



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, 2012

# Índice general

<b>1. Introducción</b>	<b>4</b>
1.1. Motivación . . . . .	5
1.2. Trabajo realizado . . . . .	8
1.3. Etiquetado gramatical . . . . .	9
<b>2. Definiciones y marco teórico</b>	<b>10</b>
2.1. Etiquetas . . . . .	11
2.2. Conjuntos de etiquetas . . . . .	15
2.2.1. Especificidad de etiquetas: Treebank vs C5 y Brown . . .	17
2.3. Corpus . . . . .	19
2.3.1. Un poco de historia . . . . .	20
2.3.2. Métodos . . . . .	20
2.4. Etiquetadores gramaticales automáticos . . . . .	22
2.4.1. Etiquetadores gramaticales basados en reglas . . . . .	23
2.4.2. Etiquetadores gramaticales de aprendizaje automático . .	23
2.5. Corpora de entrenamiento y corpora de verificación . . . . .	28
2.6. Evaluación de etiquetadores gramaticales . . . . .	29
2.7. Análisis de error . . . . .	30
2.8. Palabras desconocidas . . . . .	31
2.9. Etiquetador Gramatical TnT . . . . .	33
2.9.1. Modelo teórico . . . . .	33
2.9.2. Suavizado . . . . .	33
2.9.3. Manejo de palabras desconocidas . . . . .	34
2.10. Etiquetador Gramatical Stanford Tagger . . . . .	36
2.11. Diccionario Cobuild . . . . .	38
2.11.1. Características . . . . .	39
2.11.2. Método de construcción . . . . .	39
2.11.3. Evidencia . . . . .	40
2.11.4. Un corpus . . . . .	40
2.11.5. The Bank of English . . . . .	40
2.11.6. La lista de palabras principales . . . . .	41
2.11.7. Ejemplos . . . . .	41
2.11.8. Información gramatical . . . . .	41
2.11.9. Pragmatismo . . . . .	41
2.11.10. Definición del estilo . . . . .	41
2.12. Corpus BNC . . . . .	43
2.13. Corpus WSJ . . . . .	44

<b>3. Desarrollo</b>	<b>45</b>
3.1. Extracción de la información . . . . .	46
3.1.1. Reconocimiento de formas flexionadas . . . . .	47
3.1.2. Preprocesamiento . . . . .	48
3.2. Traducción de etiquetas . . . . .	50
3.3. Nuevo Corpus generado . . . . .	53
<b>4. Experimentación</b>	<b>55</b>
4.1. Primer experimento . . . . .	56
4.2. Segundo experimento: Etiquetar el corpus WSJ . . . . .	58
4.2.1. Etiquetar el corpus WSJ con TnT . . . . .	58
4.2.2. Etiquetar el corpus WSJ con Stanford Tagger . . . . .	64
4.3. Tercer experimento: Etiquetar el corpus BNC . . . . .	65
4.3.1. Etiquetar el corpus BNC con TnT . . . . .	65
4.3.2. Etiquetar el corpus BNC con Stanford Tagger . . . . .	66
<b>5. Conclusiones</b>	<b>67</b>
<b>6. Apendice</b>	<b>68</b>
6.0.3. Etiquetar el corpus WSJ con Stanford Tagger . . . . .	68
6.0.4. Etiquetar el corpus BNC con TnT . . . . .	73
6.0.5. Etiquetar el corpus BNC con Stanford Tagger . . . . .	77

## Capítulo 1

# Introducción

## 1.1. Motivación

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

(1) *There is no asbestos in our products now.*

El resultado de etiquetarla gramaticalmente es:

(2) *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.

A simple vista el etiquetado gramatical parece una tarea trivial o al menos sencilla, sin embargo no es así. La complejidad de este proceso reside en la ambigüedad gramatical inherente al lenguaje. Por ejemplo, la palabra *premio* puede funcionar como sustantivo:

(1) *Gané un premio*

o como verbo

(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). Ahora bien, sería interesante conocer que factor indica cual es la etiqueta correspondiente a una palabra ambigua. En (1) se puede deducir que *premio* es sustantivo porque está precedido por la palabra *un* mientras que en (2) *premio* está precedido por la palabra *te* y a partir de este hecho se puede deducir que en este caso *premio* funciona como verbo. En fin, las palabras circundantes brindan información vital para deducir el sentido gramatical en palabras ambiguas.

Como se menciona más adelante, el etiquetado gramatical juega un papel importante en áreas de la lingüística computacional como síntesis del habla, reconocimiento del habla y recuperación de la información. El etiquetado gramatical es realizado manualmente por lingüistas; especialistas en el lenguaje que se ocupan de determinar una etiqueta gramatical para cada palabra. Desde luego que también es realizado automáticamente por computadoras, mediante programas conocidos como etiquetadores gramaticales. Algunas implementaciones actuales de estos programas están basadas en el aprendizaje; toman un corpus <sup>1</sup> anotado correctamente con el cual se entrenan y luego emplean el conocimiento adquirido para etiquetar el corpus objetivo. En esa primer etapa conocida

---

<sup>1</sup>Colección de textos escritos y/o transcripciones del lenguaje oral para cierto idioma

como entrenamiento, el etiquetador gramatical obtiene, procesa y retiene información sobre cada palabra, su etiqueta asignada y su contexto. Posteriormente el etiquetador determina una etiqueta para cada palabra del corpus objetivo, analizando su ubicación y contexto y utilizando el conocimiento adquirido en la etapa previa.

Los resultados dependen en gran medida de la calidad de los datos de entrenamiento, medida en función de la cantidad de los mismos y su representatividad del lenguaje sobre el dominio abordado. Uno de los grandes problemas reside en la falta de corpus anotados para utilizar como datos de entrenamiento. Generalmente los corpora de entrenamiento son etiquetados manualmente por equipos de lingüistas especializados. Es un trabajo profundamente meticuloso y tedioso ya que el lingüista debe dar una etiqueta gramatical palabra por palabra en corpus del orden del millón de palabras. Además de la laboriosidad del trabajo, el tiempo empleado para etiquetar un corpus es sumamente extenso y como consecuencia el valor económico es significativo, ya que intervienen grupos de trabajo altamente capacitados durante períodos prolongados. El resultado de este complejo proceso es una tabla de palabras con su correspondiente etiqueta gramatical, como se muestra a continuación:

A	DT
form	NN
of	IN
asbestos	NN
once	RB
used	VCN
to	TO
make	VB
Kent	NNP
cigarette	NN
filters	NNS
has	VBZ
caused	VCN
a	DT
high	JJ
percentage	NN
of	IN
cancer	NN
deaths	NNS
among	IN
a	DT
group	NN
of	IN
workers	NNS
exposed	VCN
to	TO
it	PRP
more	RBR
than	IN
30	CD
years	NNS
ago	RB
,	,
researchers	NNS
reported	VBD
.	.

Ante la importancia que adquieren los corpora etiquetados es inevitable pensar en algún otro tipo de texto que posea información de etiquetas. Por ejemplo algunos diccionarios contienen una palabra, su definición y algunos ejemplos en donde ésta aparece con cada uno de sus sentidos. Es decir que de alguna manera un diccionario contiene por cada palabra uno o más contextos en donde ésta aparece etiquetada. Entonces si tomamos todos los ejemplos de cada palabra de un diccionario y su etiqueta podemos construir un corpus parcialmente anotado. Esta es la idea central de este trabajo.

## 1.2. Trabajo realizado

Como se mencionó en la sección anterior, la idea de este trabajo es suplir la falta de corpus de entrenamiento utilizando la información de etiquetado que posee un diccionario, generando una nueva fuente de información que servirá para entrenar etiquetadores automáticos. Este trabajo menciona detalladamente la forma de extraer la información relevante sobre etiquetas gramaticales a partir de un diccionario y las decisiones que fueron aplicadas. Esta nueva fuente de información se utiliza para entrenar etiquetadores gramaticales automáticos. Una vez entrenados dichos etiquetadores se emplean para etiquetar corpora clásicos y luego se analiza el resultado obtenido. Por último se realizan mediciones sobre el rendimiento de los etiquetadores gramaticales entrenados con esta nueva fuente de información y con los corpora clásicos de entrenamiento y se presentan las conclusiones.



### 1.3. Etiquetado gramatical

Como se mencionó anteriormente, el etiquetado gramatical es el proceso de asignar una etiqueta a cada una de las palabras de un texto según su categoría léxica. O expresado de otra manera: dada una secuencia de palabras como entrada, obtener como salida una secuencia de etiquetas para las mismas. Este proceso se realiza en base a la definición de la palabra y al contexto en que ésta aparece.

Por ejemplo en:

*Does that flight serve dinner*

*dinner* es un sustantivo y por lo tanto recibe la etiqueta para sustantivos NN.

El etiquetado gramatical brinda una gran cantidad de información sobre una palabra y sus vecinas. Por ejemplo, las etiquetas distinguen entre pronombres posesivos (mi, tu, su, etc.) y pronombres personales (Yo, Tú, Él, etc.). Saber si una palabra es un pronombre posesivo o personal nos brinda información sobre las palabras que pueden ocurrir a continuación: los pronombres posesivos generalmente son sucedidos por un sustantivo (como en *Mi comida*) mientras que los personales son sucedidos por un verbo (como en *Yo duermo*).

Utilizando esta deducción podemos aseverar que si una palabra fué etiquetada como pronombre personal, es muy probable que la próxima palabra sea un verbo. Una etiqueta gramatical también puede ofrecer información relacionada con la pronunciación de la palabra. 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) o 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.

Por último, la asignación automática de etiquetas gramaticales juega un papel importante en algoritmos de desambiguación del sentido de la palabra y en modelos lingüísticos basados en n-gramas utilizados en sistemas de reconocimiento del habla.

## 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 lingüística, reconocimiento y síntesis del habla. Se muestra como se maneja este proceso utilizando programas que lo realizan automáticamente; los etiquetadores gramaticales automáticos. Se describen implementaciones actuales que utilizan información estadística que el etiquetador utiliza 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 definición de POS o etiqueta gramatical se ha basado 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 vez que 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, etc.). 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 adjetivos 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 **fue** 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 *dobles* raciones.

## 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 descripción 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 un 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 Tree Bank*:

**Cuadro 2.1:** *Conjunto de Etiquetas Penn Tree Bank*

Etiqueta	Descripción	Ejemplo
CC	Coordinating conjunction	<i>and</i>
CD	Cardinal number	<i>1, third</i>
DT	Determiner	<i>the</i>
EX	Existential	<i>there there is</i>
FW	Foreign word	<i>d'hoevre</i>
IN	Preposition/subordinating conjunction	<i>in, of, like</i>
JJ	Adjective	<i>green</i>
JJR	Adjective, comparative	<i>greener</i>
JJS	Adjective, superlative	<i>greenest</i>
LS	List marker	<i>1)</i>
MD	Modal	<i>could, will</i>
NN	Noun, singular or mass	<i>table</i>
NNS	Noun plural	<i>tables</i>
NNP	Proper noun, singular	<i>John</i>
NNPS	Proper noun, plural	<i>Vikings</i>
PDT	Predeterminer both	<i>the boys</i>
POS	Possessive ending	<i>friend's</i>
PRP	Personal pronoun	<i>I, he, it</i>
PRP\$	Possessive pronoun	<i>my, his</i>
RB	Adverb	<i>however, usually, naturally, here, good</i>
RBR	Adverb, comparative	<i>better</i>
RBS	Adverb, superlative	<i>best</i>
RP	Particle	<i>give up</i>

**Cuadro 2.1:** Conjunto de Etiquetas Penn Tree Bank

Etiqueta	Descripción	Ejemplo
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
VCN	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	( [, (, {, i)
)	Right parenthesis	( ], ), }, i)
,	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 como 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 ago/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 vez de como un 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 ejemplo, 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.

## 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 discursos 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 ocurrencia. Con el desarrollo de la tecnología y de motores de búsqueda más precisos y eficientes, las posibilidades crecieron ampliamente.

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 influido 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 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.

### 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)* 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 construido en 1971 por el *Montreal French Project*, conteniendo 1 millón de palabras que inspiró a *Shana Poplack* a crear luego un corpus de Francés hablado mucho más grande.

Además de estos corpora de lenguaje vivo se encuentra corpora computarizado que fué construido 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. 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.

### 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) 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 ambigua. 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*). Análogamente *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 ambiguas, o lo que es lo mismo, tienen una única etiqueta posible. Sin embargo, muchas de las palabras más comunes del inglés son ambiguas, 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 ambiguas 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 ambigua 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 ambigua 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.

#### 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 primera 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 desambiguación 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 orden 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<sup>1</sup> y etiquetadores basados en el modelo de máxima entropía.

---

<sup>1</sup>Por las siglas en inglés de Hidden Markov Model

## 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.

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 ambigua 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  $\hat{\cdot}$  para decir “nuestra estimación de la secuencia de etiquetas correcta”.

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

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:

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

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

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



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:

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

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 etiquetaa 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.

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

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<sup>2</sup> de una probabilidad

---

<sup>2</sup>Por sus siglas en inglés Maximum Likelihood Estimated

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:

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

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*<sup>3</sup>, 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 etiqueta 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 cálculos.

### 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

<sup>3</sup>*is* es el presente en tercera persona del verbo *to be*

evidencia; los hechos conocidos. Estas restricciones son llamadas características o *features* y se expresan mediante funciones.

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:

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

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 ambiguas.

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<sup>4</sup>. 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 aleatoriamente 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 verificación diferente seleccionado aleatoriamente. 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.

---

<sup>4</sup>También llamado *devtest*

## 2.6. Evaluación de etiquetadores gramaticales

Los etiquetadores gramaticales generalmente son evaluados comparando su *accuracy* contra un corpus de verificación<sup>5</sup> 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*, 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* 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 experimentos realizados por *Voutilainen* encontraron que cuando a los humanos se les permitió discutir etiquetas, alcanzaron un consenso en el 100 % de las etiquetas.

Por otro lado el límite inferior sugerido por *Gale* es elegir la etiqueta más probable aplicando el modelo de unigrama para cada palabra ambigua. 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).

---

<sup>5</sup>También llamado *Gold Standard*

## 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  $n \times n$  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	VCN
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
VCN		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 VCN. La matriz anterior y el análisis de error relacionado en *Franz, Kupiec y Ratnaparkhi* 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 VCN 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.

## 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)* y *Dermatas y Kokkinakis (1995)*. 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 resumen, 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)* 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)* y *Brants (2000)*, generaliza el uso de morfología en una manera basada en datos. En este acercamiento, en vez 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  $t_i$ :

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

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  $\leq 10$ .



## 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 orden para la etiquetación gramatical. Técnicamente calcula, dada una secuencia de  $T$  palabras  $w_1, \dots, w_T$

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

para hallar las etiquetas  $t_1, \dots, t_T$ . 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 trigramas para calcular confiablemente su probabilidad asociada. Incluso establecer a cero la probabilidad de un trigramas 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 trigramas 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  $\lambda_1$ ,  $\lambda_2$  y  $\lambda_3$  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  $\lambda_1$ ,  $\lambda_2$  y  $\lambda_3$  tienen el mismo valor para todos los trigramas, o lo que es lo mismo,  $\lambda_1$ ,  $\lambda_2$  y  $\lambda_3$  son independientes del trigramas que se está calculando. Los valores  $\lambda_1$ ,  $\lambda_2$  y  $\lambda_3$  son estimados por interpolación de borrado. La idea es que se dará mayor peso a la información de unigrama, bigrama o trigramas más abundante. A continuación se presenta el algoritmo utilizado para realizar esta tarea

---

**Algoritmo 1** Cálculo de  $\lambda_1$ ,  $\lambda_2$  y  $\lambda_3 = 0$

---

**Establecer**  $\lambda_1 = \lambda_2 = \lambda_3 = 0$   
**por cada** trigramas  $t_1, t_2, t_3$  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  $\lambda_1$  en  $f(t_1, t_2, t_3)$   
    **caso**  $\frac{f(t_2, t_3)-1}{f(t_2)-1}$  : incrementar  $\lambda_2$  en  $f(t_1, t_2, t_3)$   
    **caso**  $\frac{f(t_3)-1}{N-1}$  : incrementar  $\lambda_3$  en  $f(t_1, t_2, t_3)$   
  **fin**  
**fin**  
**normalizar**  $\lambda_1, \lambda_2$  y  $\lambda_3$

---

### 2.9.3. Manejo de palabras desconocidas

TnT, al igual que muchos otros etiquetadores gramaticales, maneja las palabras desconocidas mediante 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  $P(t|l_{n-m+1}, \dots, l_n)$ , 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  $\theta_i$  a cada uno:

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

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

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

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

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

TnT utiliza desvío estándar del estimador de máxima verosimilitud para calcular los pesos  $\theta_i$ .

**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  $\theta_i$ , la misma idea que se utilizó para calcular  $\lambda_i$ .
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 mayúsculas son distintas a las de las palabras con minúsculas. Para llevar esto a cabo se utilizan *flags* en las probabilidades contextuales. En vez 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.

## 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  $h$ <sup>6</sup> 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:

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

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

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

$$\text{El sufijo de } w_i = S \wedge |S| < 5 \wedge t_i = T$$

$$\text{El prefijo de } w_i = P \wedge 1 < |P| < 5 \wedge t_i = T$$

$$w_i \text{ contiene un número} \wedge t_i = T$$

$$w_i \text{ contiene una mayúscula} \wedge t_i = T$$

$$w_i \text{ contiene un guión} \wedge t_i = T$$

---

<sup>6</sup>El contexto generalmente se define como una secuencia de varias palabras y etiquetas precediendo a la palabra actual

Las palabras raras son aquellas que aparecen pocas veces en el corpus de entrenamiento<sup>7</sup>.

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.

---

<sup>7</sup>Stanford Tagger toma como palabras raras aquellas que aparecen menos de 7 veces en el corpus

## 2.11. Diccionario Cobuild

Como se mencionó anteriormente, utilizaremos el diccionario Cobuild para generar un nuevo corpus de entrenamiento. Cobuild es una fuente de información que contiene un conjunto de entradas. Cada entrada está asociada a una palabra; posee una explicación de su significado, algunas características como su pronunciación y clase gramatical y uno o más ejemplos que muestran su uso. Para cada uno de los ejemplos de una entrada, Cobuild posee información gramatical sobre la palabra definida. Con esta información alcanza para construir un corpus parcialmente anotado. El proceso consiste en pegar o concatenar las palabras de los ejemplos y asignar la etiqueta gramatical correspondiente a la información gramatical que Cobuild posee para la palabra asociada. Por ejemplo, en la siguiente entrada Cobuild asigna la información gramatical *countable noun* para la primera aparición de la palabra (la definición) *siren* y *noun* para la segunda (el ejemplo).

```
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
```

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
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 Tree Bank y verse como:

A woman is described as a siren/NN when she is attractive and dangerous to men.  
 One of the women, another of those sirens/NNS, haughtily regarded us as we talked.  
 A sirloin/NN is a piece of beef which is cut from the lower part of a cows back.  
 ... a sirloin/NN of Scotch beef.  
 A boy is described as a sissy/NN, especially by other boys, if he does not like sport and  
 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 parcialmente anotado 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* y el corpus *Collins*. Su siglas significan: *Collins Birmingham University International Language Database*.

El corpus *Collins* es una base de datos con alrededor de 2.5 billones de palabras en inglés. Contiene material escrito de websites, diarios, revistas y libros publicados en todo el mundo, y material hablado de radio, TV y conversaciones diarias.

A su vez el *Bank of English* forma parte del corpus *Collins*. Contiene 650 millones de palabras cuidadosamente seleccionadas para dar reflejo preciso y balanceado del inglés que se usa día a día.

El corpus y el software que se utilizó para analizarlo ayudó al equipo de Cobuild a ordenar la información y ganar percepción sobre la manera en que se utilizan las palabras: sus significados, sus patrones gramaticales típicos y las maneras en que están relacionadas con otras palabras.

El diccionario Cobuild fué concebido teniendo especial atención en los ejemplos expuestos. El proceso de agregado de palabras fué muy cuidadoso. Cada palabra incluida en el diccionario fué elegida utilizando información sobre la frecuencia de ocurrencia de la misma. Por ejemplo, alrededor del 90 % del inglés hablado y escrito está constituido por aproximadamente 3.500 palabras.

Todos los ejemplos del diccionario Cobuild muestran patrones gramaticales típicos, vocabulario típico y contextos típicos para cada palabra. En consecuencia, Cobuild presenta una cantidad exhaustiva del vocabulario inglés derivado de observaciones directas del lenguaje.

### 2.11.2. Método de construcción

En 1987 se publicó el diccionario Cobuild basado en un corpus de 20 millones de palabras. A continuación se construyó un nuevo corpus, el *Bank of English* con alrededor de 650 millones de palabras. La nueva edición del diccionario

Cobuild se basa en este nuevo corpus. La construcción de Cobuild fué un proceso en donde se decidió que palabras y frases presentes en el corpus incluir. Luego se examinó el lenguaje palabra por palabra y frase por frase con el objetivo de dar clara cuenta de cada significado y uso. Entonces para cada entrada se escribió la definición, se eligieron ejemplos típicos, y se agregó información sobre la pronunciación, la gramática, semántica, pragmatismos y frecuencia.

### **2.11.3. Evidencia**

Un diccionario debe comenzar por la evidencia, los hechos. Los hablantes de un lenguaje conocen mucho sobre éste porque cada día leen y hablan sin esfuerzo durante horas. Sin embargo no son capaces de explicar exactamente que es lo que hacen. La mayoría de las personas no son conscientes de la habilidad que poseen para utilizar un lenguaje; no pueden examinarlo en detalle, simplemente lo utilizan para comunicarse. Aquellos que aprenden a observar el lenguaje cuidadosamente pueden expresar y organizar algunos de los hechos sobre éste basados en la experiencia. De todas maneras hay muchos hechos sobre el lenguaje que no pueden ser descubiertos simplemente pensando y reflexionando sobre él, incluso leyendo y escuchando muy atentamente. Es por eso que Cobuild empleó las computadoras para identificar estos hechos.

### **2.11.4. Un corpus**

El resultado fué que Cobuild estableció la utilización de corpora para construir diccionarios. Los creadores de Cobuild sabían que necesitaban millones de palabras de inglés, hablado y escrito, americano y británico, formal e informal, sobre hechos y sobre ficción, etc. Esta evidencia reunida durante varios años, permitió encontrar las palabras y expresiones más utilizadas. Cuando una palabra tiene varios significados existe la capacidad de ver cuales son los significados importantes, y que frases se deben incluir. Tomaron como filosofía y fueron conscientes de que todos los detalles de un uso natural de una palabra son esenciales y no pueden ser falsificados. Se dieron cuenta de que debían utilizar ejemplos reales siguiendo la tradición de los grandes lingüistas, en lugar de crearlos.

### **2.11.5. The Bank of English**

Diseñando el corpus *The Bank of English* se balancearon un número de factores importantes (inglés hablado y escrito, americano y británico y otras características: hablantes de comunidades nativas, libros y revistas y más clasificaciones dentro de éstas).

Dentro del componente hablado, el tipo de lenguaje más difícil de recolectar fué como siempre la conversación informal grabada en la vida diaria de la gente común, sin pensar de que su lenguaje está siendo preservado en un corpus. Cada conversación tuvo que ser grabada y transcrita por expertos para luego ser ingresada en una computadora. Esta clase de lenguaje improvisado es de un interés particular para los constructores de diccionarios. El *Bank of English* cuenta con un total de 15 millones de palabras de este tipo de grabaciones de lenguaje hablado.



### 2.11.6. La lista de palabras principales

Es mucho más fácil decidir qué palabras y frases incluir y cuales omitir, cuando se tienen cifras exactas sobre una cantidad tan grande de lenguaje. Un diccionario (incluso un gran diccionario) es capaz de presentar solo los hechos más importantes del lenguaje y los compiladores necesitan buena evidencia para realizar sus selecciones. 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 contemporáneo.

### 2.11.7. Ejemplos

Todos los ejemplos fueron cuidadosamente seleccionados, como se dijo anteriormente, del corpus *The Bank of English* para mostrar los patrones que aparecen frecuentemente junto a una palabra o frase.

Desde luego éstos ayudan a mostrar el significado de la palabra exhibiendo su uso. Más aún, las investigaciones sugieren que un gran número de usuarios comienza leyendo los ejemplos antes que el significado.

Las definiciones de Cobuild son bastante claras por sí mismas y como los ejemplos son piezas de texto genuinas y han sido elegidos cuidadosamente en base al uso de la palabra, pueden ser de confianza para exhibir la palabra en un contexto natural.

### 2.11.8. 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.

### 2.11.9. Pragmatismo

Muchos usos de una palabra necesitan más de una frase para explicar apropiadamente su significado.

El estudio y descripción de las formas en que la gente utiliza el lenguaje para realizar cosas es llamado pragmatismo. Este aspecto del lenguaje es muy importante y fácil de omitir. Cobuild posee mucha información sobre pragmatismo y la expone mediante un símbolo especial en cada entrada. Por ejemplo *and things like that* es definido como una expresión utilizada para ampliar el rango de una lista.

### 2.11.10. Definición del estilo

La característica más distintiva de Cobuild en su primera versión fué el uso de frases completas en las definiciones. El significado de una palabra fué 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.

Las definiciones cortas no pueden decir demasiado. Por ejemplo, el primer sentido de verbo de *mean* podría ser definido como solo *signify*, que es cierto, pero no es todo lo que se puede decir. Cobuild expone esto: *If you want to know...* es decir que ese sentido surge cuando alguien está en la búsqueda de información. La palabra *if* indica que esta es una opción, pero una perfectamente normal, y *you* nos dice que no es una característica de ningún grupo particular de gente (compararlo con *if a policeman arrests you...*). Entonces la definición dice lo que alguien puede querer saber sobre el significado de una *palabra*, *código*, *señal o gesto*, indicando que esas son las típicas clases de temas que serán encontradas con este sentido de *mean*. Solo después de toda esta información viene el equivalente de *signify*: *lo que se refiere a o a que mensaje transmite*. Entonces hay 12 palabras antes de la palabra principal en este sentido, pero cada una de ellas transmite información vital que sería muy difícil de incluir en una definición corta.

## 2.12. Corpus BNC

El *British National Corpus* (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 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 transcrito 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* (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 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

### 3.1. Extracción de la información

El diccionario Cobuild guarda su información en un archivo de texto plano con un formato particular. El primer desafío de este trabajo fué comprender y extraer la información almacenada en ese archivo. A continuación se muestra un pequeño fragmento del mismo para ejemplificar

```
DICTIONARY_ENTRY
ace
aces
*e*!is
If you are or come within an ace of something, you very nearly do or experience it.
He came within an ace of being run over.
phrase: verb inflects
phrase

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
```

Cada entrada arriba presentada tiene la característica de poseer una cantidad variable de campos y no es posible identificarlos exactamente. Sin embargo, contienen algunos rasgos comunes: 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
If you are or come within an ace of something, you very nearly do or experience it.
→ definición
He came within an ace of being run over. → ejemplo
phrase: verb inflects → etiqueta
phrase → etiqueta
```

Estas entradas, que conforman el diccionario Cobuild y que constituyen la fuente de información principal sobre la cual se basa este trabajo, fueron cuidadosamente procesadas y refinadas intentando mantener toda la información disponible. El primer desafío de esta etapa consistió en recuperar las entradas con toda la información gramatical disponible; explícita e implícita. Una primer

tarea fué reconocer y registrar información relacionada a las formas flexionadas de la palabra (plurales, pasados, etc.), es decir, obtener información gramatical implícita.

### 3.1.1. Reconocimiento de formas flexionadas

En muchas entradas del diccionario Cobuild ocurre la palabra, uno o más ejemplos en donde ésta aparece con cierto sentido (indicado por medio de etiquetas gramaticales) pero dentro de los ejemplos hay apariciones de formas flexionadas. Tomemos la siguiente entrada:

DICTIONARY\_ENTRY

bite → *palabra*

bites, biting, bit, bitten → *formas flexionadas*

b\*a\*!it → *pronunciación*

If an object or surface bites, it grips another object or surface rather than slipping on  
→ *definición*

Let the clutch in slowly until it begins to bite. → *ejemplo*

verb → *etiqueta para la definición*

verb → *etiqueta para el ejemplo*

Aquí arriba se puede observar una entrada del diccionario para la palabra *bite*, que contiene la definición y un ejemplo de esta palabra con sus respectivas etiquetas:

(1) *If an object or surface bites, it grips another object or surface rather than slipping on it or against it.*

(2) *Let the clutch in slowly until it begins to bite.*

En (2) aparece la palabra *bite* en su forma regular con la etiqueta *verb* mientras que en (1) aparece la forma flexionada *bites* con la etiqueta *verb*. En este caso (1) está ofreciendo más información gramatical que la expuesta por medio de la etiqueta. Reconociendo la forma flexionada (*bites*) podemos adicionarle información extra a la etiqueta *verb*; en vez de guardar la etiqueta *Penn Treebank* correspondiente a *verb* (VB), en este caso guardaríamos la etiqueta VBZ (verbo de tiempo presente en tercera persona singular) que contiene más información gramatical que VB.

Las entradas de Cobuild exponen las formas derivadas de la palabra que pueden contener los ejemplos. En el ejemplo presentado anteriormente la palabra es *bite* y las formas derivadas de *bite* que muestra la entrada son *bites*, *biting*, *bit* y *bitten*. Con esta información y la etiqueta que fué anotada en Cobuild (*verb*) se pueden inferir y generar etiquetas de *Penn Treebank* con información adicional.

Como ya se mencionó anteriormente, en este caso la forma *bites* (derivada de la palabra *bite*) que aparece en la definición posee la etiqueta *verb*. La tarea aquí será reconocer que *bites* es un verbo de tiempo presente en tercera persona singular a partir de que *bites* está etiquetada como verbo y de que la palabra de la cual deriva es *bite*. Es decir, inferir el tipo de la forma derivada a partir de la palabra y la etiqueta asignada por Cobuild.

Con el objetivo de identificar las formas derivadas de una palabra se desarrollaron reglas y métodos para su reconocimiento, buscando preservar y aprovechar toda la información que ofrece Cobuild. Entonces, a partir de esta información: la palabra, la forma en que ocurre y la etiqueta asignada se aplican las siguientes reglas para reconocer información adicional a la etiqueta gramatical.

---

**Algoritmo 2** Reconocimiento de formas derivadas

---

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**

---

Aplicando algoritmos de extracción y el algoritmo de reconocimiento de formas derivadas explicado anteriormente se obtiene un nuevo corpus parcialmente anotado a partir del diccionario Cobuild. A continuación este corpus será procesado y utilizado como corpus de entrenamiento.

### 3.1.2. Preprocesamiento

Se tomó el archivo cobuild en bruto, se hizo un primer preprocesado eliminando caracteres no ascii para que pudiera ser legible. Luego se separó cada entrada y se procesó individualmente identificando el/los ejemplos de cada entrada, la palabra que se está definiendo y su etiqueta asociada. También se tuvieron en cuenta palabras derivadas y cuando fué posible (cuando no hubo ambigüedad) se infirieron etiquetas para las mismas. Luego se tradujeron las etiquetas de Cobuild a Penn Tree Bank, para esto se utilizó un diccionario construido a doc. No fué posible incluir todas las etiquetas Cobuild, ya que no se conoce el formato concreto de las mismas ni pertenecen a ningún conjunto de etiquetas documentado. Sin embargo se realizó un análisis para contar las etiquetas Cobuild que más se repetían y en base a esas etiquetas se construyó la traducción. También cabe aclarar que las etiquetas Cobuild son muy ricas sintácticamente y un poco de esa información se perdió en la traducción ya que las etiquetas Penn Tree Bank carecen de ese nivel de detalle. Una vez realizado este proceso, verificamos que el etiquetado haya sido correcto. Se realizó una rutina que etiquetara automáticamente todos los ejemplos de Cobuild (se utilizó el etiquetador TnT)



y se comparó el resultado contra las etiquetas asignadas (se generó una matriz de confusión). El resultado fué de 71 % de aciertos. Se analizaron las etiquetas que diferían con mayor frecuencia, y en algunos casos se analizaron palabras particulares donde las etiquetas no coincidían. Se corrigieron las traducciones y se ajustaron los algoritmos del proceso de extracción y traducción de etiquetas hasta alcanzar un grado de errores mínimo. Los máximos focos de error que no se pudieron corregir se deben a palabras etiquetadas con VBD cuando son VBN y viceversa. Estas etiquetas son difícilmente desambiguables automáticamente.

Una vez obtenidos los ejemplos con las etiquetas traducidas a Penn Tree Bank, se realiza un nuevo proceso para obtener precisión gramatical perdida en la traducción

Para eso se comparan las etiquetas asignadas 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:

Si se obtuvo de Cobuild NN pero TnT asigno NNS, NNP o NNPS, asignar tag TnT  
Si se obtuvo de Cobuild NNS pero TnT asigno NNPS, asignar NNPS  
Si se obtuvo de Cobuild VB pero TnT asigno VBN, VBD, VBZ, VBP, VBG asignar tag TnT  
Si se obtuvo de Cobuild JJ pero TnT asigno JJR o JJS asignar tag TnT  
Si se obtuvo de Cobuild RB pero TnT asigno RBR o RBS asignar tag TnT  
Si se obtuvo de Cobuild WP pero TnT asigno WP\$ asignar tag TnT  
Si se obtuvo de Cobuild PRP pero TnT asigno PRP\$ asignar tag TnT

Luego de este proceso, el análisis de comparación contra las etiquetas asignadas por TnT dió un 86 % de aciertos.

### 3.2. Traducción de etiquetas

Para cada una de sus definiciones, el diccionario Cobuild expone información gramatical expresada mediante etiquetas. Estas etiquetas gramaticales poseen un formato propio. Por ejemplo en la siguiente entrada de Cobuild para la palabra *canary*

```
DICTIONARY_ENTRY
canary
canaries
A canary is a small yellow bird which sings beautifully.
People sometimes keep canaries in cages as pets.
countable noun
noun
```

Se expone la definición (1) y un ejemplo (2), ambos con información gramatical sobre la palabra:

- (1) *A canary is a small yellow bird which sings beautifully.*  
(2) *People sometimes keep canaries in cages as pets.*

Se puede apreciar la etiqueta *noun* asignada por Cobuild para *canary*.

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 poder medir posteriormente los resultados. Es por eso que se tomó la decisión de traducir estas etiquetas propias de Cobuild en etiquetas *Penn Treebank*, conjunto con el cual está anotado el *Gold Standard*.

A continuación se presenta la tabla de traducción empleada:

**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
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

**Cuadro 3.1:** *Tabla de traducción de etiquetas*

<b>Etiqueta Cobuild</b>	<b>Etiqueta Penn Treebank</b>
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
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

**Cuadro 3.1:** *Tabla de traducción de etiquetas*

<b>Etiqueta Cobuild</b>	<b>Etiqueta Penn Treebank</b>
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

### 3.3. Nuevo Corpus generado

A partir del corpus parcialmente anotado obtenido en el proceso de extracción, se completarán las anotaciones automáticamente con un etiquetador gramatical manteniendo las etiquetas gramaticales obtenidas a partir de la información procedente del diccionario Cobuild. Es decir, una vez finalizado el proceso de extracción de información desde el diccionario, se obtiene un corpus nuevo con las etiquetas gramaticales correspondientes a las palabras definidas en el diccionario. A continuación se exhibe un fragmento del mismo:

```
A
canary NN
is
a
small
yellow
bird
which
sings
beautifully
.
People
sometimes
keep
canaries NNS
in
cages
as
pets
.
```

Este es el resultado de extracción y reconocimiento de formas flexionadas correspondiente a la entrada de Cobuild:

```
DICTIONARY_ENTRY
canary
canaries
A canary is a small yellow bird which sings beautifully.
People sometimes keep canaries in cages as pets.
countable noun
noun
```

Se puede apreciar que se ha reconocido *canaries* como el plural de *canary* (etiqueta NNS) y que se han reconocido y extraído los ejemplos de estas palabras asignando las etiquetas gramaticales traducidas a partir de las etiquetas del diccionario correspondientes a *canary* (countable noun/NN) y *canaries* (noun/NNS).

El próximo paso será el de completar las anotaciones gramaticales para todas las palabras restantes. Este proceso se realiza anotando el corpus plano (sin las etiquetas obtenidas de Cobuild) con el etiquetador gramatical automático TnT. Luego se une este corpus anotado por TnT con el corpus anotado parcialmente procedente de Cobuild, preservando todas las etiquetas del diccionario.

El resultado que se muestra a continuación es un nuevo corpus obtenido a partir de Cobuild, con las anotaciones que este provee y completado con anotaciones obtenidas mediante etiquetación automática utilizando TnT.

A	DT
canary	NN
is	VBZ
a	DT
small	JJ
yellow	JJ
bird	NN
which	WDT
sings	VBZ
beautifully	RB
.	.
People	NNS
sometimes	RB
keep	VB
canaries	NNS
in	IN
cages	NNS
as	IN
pets	NNS
.	.

## 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). Es decir, la información extraída de COBUILD, como se mencionó anteriormente, es la unión de definiciones y ejemplos, con la información gramatical correspondiente a la palabra definida. A continuación se presenta un pequeño extracto:

A	are
cat NN	often
is	kept
a	as
small	pets
furry	.
animal	She
with	put
a	out
tail	a
,	hand
whiskers	and
,	stroked
and	the
sharp	cat NN
claws	softly
that	...
kills	...
smaller	domestic
animals	animals
such	such
as	as
mice	dogs
and	and
birds	cats NNS
.	.
Cats NNS	

Esta es la información extraída de COBUILD para la palabra *cat*; la unión de la definición:

*A cat is a small furry animal with a tail, whiskers, and sharp claws that kills smaller animals such as mice and birds. Cats are often kept as pets.*

y 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 eti-



quetará utilizando TnT entrenado con el corpus de entrenamiento Wall Street Journal (de ahora en más WSJ) <sup>1</sup> y luego se realizará la comparación. La matriz de confusión<sup>2</sup> generada a partir de dicha comparación es la siguiente:

**Cuadro 4.1:** Matriz de confusión para etiquetas extraídas de COBUILD vs generadas por TnT

COBUILD \ TnT	NN	VB	JJ	VBN	RB	VBG	NNP	IN	VBZ	NNS
NN	-	<b>556</b>	<b>1953</b>	52	86	276	-	8	-	-
VB	<b>2616</b>	-	<b>614</b>	-	42	-	77	15	-	5
JJ	<b>1577</b>	96	-	<b>1361</b>	<b>634</b>	<b>555</b>	<b>281</b>	30	-	16
VBN	-	-	-	-	-	-	-	-	-	-
RB	219	23	<b>408</b>	10	-	9	34	249	-	11
VBG	-	-	-	-	-	-	-	-	-	-
NNP	-	-	-	-	-	-	-	-	-	-
IN	-	-	-	-	-	-	-	-	-	-
VBZ	-	-	-	-	-	-	-	-	-	-
NNS	83	1	17	-	1	2	104	3	192	-

Porcentaje de aciertos: 99,16 %

Cantidad de errores: 13082

Se puede apreciar un alto porcentaje de aciertos entre las etiquetas extraídas de COBUILD (99,16 %) 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 VB, NN y JJ de COBUILD cuando son etiquetadas como NN, JJ y NN por TnT respectivamente.

<sup>1</sup>Wall Street Journal es un corpus anotado, parte del Penn Treebank

<sup>2</sup>Las matrices de confusión presentadas de aquí en adelante contienen las primeras 10 etiquetas de mayor error

## 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 entrenarán 2 etiquetadores gramaticales (Stanford Tagger y TnT) y se etiquetará con ellos el corpus Wall Street Journal (WSJ). Posteriormente se realizarán mediciones de desempeño pertinentes.

### 4.2.1. Etiquetar el corpus WSJ con TnT

La primer evaluación de este segundo experimento consiste en entrenar el etiquetador gramatical TnT 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 4.2:** *WSJ original contra WSJ etiquetado con TnT (entrenado con WSJ)*

TnT(WSJ) WSJ	JJ	NNP	VBN	RB	IN	RP	NNPS	VBD	NN	VB
<b>NN</b>	<b>2592</b>	<b>1649</b>	28	79	18	7	-	39	-	316
<b>VBD</b>	70	5	<b>1642</b>	-	-	-	-	-	42	41
<b>IN</b>	108	30	-	<b>1476</b>	-	<b>1376</b>	-	-	4	2
<b>RB</b>	726	45	4	-	<b>1459</b>	857	-	1	200	37
<b>VBN</b>	<b>1262</b>	22	-	-	-	-	-	<b>1146</b>	36	53
<b>NNP</b>	484	-	4	46	23	1	<b>1236</b>	3	378	7
<b>JJ</b>	-	699	<b>916</b>	601	69	23	1	42	903	70
<b>VBG</b>	424	20	-	-	-	-	-	-	884	-
<b>VBP</b>	26	10	33	11	10	1	1	51	349	872
<b>WDT</b>	-	-	-	-	722	-	-	-	-	-

Aciertos: 1.226.484 (97,10 %)

Errores: 36.636 (2,90 %)

**Cuadro 4.3:** *WSJ original contra WSJ etiquetado con TnT (entrenado con WSJ + NFI)*

TnT(WSJ+NFI) WSJ	JJ	NNP	VBN	IN	RP	NNPS	RB	VBD	VB	NN
<b>NN</b>	<b>2794</b>	<b>2081</b>	46	16	7	-	87	43	315	-
<b>VBD</b>	73	10	<b>1716</b>	-	-	-	-	-	37	22
<b>RB</b>	799	62	4	<b>1664</b>	<b>1128</b>	-	-	1	50	204
<b>IN</b>	107	51	-	-	<b>1559</b>	-	<b>1222</b>	-	3	4
<b>VBN</b>	<b>1369</b>	27	-	-	-	-	-	<b>1219</b>	42	33
<b>NNP</b>	358	-	4	19	1	<b>1257</b>	44	3	9	263
<b>JJ</b>	-	1013	893	87	28	1	634	47	74	864
<b>VBP</b>	30	12	33	10	1	1	8	56	885	379

**Cuadro 4.3:** *WSJ original contra WSJ etiquetado con TnT (entrenado con WSJ + NFI)*

$\begin{matrix} \text{TnT(WSJ+NFI)} \\ \text{WSJ} \end{matrix}$	JJ	NNP	VBN	IN	RP	NNPS	RB	VBD	VB	NN
<b>VBG</b>	518	27	-	-	-	-	-	-	-	816
<b>VBZ</b>	-	11	-	-	-	2	-	-	-	1

Aciertos: 1.226.967 (97,14 %)

Errores: 36.153 (2,86 %)

Se puede observar que el rendimiento del etiquetador TnT entrenado con WSJ+NFI es un poco mejor (97,14 %) que el rendimiento de TnT entrenado con WSJ (97,1 %). 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:** *1 mitad WSJ original contra 1 mitad WSJ etiquetado con TnT (entrenado con 2 mitad de WSJ)*

$\begin{matrix} \text{TnT(2WSJ)} \\ \text{WSJ} \end{matrix}$	JJ	NNP	VBN	NN	VBD	IN	RB	RP	VB	NNPS
<b>NN</b>	<b>1959</b>	<b>1154</b>	26	-	24	5	60	2	269	2
<b>VBD</b>	76	12	<b>1129</b>	19	-	-	1	-	29	-
<b>JJ</b>	-	545	<b>801</b>	<b>1039</b>	52	19	313	9	62	1
<b>VBN</b>	<b>617</b>	23	-	24	<b>819</b>	-	-	-	36	-
<b>RB</b>	432	25	3	91	2	<b>808</b>	-	318	19	-
<b>IN</b>	71	24	1	3	-	-	<b>634</b>	<b>615</b>	1	-
<b>VBP</b>	26	19	19	285	33	6	4	-	613	1
<b>NNP</b>	419	-	8	534	11	19	43	-	20	600
<b>VBG</b>	276	22	-	577	-	-	-	-	-	-
<b>NNPS</b>	26	549	-	-	-	-	-	-	-	-

Aciertos: 607.876 (96,25 %)

Errores: 23.695 (3,75 %)

**Cuadro 4.5:** *1 mitad WSJ original contra 1 mitad WSJ etiquetado con TnT (entrenado con 2 mitad de WSJ + NFI)*

$\begin{matrix} \text{TnT(2WSJ+NFI)} \\ \text{WSJ} \end{matrix}$	JJ	NNP	VBN	IN	NN	RP	VBD	NNPS	VBG	VB
--	----	-----	-----	----	----	----	-----	------	-----	----

**Cuadro 4.5:** 1 mitad WSJ original contra 1 mitad WSJ etiquetado con TnT (entrenado con 2 mitad de WSJ + NFI)

TnT(2WSJ+NFI) WSJ	JJ	NNP	VBN	IN	NN	RP	VBD	NNPS	VBG	VB
<b>NN</b>	<b>1759</b>	<b>1287</b>	29	6	-	3	27	1	556	213
<b>VBD</b>	54	17	<b>1039</b>	-	10	-	-	-	-	17
<b>RB</b>	434	30	2	<b>872</b>	81	511	1	-	-	23
<b>JJ</b>	-	<b>689</b>	<b>612</b>	37	<b>838</b>	6	29	-	219	45
<b>IN</b>	65	33	-	-	3	<b>749</b>	-	-	2	1
<b>VBN</b>	<b>654</b>	24	-	-	16	-	<b>708</b>	-	-	21
<b>NNP</b>	334	-	6	15	356	-	4	558	23	18
<b>VBP</b>	14	19	20	6	248	-	28	1	-	534
<b>NNPS</b>	20	510	-	-	1	-	-	-	-	-
<b>VBG</b>	322	22	-	-	460	-	-	-	-	-

Aciertos: 609.255 (96,47 %)

Errores: 22.316 (3,53 %)

**Cuadro 4.6:** 2 mitad WSJ original contra 2 mitad WSJ etiquetado con TnT (entrenado con 1 mitad de WSJ)

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

Aciertos: 607.593 (96,21 %)

Errores: 23.956 (3,79 %)

**Cuadro 4.7:** 2 mitad WSJ original contra 2 mitad WSJ etiquetado con TnT (entrenado con 1 mitad de WSJ)

TnT(1WSJ+NFI) WSJ	JJ	NNP	VBN	RP	IN	NN	NNPS	VBD	RB	VBG
<b>NN</b>	<b>1831</b>	<b>1261</b>	33	4	10	-	-	28	41	518
<b>VBD</b>	39	12	<b>1065</b>	-	-	14	-	-	-	-
<b>IN</b>	43	26	-	<b>870</b>	-	4	-	-	679	1
<b>RB</b>	411	51	2	<b>682</b>	<b>861</b>	137	-	-	-	1
<b>VBN</b>	<b>829</b>	26	-	-	-	22	-	<b>726</b>	-	-

**Cuadro 4.7:** 2 mitad WSJ original contra 2 mitad WSJ etiquetado con TnT (entrenado con 1 mitad de WSJ)

TnT(1WSJ+NFI) WSJ	JJ	NNP	VBN	RP	IN	NN	NNPS	VBD	RB	VBG
<b>JJ</b>	-	664	495	21	51	<b>819</b>	2	32	372	125
<b>NNP</b>	333	-	17	1	11	321	<b>790</b>	4	28	15
<b>VBP</b>	22	9	15	1	7	194	-	31	4	-
<b>VBG</b>	322	30	-	-	-	462	-	-	1	-
<b>VBZ</b>	1	14	-	-	-	1	7	-	-	-

Aciertos: 608.633 (96,37 %)

Errores: 22.916 (3,63 %)

Se puede apreciar una leve mejoría en el porcentaje de etiquetas acertadas para el modelo que incorpora NFI; 96,25 % contra 96,47 % y 96,21 % contra 96,37 % 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 presentan las matrices de confusión entre las mitades de WSJ etiquetado con TnT entrenado con la mitad restante con y sin NFI.

**Cuadro 4.8:** 1 mitad WSJ etiquetado por TnT (entrenado con 2 mitad WSJ) vs 1 mitad WSJ etiquetado con TnT (entrenado con 2 mitad de WSJ + NFI)

TnT(2WSJ+NFI) TnT(2WSJ)	NN	JJ	VBN	NNP	VBD	VBG	VBP	VB	RP	IN
<b>JJ</b>	<b>592</b>	-	149	<b>353</b>	38	64	11	62	1	26
<b>NN</b>	-	<b>580</b>	30	<b>488</b>	16	<b>338</b>	97	<b>315</b>	-	3
<b>VBD</b>	8	37	<b>506</b>	13	-	-	6	7	-	-
<b>VBN</b>	17	<b>377</b>	-	7	<b>483</b>	-	4	8	-	-
<b>VB</b>	203	55	25	12	22	-	<b>321</b>	-	1	-
<b>RB</b>	29	147	1	14	1	-	-	2	292	229
<b>VBP</b>	99	3	-	1	11	-	-	223	-	2
<b>VBG</b>	167	217	-	36	1	-	-	1	-	-
<b>VBZ</b>	-	-	-	6	-	-	-	1	-	-
<b>NNS</b>	56	9	-	115	-	-	-	-	-	-

Aciertos: 621.391 (98,39 %)

Errores: 10.184 (1,61 %)

**Cuadro 4.9:** 2 mitad WSJ etiquetado por TnT (entrenado con 1 mitad WSJ) vs 2 mitad WSJ etiquetado con TnT (entrenado con 1 mitad de WSJ + NFI)

TnT(1WSJ) \ TnT(1WSJ+NFI)	JJ	VBN	NN	VBD	NNP	RP	IN	VB	VBP	VBG
NN	<b>765</b>	28	-	27	<b>436</b>	1	4	<b>292</b>	72	270
VBD	59	<b>525</b>	11	-	2	-	-	8	3	-
JJ	-	193	<b>497</b>	47	<b>335</b>	-	11	48	23	82
VBN	<b>286</b>	-	22	<b>439</b>	4	-	-	11	5	-
RB	160	-	38	-	30	<b>326</b>	<b>301</b>	15	3	-
VB	44	18	173	16	17	1	1	-	275	1
VBZ	-	-	-	-	5	-	-	1	-	-
VBP	14	3	101	7	4	-	1	219	-	-
VBG	190	1	156	2	18	-	-	6	2	-
NNS	9	-	50	-	72	-	-	-	-	-

Aciertos: 621.749 (98,45 %)

Errores: 9.802 (1,55 %)

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 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	96.26 %
TnT entrenado con el segundo 1/4 de WSJ	95.89 %
TnT entrenado con el segundo 1/4 de WSJ + NFI	96.26 %
TnT entrenado con el tercer 1/4 de WSJ	95.91 %
TnT entrenado con el tercer 1/4 de WSJ + NFI	96.29 %
TnT entrenado con el cuarto 1/4 de WSJ	95.9 %
TnT entrenado con el cuarto 1/4 de WSJ + NFI	96.30 %

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
- 96.1 % 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.

#### 4.2.2. 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:

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:

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:

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:

En todos los casos se observa un comportamiento similar que el observado al etiquetar con TnT. Los datos y el detalle de estos experimentos se puede consultar en el apéndice



### 4.3. Tercer experimento: Etiquetar el corpus BNC

El tercer experimento realizado tiene como objetivo evaluar la nueva fuente de información obtenida (NFI) como corpus de entrenamiento. Para esto se entrenarán 2 etiquetadores gramaticales (Stanford Tagger y TnT) y se etiquetará con ellos el British National Corpus (BNC).

#### 4.3.1. 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 contruye la matriz de confusión:

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:

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:

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:

En todos los casos (al igual que en los experimentos anteriores) se puede apreciar una leve mejora cuando se introduce NFI en el corpus de entrenamiento. Los datos y el detalle de estos experimentos se puede consultar en el apéndice.

### 4.3.2. 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:

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:

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:

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.

En todos los casos (al igual que en los experimentos anteriores) se puede apreciar una leve mejora cuando se introduce NFI en el corpus de entrenamiento. Los datos y el detalle de estos experimentos se puede consultar en el apéndice.

## Capítulo 5

# Conclusiones

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 8 % de etiquetas por palabra, es decir que de cada 100 palabras que se extraen de los ejemplos del diccionario solo 8 poseen una etiqueta gramatical.

## Capítulo 6

# Apendice

### 6.0.3. 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:** *WSJ original contra WSJ etiquetado con MaxEnt (entrenado con WSJ)*

MaxEnt(WSJ)	JJ	IN	NN	NNP	VBD	RB	VBN	VBP	RP	JJR
WSJ										
NN	<b>1726</b>	15	-	<b>1132</b>	16	61	18	28	1	2
RB	736	<b>1593</b>	189	139	-	-	3	1	293	36
JJ	-	60	<b>1276</b>	632	51	515	762	7	-	5
VBN	<b>894</b>	-	44	25	<b>1052</b>	1	-	5	-	-
NNPS	40	-	-	<b>997</b>	-	-	-	-	-	-
IN	87	-	4	22	-	<b>959</b>	-	2	527	-
VBG	196	-	<b>829</b>	14	-	-	1	1	-	-
VBD	40	-	26	8	-	-	<b>806</b>	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:** *WSJ original contra WSJ etiquetado con MaxEnt (entrenado con WSJ + NFI)*

MaxEnt(WSJ+NFI)	JJ	IN	NNP	NN	RB	VBD	RP	VBN	VBP	NNPS
WSJ										
NN	<b>1918</b>	16	<b>1314</b>	-	73	21	3	19	30	-
RB	742	<b>1403</b>	141	177	-	-	527	3	3	-
JJ	-	68	685	<b>1260</b>	583	45	2	851	8	-

**Cuadro 6.2:** *WSJ original contra WSJ etiquetado con MaxEnt (entrenado con WSJ + NFI)*

MaxEnt(WSJ+NFI) WSJ	JJ	IN	NNP	NN	RB	VBD	RP	VBN	VBP	NNPS
<b>IN</b>	107	-	29	3	<b>1062</b>	-	<b>1005</b>	-	2	-
<b>VBN</b>	<b>980</b>	-	29	49	-	<b>1049</b>	-	-	6	-
<b>NNPS</b>	39	-	<b>935</b>	-	-	-	-	-	-	-
<b>VBD</b>	34	-	10	24	-	-	-	<b>923</b>	15	-
<b>VBG</b>	294	-	25	817	-	-	-	1	1	-
<b>NNP</b>	555	29	-	458	21	4	1	9	4	546
<b>VB</b>	62	10	29	361	13	50	-	38	555	-

Aciertos: 1.234.495 (97,73 %)

Errores: 28.629 (2,27 %)

Se puede observar que el rendimiento del etiquetador entrenado con WSJ es un poco mejor (97,9 %) que cuando es entrenado con WSJ + NFI (97,73 %). 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:** *1 mitad WSJ original contra 1 mitad WSJ etiquetado con MaxEnt (entrenado con 2 mitad de WSJ)*

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

Aciertos: 610.045 (96,59 %)

Errores: 21.529 (3,41 %)

**Cuadro 6.4:** 1 mitad WSJ original contra 1 mitad WSJ etiquetado con MaxEnt (entrenado con 2 mitad de WSJ + NFI)

MaxEnt(2WSJ+NFI)	JJ	NN	NNP	IN	VBN	VBD	RP	RB	NNS	VB
WSJ										
NN	<b>1434</b>	-	<b>1033</b>	6	13	14	1	41	124	124
JJ	-	<b>1119</b>	<b>571</b>	31	<b>625</b>	31	1	334	68	26
RB	470	81	73	<b>851</b>	2	1	251	-	1	19
NNPS	27	3	<b>834</b>	-	-	-	-	-	215	-
VBD	35	16	14	-	<b>748</b>	-	-	-	-	10
VBN	564	26	34	-	-	<b>642</b>	-	-	-	11
IN	80	7	27	-	-	-	<b>573</b>	531	-	1
VBG	262	531	26	-	1	-	-	-	-	-
NNP	445	497	-	19	3	6	-	24	141	16
VBZ	-	1	17	-	-	-	-	-	412	-

Aciertos: 611.099 (96,76 %)

Errores: 20.475 (3,24 %)

**Cuadro 6.5:** 2 mitad WSJ original contra 2 mitad WSJ etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

MaxEnt(1WSJ)	JJ	NN	IN	NNP	VBD	RB	VBN	NNPS	NNS	VBG
WSJ										
NN	<b>1604</b>	-	12	<b>916</b>	16	36	21	-	150	381
JJ	-	<b>1197</b>	45	522	37	381	483	-	46	202
RB	466	168	<b>944</b>	62	1	-	1	-	3	1
VBN	<b>863</b>	29	-	32	<b>779</b>	1	-	-	-	-
IN	50	3	-	26	-	<b>698</b>	-	-	-	2
NNPS	16	-	-	<b>651</b>	-	-	-	-	167	-
VBG	198	<b>572</b>	-	19	-	-	-	-	-	-
VBD	66	43	2	16	-	-	<b>570</b>	-	-	-
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:** 2 mitad WSJ original contra 2 mitad WSJ etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

MaxEnt(1WSJ+NFI)	JJ	NNP	NN	IN	VBN	RB	VBD	RP	NNPS	NNS
WSJ										
NN	<b>1482</b>	<b>1011</b>	-	10	18	42	12	2	-	146
JJ	-	522	<b>997</b>	43	444	382	26	8	1	40
RB	438	105	141	<b>819</b>	1	-	-	344	-	1
VBN	<b>810</b>	27	28	-	-	-	<b>706</b>	-	-	-
VBD	40	11	20	-	<b>742</b>	-	-	-	-	-

**Cuadro 6.6:** 2 mitad WSJ original contra 2 mitad WSJ etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

MaxEnt(1WSJ+NFI)	JJ	NNP	NN	IN	VBN	RB	VBD	RP	NNPS	NNS
WSJ										
<b>IN</b>	50	29	3	-	-	<b>727</b>	-	<b>586</b>	-	-
<b>NNPS</b>	13	<b>597</b>	-	-	-	-	-	-	-	171
<b>VBG</b>	256	21	530	-	-	-	-	-	-	-
<b>NNP</b>	475	-	467	16	8	17	3	1	483	140
<b>VBZ</b>	-	9	1	-	-	-	-	-	2	406

Aciertos: 610.874 (96,73 %)

Errores: 20.676 (3,27 %)

Se puede apreciar una leve mejoría en el porcentaje de etiquetas acertadas; 96,23 % contra 96,46 % y 96,20 % contra 96,36 % 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 WSJ + NFI.

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

**Cuadro 6.7:** 1 mitad WSJ etiquetado por MaxEnt (entrenado con 2 mitad WSJ) vs 1 mitad WSJ etiquetado con MaxEnt (entrenado con 2 mitad de WSJ + NFI)

MaxEnt(2WSJ+NFI)	JJ	NN	RP	VBN	RB	NNP	VB	IN	VBD	VBG
MaxEnt(2WSJ)										
<b>NN</b>	<b>686</b>	-	2	13	46	<b>291</b>	<b>276</b>	-	21	213
<b>JJ</b>	-	<b>596</b>	1	175	202	183	48	15	19	49
<b>IN</b>	17	6	<b>507</b>	-	<b>318</b>	8	-	-	-	-
<b>VBD</b>	29	9	-	<b>460</b>	-	11	11	-	-	-
<b>VBN</b>	<b>309</b>	18	-	-	-	9	2	-	248	-
<b>NNP</b>	<b>268</b>	242	-	5	8	-	14	4	-	10
<b>WDT</b>	-	-	-	-	-	-	-	<b>252</b>	-	-
<b>VBP</b>	15	123	-	5	2	6	246	-	13	-
<b>RB</b>	128	17	206	1	-	65	14	141	-	-
<b>VBG</b>	199	196	-	-	1	26	-	-	-	-

Aciertos: 622.105 (98,50 %)

Errores: 9.469 (1,50 %)

**Cuadro 6.8:** 2 mitad WSJ etiquetado por MaxEnt (entrenado con 1 mitad WSJ) vs 2 mitad WSJ etiquetado con MaxEnt (entrenado con 1 mitad de WSJ + NFI)

MaxEnt(1WSJ+NFI) MaxEnt(1WSJ)	JJ	NN	VBN	RP	NNP	RB	VB	VBD	IN	NNS
<b>NN</b>	<b>611</b>	-	25	1	<b>344</b>	32	244	27	5	45
<b>JJ</b>	-	<b>513</b>	<b>254</b>	-	196	195	48	37	9	23
<b>VBD</b>	28	12	<b>494</b>	-	3	1	2	-	-	-
<b>IN</b>	9	3	-	<b>494</b>	18	<b>326</b>	-	2	-	-
<b>VBZ</b>	17	132	6	-	4	-	<b>283</b>	15	2	1
<b>VBN</b>	246	18	-	-	7	-	9	<b>257</b>	-	-
<b>WDT</b>	-	-	-	-	-	-	-	-	<b>255</b>	-
<b>RB</b>	149	13	1	222	70	-	19	-	166	-
<b>NNP</b>	211	215	8	-	-	26	24	4	5	87
<b>VBZ</b>	-	1	-	-	5	1	-	-	-	208

Aciertos: 622.115 (98,51 %)

Errores: 9.435 (1,49 %)

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	96.57 %
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	96.24 %
Stanford Tagger entrenado con el cuarto 1/4 de WSJ + NFI	96.53 %

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



- 96.27 % 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.

#### 6.0.4. 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 contruye la matriz de confusión:

**Cuadro 6.10:** *BNC original contra BNC etiquetado con TnT (entrenado con WSJ)*

<div>TnT(BNC)</div> <div>BNC</div>	NNP	JJ	NN	VBN	NNS	WRB	CD	NNPS	VBD	RB
<b>NN1</b>	<b>26585</b>	<b>5739</b>	-	146	732	3	52	13	157	378
<b>AJ0</b>	<b>8608</b>	-	2352	<b>3680</b>	50	1	34	24	327	1092
<b>DT0</b>	83	<b>7771</b>	208	-	1	-	-	-	-	988
<b>AV0</b>	1014	2028	<b>4277</b>	10	242	6	188	-	7	-
<b>NN0</b>	469	342	-	7	<b>3133</b>	-	<b>2601</b>	18	2	8
<b>CJS</b>	217	315	581	20	76	<b>2903</b>	-	-	43	1202
<b>NN2</b>	2472	75	690	2	-	-	77	<b>2578</b>	-	5
<b>VVN</b>	56	361	94	-	-	-	-	-	2365	1
<b>VVD</b>	54	235	87	2159	-	-	-	-	-	6
<b>AVP</b>	25	7	140	-	-	1	-	-	-	2070

Aciertos: 1.849.040 (92,47 %)

Errores: 150.675 (7,53 %)

**Cuadro 6.11:** *BNC original contra BNC etiquetado con TnT (entrenado con WSJ + NFI)*

<div>TnT(WSJ+NFI)</div> <div>BNC</div>	NNP	JJ	NN	NNS	VBN	WRB	NNPS	VBD	CD	WDT
<b>NN1</b>	<b>26413</b>	<b>4240</b>	-	663	101	1	8	118	56	1
<b>AJ0</b>	<b>8621</b>	-	1806	33	<b>3361</b>	1	21	291	20	1
<b>DT0</b>	75	<b>7689</b>	206	1	-	-	-	-	-	802
<b>AV0</b>	1068	1756	<b>4347</b>	226	16	2	-	4	125	4
<b>NN0</b>	447	474	-	<b>3366</b>	9	-	14	-	2120	-
<b>CJS</b>	222	252	705	101	18	<b>2903</b>	6	45	-	54
<b>NN2</b>	2328	85	795	-	-	-	<b>2542</b>	-	19	-
<b>VVN</b>	46	458	79	-	-	-	-	<b>2349</b>	-	-
<b>VVD</b>	53	217	76	-	2343	-	-	-	-	-
<b>CJT</b>	-	-	1	-	-	-	-	-	-	1857

Aciertos: 1.854.577 (92,74 %)

Errores: 145.138 (7,26 %)

Se puede observar que el rendimiento del etiquetador TnT entrenado con WSJ+NFI es un poco mejor (97,14%) que el rendimiento de TnT entrenado con WSJ (97,1 %). 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:** BNC original contra BNC etiquetado con TnT (entrenado con 2 mitad de WSJ)

TnT(2WSJ) BNC	NNP	JJ	VBN	NN	NNS	WRB	VBD	RB	VBG	NNPS
NN1	<b>27194</b>	<b>6142</b>	180	-	804	5	191	396	1735	9
AJ0	<b>8813</b>	-	<b>4421</b>	<b>3008</b>	69	1	371	1106	2191	22
DT0	82	<b>7802</b>	-	200	1	-	-	903	-	-
AV0	1008	2145	15	<b>4196</b>	314	5	6	-	41	2
NN0	483	650	8	-	<b>3410</b>	-	1	9	4	19
NN2	<b>2962</b>	76	1	734	-	-	-	19	-	2174
CJS	202	314	27	566	86	<b>2904</b>	45	1152	15	-
VVN	44	432	-	106	-	-	2707	1	7	-
VVD	45	258	2390	122	-	-	-	7	20	-
AVP	18	7	-	144	-	-	-	2335	-	-

Aciertos: 1.841.617 (92,09 %)

Errores: 158.098 (7,91 %)

**Cuadro 6.13:** BNC original contra BNC etiquetado con TnT (entrenado con 2 mitad de WSJ+NFI)

TnT(2WSJ+NFI) BNC	NNP	JJ	NN	VBN	NNS	WRB	NNPS	VBD	CD	WDT
NN1	<b>26728</b>	<b>4264</b>	-	101	663	1	8	119	61	1
AJ0	<b>8795</b>	-	1793	<b>3481</b>	34	-	21	295	22	-
DT0	86	<b>7648</b>	204	-	1	-	-	-	-	869
AV0	1074	1881	<b>4382</b>	15	216	1	-	4	130	4
NN0	445	424	-	9	<b>3380</b>	-	20	-	2177	-
CJS	229	260	738	15	102	<b>2903</b>	1	48	-	17
NN2	<b>2670</b>	76	795	-	-	-	2331	-	22	-
VVD	48	227	87	<b>2458</b>	-	-	-	-	-	-
VVN	41	481	82	-	-	-	-	2318	-	-
CJT	6	-	1	-	-	-	-	-	-	1857

Aciertos: 1.853.072 (92,67 %)

Errores: 146.643 (7,33 %)

**Cuadro 6.14:** BNC original contra BNC etiquetado con TnT (entrenado con 1 mitad de WSJ)

<div>TnT(1WSJ)</div> <div>BNC</div>	NNP	JJ	NN	VBN	NNS	WRB	NNPS	VBD	VBG	RB
NN1	<b>26286</b>	<b>6528</b>	-	187	783	3	36	169	1614	435
AJ0	<b>8503</b>	-	<b>2921</b>	<b>3713</b>	64	1	28	476	2027	1238
DT0	84	<b>7763</b>	232	-	1	-	-	-	-	976
AV0	1157	2031	<b>4455</b>	30	515	1	1	8	42	-
NN0	478	796	-	6	<b>3225</b>	-	21	2	14	4
CJS	193	235	626	24	77	<b>2903</b>	6	51	14	1165
NN2	2504	87	863	2	-	-	<b>2714</b>	-	-	1
VVN	57	615	95	-	-	-	-	2581	13	5
PRP	733	1415	2260	57	499	3	2	-	467	614
VVD	67	317	99	2126	1	-	-	-	27	5

Aciertos: 1.842.527 (92,14 %)

Errores: 157.188 (7,86 %)

**Cuadro 6.15:** BNC original contra BNC etiquetado con TnT (entrenado con 1 mitad de WSJ+NFI)

<div>TnT(1WSJ+NFI)</div> <div>BNC</div>	NNP	JJ	NN	NNS	VBN	WRB	NNPS	VBD	CD	WDT
NN1	<b>26360</b>	<b>4345</b>	-	677	107	2	14	127	63	3
AJ0	<b>8729</b>	-	1808	35	<b>3290</b>	1	20	303	24	1
DT0	91	<b>7655</b>	210	1	-	-	-	-	-	870
AV0	1225	1630	<b>4409</b>	215	18	-	1	4	143	4
NN0	442	501	-	<b>3321</b>	8	-	21	-	2174	-
CJS	221	256	696	100	18	<b>2903</b>	6	47	-	63
NN2	2327	85	815	-	-	-	<b>2568</b>	-	17	-
VVD	60	218	76	-	<b>2429</b>	-	-	-	-	-
VVN	62	481	84	-	-	-	-	2381	-	-
CJT	-	-	1	-	-	-	-	-	-	1867

Aciertos: 1.853.701 (92,70 %)

Errores: 146.014 (7,30 %)

Se puede apreciar una leve mejoría en el porcentaje de etiquetas acertadas para el modelo que incorpora NFI; 92,09 % contra 92,67 % y 92,14 % contra 92,7 % 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 presentan las matrices de confusión entre las mitades de WSJ etiquetado con TnT entrenado con la mitad restante con y sin NFI.

**Cuadro 6.16:** BNC etiquetado por TnT (entrenado con 1 mitad WSJ) vs BNC etiquetado con TnT (entrenado con 1 mitad de WSJ + NFI)

TnT(1WSJ+NFI) TnT(1WSJ)	NN	JJ	VBN	VB	NNP	NNS	VBD	VBP	VBG	RB
<b>JJ</b>	<b>5144</b>	-	1092	693	<b>1701</b>	128	283	82	474	913
<b>NN</b>	-	<b>3630</b>	65	<b>2399</b>	<b>2215</b>	390	58	417	1168	661
<b>VBD</b>	115	366	<b>2487</b>	93	47	1	-	21	-	7
<b>VB</b>	<b>1960</b>	291	131	-	230	1	97	1290	5	103
<b>VBZ</b>	12	65	-	5	38	<b>1931</b>	4	3	-	16
<b>NNP</b>	<b>1870</b>	1178	42	270	-	618	21	32	87	152
<b>VBN</b>	155	1512	-	70	78	1	<b>1865</b>	16	5	12
<b>VBP</b>	641	125	24	1343	41	2	38	-	-	27
<b>RB</b>	491	1215	4	238	279	5	3	9	-	-
<b>VBG</b>	1028	1198	5	21	188	5	23	7	-	5

Aciertos: 1.938.000 (96,91 %)

Errores: 61.726 (3,09 %)

**Cuadro 6.17:** BNC etiquetado por TnT (entrenado con 2 mitad WSJ) vs BNC etiquetado con TnT (entrenado con 2 mitad de WSJ + NFI)

TnT(2WSJ+NFI) TnT(2WSJ)	NN	JJ	VBN	VB	NNP	NNS	VBD	VBP	RP	VBG
<b>JJ</b>	<b>5020</b>	-	846	433	1470	73	260	77	2	324
<b>NN</b>	-	<b>4138</b>	67	<b>2355</b>	<b>2027</b>	180	81	455	3	932
<b>VBD</b>	133	304	<b>2442</b>	46	38	2	-	25	-	-
<b>NNP</b>	<b>2220</b>	1337	23	242	-	708	10	22	-	85
<b>VB</b>	<b>2193</b>	359	116	-	140	3	189	1602	2	4
<b>VBN</b>	185	<b>2091</b>	-	53	28	-	<b>1882</b>	18	-	2
<b>VBZ</b>	16	31	-	3	51	<b>1998</b>	-	10	-	-
<b>VBP</b>	599	50	10	1271	37	1	69	-	-	-
<b>RB</b>	539	1146	-	467	237	26	1	24	1177	-
<b>VBG</b>	1166	1172	-	5	200	3	10	-	-	-

Aciertos: 1.938.152 (96,92 %)

Errores: 61.574 (3,08 %)

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	92.62 %
TnT entrenado con el segundo 1/4 de WSJ	91.74 %
TnT entrenado con el segundo 1/4 de WSJ + NFI	92.63 %
TnT entrenado con el tercer 1/4 de WSJ	91.64 %
TnT entrenado con el tercer 1/4 de WSJ + NFI	92.62 %
TnT entrenado con el cuarto 1/4 de WSJ	91.64 %
TnT entrenado con el cuarto 1/4 de WSJ + NFI	92.58 %

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
- 92.55 % 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.

### 6.0.5. 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:** BNC original contra BNC etiquetado con MaxEnt (entrenado con WSJ)

MaxEnt(BNC) BNC	NNP	JJ	NN	VBN	NNS	WRB	RB	CD	VBG	NNPS
<b>NN1</b>	<b>26141</b>	<b>4045</b>	-	115	533	-	253	16	1143	7
<b>AJ0</b>	<b>8675</b>	-	<b>2860</b>	<b>3276</b>	30	-	1033	3	2054	12
<b>DT0</b>	119	<b>8132</b>	192	-	5	-	443	-	-	-
<b>AV0</b>	982	2314	<b>3753</b>	159	236	-	-	160	85	4

**Cuadro 6.19:** BNC original contra BNC etiquetado con MaxEnt (entrenado con WSJ)

MaxEnt(BNC) BNC	NNP	JJ	NN	VBN	NNS	WRB	RB	CD	VBG	NNPS
<b>NN0</b>	567	1115	-	19	<b>3132</b>	-	10	2605	2	5
<b>NN2</b>	<b>2982</b>	90	750	-	-	-	6	47	-	1959
<b>CJS</b>	60	334	247	57	104	<b>2901</b>	522	1	35	2
<b>AVP</b>	43	17	137	-	-	-	2691	-	-	-
<b>UNC</b>	1754	255	540	9	199	-	2	426	1	18
<b>CJT</b>	-	1	-	-	-	-	-	-	-	-

Aciertos: 1.856.979 (92,86 %)

Errores: 142.739 (7,14 %)

**Cuadro 6.20:** BNC original contra BNC etiquetado con MaxEnt (entrenado con WSJ + NFI)

MaxEnt(WSJ+NFI) BNC	NNP	JJ	NN	CD	NNS	VBN	WRB	RB	NNPS	VBG
<b>NN1</b>	<b>26206</b>	<b>3864</b>	-	22	663	108	-	277	1	1166
<b>AJ0</b>	<b>8263</b>	-	2099	4	25	<b>3145</b>	-	876	12	1707
<b>DT0</b>	109	<b>7836</b>	188	-	4	-	-	793	-	-
<b>AV0</b>	840	2210	<b>3640</b>	195	279	123	-	-	2	70
<b>NN0</b>	469	863	-	<b>3222</b>	<b>3146</b>	6	-	-	8	9
<b>CJS</b>	102	374	357	-	147	37	<b>2901</b>	776	2	39
<b>NN2</b>	<b>2643</b>	68	783	74	-	1	-	8	1844	-
<b>AVP</b>	39	6	140	-	-	-	-	2101	-	-
<b>PRP</b>	497	1680	887	1	572	115	-	711	1	624
<b>VVN</b>	76	461	87	-	1	-	-	1	-	7

Aciertos: 1.859.888 (93,01 %)

Errores: 139.830 (6,99 %)

Se puede observar que el rendimiento del etiquetador entrenado con WSJ es un poco mejor (93,01 %) que cuando es entrenado con WSJ + NFI (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:** BNC original contra BNC etiquetado con MaxEnt (entrenado con 2 mitad de WSJ)

MaxEnt(2WSJ) BNC	NNP	JJ	NN	NNS	VBN	WRB	RB	CD	VBG	NNPS
NN1	<b>26061</b>	<b>4689</b>	-	620	97	-	248	6	1296	8
AJ0	<b>8570</b>	-	<b>3574</b>	49	<b>3001</b>	-	1100	6	2190	10
DT0	133	<b>8068</b>	224	1	-	-	478	1	-	-
AV0	985	2428	<b>3611</b>	475	143	-	-	164	77	9
NN0	554	1404	-	<b>3128</b>	7	-	14	2633	7	6
NN2	<b>2984</b>	146	855	-	-	-	4	46	-	2054
CJS	98	210	237	146	34	<b>2901</b>	568	3	21	2
AVP	47	6	152	-	-	-	2793	-	-	-
UNC	1763	268	454	212	3	-	3	518	1	20
CJT	-	1	-	-	-	-	-	-	-	-

Aciertos: 1.851.792 (92,60 %)

Errores: 147.926 (7,40 %)

**Cuadro 6.22:** BNC original contra BNC etiquetado con MaxEnt (entrenado con 2 mitad de WSJ+NFI)

MaxEnt(2WSJ+NFI) BNC	NNP	JJ	NN	CD	NNS	VBN	WRB	RB	NNPS	VBG
NN1	<b>26036</b>	<b>3867</b>	-	17	657	109	-	285	3	1179
AJ0	<b>8130</b>	-	2151	5	24	<b>3011</b>	-	897	13	1653
DT0	108	<b>7742</b>	215	-	3	-	-	867	-	-
AV0	875	2220	<b>3538</b>	203	276	129	-	-	1	68
NN0	452	866	-	<b>3240</b>	<b>3138</b>	8	-	-	11	8
CJS	123	327	333	2	158	58	<b>2901</b>	908	2	17
NN2	<b>2513</b>	75	800	81	-	1	-	10	1922	-
AVP	40	5	142	-	-	-	-	2014	-	-
VVD	75	203	89	-	-	1573	-	9	-	13
PRP	495	1528	758	2	689	109	-	723	-	587

Aciertos: 1.859.947 (93,01 %)

Errores: 139.771 (6,99 %)

**Cuadro 6.23:** BNC original contra BNC etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

MaxEnt(1WSJ) BNC	NNP	JJ	NN	VBN	NNS	WRB	RB	CD	VBG	VBD
NN1	<b>27101</b>	<b>4519</b>	-	146	550	-	241	7	1369	116
AJ0	<b>9043</b>	-	<b>3838</b>	<b>3837</b>	43	-	1025	3	2238	296
DT0	128	<b>8324</b>	185	-	1	-	461	-	-	-
AV0	1071	2583	<b>3445</b>	141	311	-	-	170	71	78
NN2	<b>3415</b>	83	885	-	-	-	7	41	-	1

**Cuadro 6.23:** BNC original contra BNC etiquetado con MaxEnt (entrenado con 1 mitad de WSJ)

MaxEnt(1WSJ) BNC	NNP	JJ	NN	VBN	NNS	WRB	RB	CD	VBG	VBD
<b>NN0</b>	573	955	-	24	<b>3189</b>	-	5	2732	3	8
<b>CJS</b>	82	412	267	54	106	<b>2901</b>	400	1	23	63
<b>AVP</b>	21	26	140	-	-	-	2859	-	-	-
<b>VVN</b>	85	464	126	-	1	-	1	-	1	1986
<b>UNC</b>	1757	181	509	9	242	-	3	442	1	5

Aciertos: 1.848.799 (92,45 %)

Errores: 150.919 (7,55 %)

**Cuadro 6.24:** BNC original contra BNC etiquetado con MaxEnt (entrenado con 1 mitad de WSJ+NFI)

MaxEnt(1WSJ+NFI) BNC	NNP	JJ	NN	CD	VBN	NNS	WRB	RB	VBD	NNPS
<b>NN1</b>	<b>26776</b>	<b>3786</b>	-	20	109	684	-	282	94	1
<b>AJ0</b>	<b>8368</b>	-	2113	3	<b>3249</b>	32	-	853	215	10
<b>DT0</b>	108	<b>7768</b>	192	-	-	1	-	883	-	-
<b>AV0</b>	950	2360	<b>3475</b>	194	90	325	-	-	39	2
<b>NN0</b>	473	752	-	<b>3302</b>	2	<b>3193</b>	-	-	4	4
<b>CJS</b>	151	409	348	-	33	150	<b>2901</b>	856	39	1
<b>NN2</b>	<b>2831</b>	66	797	74	1	-	-	8	1	1636
<b>AVP</b>	32	6	139	-	-	-	-	2080	-	-
<b>VVN</b>	77	492	94	-	-	1	-	1	1756	-
<b>PRP</b>	605	1613	925	-	129	741	-	802	122	-

Aciertos: 1.857.971 (92,91 %)

Errores: 141.747 (7,09 %)

Se puede apreciar una leve mejoría en el porcentaje de etiquetas acertadas; 92,6 % contra 93,01 % y 92,45 % contra 92,91 % 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:** *BNC etiquetado por MaxEnt (entrenado con 2 mitad WSJ) vs BNC etiquetado con MaxEnt (entrenado con 2 mitad de WSJ + NFI)*

MaxEnt(2WSJ+NFI) MaxEnt(2WSJ)	JJ	NN	NNP	VBN	NNS	VB	RP	VBG	RB	VBD
<b>NN</b>	<b>4691</b>	-	<b>2413</b>	152	392	<b>1859</b>	11	1579	813	192
<b>JJ</b>	-	<b>3864</b>	1168	1524	85	356	2	336	1233	201
<b>NNP</b>	<b>1688</b>	<b>2727</b>	-	80	938	308	-	117	267	61
<b>VBD</b>	310	107	17	<b>2345</b>	-	135	-	6	21	-
<b>VBZ</b>	23	16	35	1	<b>1922</b>	6	-	-	24	9
<b>IN</b>	146	247	203	17	32	443	<b>1854</b>	75	1408	102
<b>VBP</b>	190	1075	35	28	15	<b>1693</b>	-	4	64	123
<b>VBN</b>	1351	99	69	-	13	66	-	12	20	1362
<b>VBG</b>	1220	1166	139	11	-	25	-	-	3	2
<b>VB</b>	300	1125	295	135	11	-	-	17	127	88

Aciertos: 1.933.574 (96,69 %)

Errores: 66.144 (3,31 %)

**Cuadro 6.26:** *BNC etiquetado por MaxEnt (entrenado con 1 mitad WSJ) vs BNC etiquetado con MaxEnt (entrenado con 1 mitad de WSJ + NFI)*

MaxEnt(1WSJ+NFI) MaxEnt(1WSJ)	JJ	NN	NNP	VBN	NNS	VB	RP	RB	VBG	VBD
<b>NN</b>	<b>5058</b>	-	<b>2370</b>	127	414	<b>1895</b>	15	372	1438	115
<b>JJ</b>	-	<b>3812</b>	1138	1148	80	360	8	1331	272	130
<b>NNP</b>	<b>2055</b>	<b>2670</b>	-	44	957	280	-	218	104	24
<b>VBD</b>	384	188	31	<b>2361</b>	-	186	-	21	28	-
<b>VBZ</b>	20	63	35	-	<b>2074</b>	37	-	43	1	12
<b>VBN</b>	<b>1973</b>	166	73	-	5	62	-	30	15	1366
<b>IN</b>	169	302	246	14	49	344	<b>1807</b>	1485	63	18
<b>VBP</b>	232	1021	27	27	12	1510	-	77	5	150
<b>RB</b>	1218	453	209	13	34	241	1081	-	21	13
<b>VB</b>	302	1204	269	145	20	-	-	139	23	107

Aciertos: 1.933.672 (96,70 %)

Errores: 66.046 (3,30 %)

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
------------	------------------------

**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	92.92 %
Stanford Tagger entrenado con el segundo 1/4 de WSJ	92.10 %
Stanford Tagger entrenado con el segundo 1/4 de WSJ + NFI	92.91 %
Stanford Tagger entrenado con el tercer 1/4 de WSJ	92.14 %
Stanford Tagger entrenado con el tercer 1/4 de WSJ + NFI	92.89 %
Stanford Tagger entrenado con el cuarto 1/4 de WSJ	91.98 %
Stanford Tagger entrenado con el cuarto 1/4 de WSJ + NFI	92.83 %

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] Thorsten Brants, TnT: a statistical part-of-speech tagger, Proceedings of the sixth conference on Applied natural language processing, p.224-231, April 29-May 04, 2000, Seattle, Washington
- [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] Reference Guide for the British National Corpus (World Edition) edited by Lou Burnard, October 2000
- [7] 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
- [8] Brown K. (Editor) 2005. Encyclopedia of Language and Linguistics 2nd Edition. Oxford: Elsevier.
- [9] 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. The BNC Handbook Exploring the British National Corpus with SARA Guy Aston and Lou Burn