annotation of Quranic Arabic'. Language Resources and Evaluation Journal. 2011.

[27] Wallis, S. and Nelson G. 'Knowledge discovery in grammatically analysed corpora'. Data Mining and Knowledge Discovery, 5: 307-340. 2001.