SEP
SECRETARÍA DE EDUCACIÓN PÚBLICA

TECNOLÓGICO NACIONAL DE MÉXICO

Instituto Tecnológico de Ciudad Madero

División de Estudios de Posgrado e Investigación

Maestría en Ciencias de la Computación

"POR MI PATRIA Y POR MI BIEN"

TESIS

COMPLEMENTACIÓN DE UN ANALIZADOR SINTÁCTICO DEL IDIOMA ESPAÑOL PARA LA VERIFICACIÓN DE CONGRUENCIA DE LOS ACCIDENTES GRAMATICALES

Que para obtener el grado de
Maestro en Ciencias de la Computación

Presenta
Ing. Lemuel Rodríguez Moya

Director de Tesis
Dr. José Antonio Martínez Flores

Co-Director de Tesis
Dr. Rodolfo Abraham Pazos Rangel

Cd. Madero, Tamaulipas, México **Mayo 2017**

"Año del Centenario de la Promulgación de la Constitución Política de los Estados Unidos Mexicanos"

Cd. Madero, Tamps; a 22 de Mayo de 2017.

OFICIO No.: U5.088/17
AREA: DIVISIÓN DE ESTUDIOS
DE POSGRADO E INVESTIGACIÓN
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN DE TESIS

ING. LEMUEL RODRÍGUEZ MOYA
NO. DE CONTROL G09070427
P R E S E N T E

Me es grato comunicarle que después de la revisión realizada por el Jurado designado para su examen de grado de Maestría en Ciencias de la Computación, el cual está integrado por los siguientes catedráticos:

PRESIDENTE :	DRA.	MARÍA LUCILA MORALES RODRÍGUEZ
SECRETARIO :	DR.	RODOLFO ABRAHAM PAZOS RANGEL
VOCAL :	DR.	JOSÉ ANTONIO MARTÍNEZ FLORES
SUPLENTE	DR.	JUAN JAVIER GONZÁLEZ BARBOSA
DIRECTOR DE TESIS:	DR.	JOSÉ ANTONIO MARTÍNEZ FLORES
CO-DIRECTORA DE TESIS:	DR.	RODOLFO ABRAHAM PAZOS RANGEL

Se acordó autorizar la impresión de su tesis titulada:

"COMPLEMENTACIÓN DE UN ANALIZADOR SINTÁCTICO DEL IDIOMA ESPAÑOL PARA LA VERIFICACIÓN DE CONGRUENCIA DE LOS ACCIDENTES GRAMATICALES"

Es muy satisfactorio para la División de Estudios de Posgrado e Investigación compartir con Usted el logro de esta meta.

Espero que continúe con éxito su desarrollo profesional y dedique su experiencia e inteligencia en beneficio de México.

A T E N T A M E N T E
"POR MI PATRIA Y POR MI BIEN"®

DRA. ADRIANA ISABEL REYES DE LA TORRE
JEFA DE LA DIVISIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA
TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE CIUDAD MADERO
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

c.c.p.- Archivo
Minuta

AIRT'NÚCO 'jar

Declaración de Originalidad

Declaro y prometo que este documento de tesis es producto de mi trabajo original y que no infringe los derechos de terceros, tales como derechos de publicación, derechos de autor, patentes y similares.

Además, declaro que las citas textuales que he incluido las cuales aparecen entre comillas y en los resúmenes que he realizado de publicaciones ajenas, indico explícitamente los datos de los autores y publicaciones.

Además, en caso de infracción a los derechos de terceros derivados de este documento de tesis, acepto la responsabilidad de la infracción y relevo de esta a mi director y codirector de tesis, así como al Instituto Tecnológico de Ciudad Madero y sus autoridades.

_____

Ing. Lemuel Rodríguez Moya

Agradecimientos

Le agradezco a todas las personas que una manera u otra apoyaron para que este logro se fuera posible.

Primero le agradezco A mis padres Lourdes Moya Cobo y Eugenio Rodríguez González por su apoyo incondicional brindado en estos años.

También a los miembros de mi comité tutorial de la tesis: Dra. María Lucila Morales Rodríguez , Dr. Rodolfo Abraham Pazos Rangel, Dr. José Antonio Martínez Flores y Dr. Juan Javier González Barbosa.

En especial a mi asesor el Dr. José Antonio Martínez por todo el apoyo brindado en este proyecto de tesis y por encaminarme a lo largo del desarrollo de este proyecto.

Al Dr. Rodolfo Abraham Pazos Rangel por sus enseñanzas y ayuda en el desarrollo de mis estudios en este posgrado.

A la Dra. María Lucila Morales Rodríguez por sus críticas constructivas, observaciones y sugerencias, que fueron de ayuda en el desarrollo de esta tesis.

A mis compañeros de generación Enith Martínez Cruz, Ricardo Rojas Hernández, Leonor Hernández Ramírez por su amistad en estos años.

Resumen

En la sociedad actual la mayor parte de la información se encuentra almacenada en bases de datos. Muchas estrategias se han desarrollado para hacer uso de esta información y transformarla en conocimiento útil para el desarrollo de la sociedad. Una de las estrategias actuales son las interfaces de lenguaje natural de consulta a bases de datos dado su facilidad de uso en conjunto con las bases de datos. Una parte importante de estas interfaces es el modulo traductor, encargado de procesar el lenguaje natural para transformarlo en una consulta en lenguaje SQL. En este trabajo se considera integrar un analizador sintáctico a una Interfaz, desafortunadamente éste no detecta las incongruencias de accidentes gramaticales entre palabras de una oración, ocasionando interpretaciones erróneas del sentido de la oración. Por lo tanto se busca agregar al analizador sintáctico la detección de las incongruencias de accidentes gramaticales para que se pueda dar la concordancia entre palabras de una oración y que esto apoye al análisis el sentido de una oración.

Palabras clave: PLN, congruencia de accidentes gramaticales, concordancia de palabras**.**

Contenido

Lista de figuras

Lista de tablas

Capítulo 1 Introducción

Desde los años 50 existe preocupación en la comunidad científica por desarrollar mecanismos de Procesamiento de Lenguaje Natural (PLN). El trabajo "Computing machinery and intelligence" publicado en 1950 por Alan Turing en la revista *Mind* proponía el test de Turing como criterio de inteligencia (Turing, 1950), en el cual se buscaba que una máquina tuviera un comportamiento inteligente similar al de un humano. En dicho trabajo se colocó a un humano a conversar con una máquina por medio textual. Si bien su trabajo no buscaba que la máquina procesara el lenguaje natural, se midió su capacidad para generar respuestas similares a las de una persona. Su trabajo se realizó en el área de inteligencia artificial, éste se puede considerar como una primera aproximación al área de lenguaje natural donde se pretende que las máquinas puedan comunicarse de manera autónoma y precisa con los seres humanos.

En el mismo período científicos como Georgetown comenzaron a incursionar en el área de PLN (Hutchins, 2005). El trabajo de Georgetown en 1954 involucró traducción automática de más de sesenta oraciones del ruso al inglés. La comunidad científica de la época presumió que el problema de desarrollo de la traducción automática sería un problema que en pocos años sería resuelto, sin embargo, la traducción automática continua lanzando muchas incógnitas aún en los días actuales. Teorías lingüísticas como las de Noam Chomsky (por ejemplo, la Gramática Transformacional generativa) Chomsky, N. (1980a) con sus fundamentos teóricos desalentaron el tipo de lingüística de corpus, basada en el aprendizaje de máquina para el procesamiento del lenguaje dado la complejidad del trabajo. A pesar de esto, la creciente demanda de información asociada con el aumento constante del poder de cómputo de las computadoras modernas ha despertado la necesidad de elaborar sistemas para el almacenamiento y manejo de información. El procesamiento correcto del lenguaje natural puede explotar la información generada por los seres humanos y transformarla en nueva información, útil en los procesos de desarrollo y mantenimiento de la sociedad.

En los días actuales existen muchas herramientas mediante las cuales se puede realizar el PLN. Las Interfaces de Lenguaje Natural de Consulta a Bases de Datos (ILNBD) son una propuesta de solución al manejo y procesamiento de la información almacenada en bases de datos; la facilidad de operación de estos sistemas es de suma importancia para garantizar que cualquier persona pueda usarlos. Además de lo anterior, es necesario que estos sistemas produzcan buenas salidas y entreguen resultados correctos y confiables. De entre los grandes problemas actuales de las ILNBDs están los errores de traducción, las ambigüedades y la imprecisión de sus respuestas. Los problemas de este tipo en estos sistemas pueden generar confusiones de interpretación por parte de los usuarios y consecuentemente generar errores en la toma de decisiones. El PLN es actualmente el campo que

combina las tecnologías de la ciencia computacional (como la inteligencia artificial, el aprendizaje automático o la inferencia estadística) con la lingüística aplicada, buscando hacer posible la comprensión y el procesamiento por ordenador, de información expresada en lenguaje humano para determinadas tareas. Una parte del PLN es el análisis léxico-sintáctico, se han realizado muchos trabajos para hacerlo más robusto y preciso, dado que este análisis es muy amplio, todavía existe interés en explorar éste. En este trabajo se elaboró un módulo que detecta errores en oraciones de la lengua española relacionados con los accidentes gramaticales.

1.1 Objetivos

1.1.1 General

Complementar un analizador sintáctico del idioma español con algoritmos capaces de detectar errores en oraciones de la lengua española relacionados con los accidentes gramaticales de género y número.

1.1.2 Específicos

- Identificar los accidentes gramaticales que se presentan en la lengua española.
- Identificar y construir reglas de concordancias entre palabras de una oración.
- Estudiar el analizador léxico de la ILNBD versión [Aguirre, 2014] y su lexicón.
- Ampliar el lexicón creado en el trabajo de [Aguirre, 2014] con palabras de la RAE.
- Construir arboles sintácticos que representen el resultado de la reducción sintáctica.
- Desarrollar algoritmos que complementen el analizador sintáctico propuesto por [Mellado, 2014] para detectar la concordancia entre palabras de una oración.
- Evaluar la funcionalidad del nuevo analizador sintáctico con casos de prueba variados.

1.2 Justificación

En los días actuales prácticamente toda la información ya sea de empresas, bancos, tiendas y organismos gubernamentales se almacena en bases de datos. El correcto manejo de la información en las bases de datos y el desarrollo de mecanismos que permitan manipular dicha información como las ILNBDs, es de suma importancia para el perfecto funcionamiento de la sociedad. Por lo tanto es imprescindible que el funcionamiento de estos sistemas que manipulan la información

almacenada en grandes bases de datos sea correcto y fiable, ya que la información que manejan constituye los indicios del desarrollo del hombre en la tierra.

El almacenamiento, la conservación y el aprovechamiento de la información es muy importante para que se preserve la historia y el funcionamiento de la vida humana en la modernidad, ya que sin esto, todas las operaciones que realiza el ser humano, no sólo en el ámbito de la informática, sino que también en toda la estructura de la vida en las ciudades (electricidad, industrias, sistemas, etcétera) serían imposibles de realizar. Por medio de este trabajo se busca mejorar el analizador léxico de (Aguirre, 2014)homogenizando las etiquetas del lexicón; el analizador sintáctico de (Mellado, 2014)creando el(los) árbol(es) sintáctico(s) de una oración; lo que permitirá eficientar el analizador semántico (siguiente fase de análisis de una oración) de la Interfaz Ver. (Aguirre, 2014) Como resultado de este trabajo se espera reducir el número de errores generados por las diferentes excepciones de las reglas de la gramática española y la posibilidad de detectar errores introducidos por parte del usuario en la oración sometida a análisis.

1.3 Antecedentes del Proyecto

En el trabajo **“Implementación de un Analizador Sintáctico del Idioma Español para una Interfaz de lenguaje natural a Base de Datos”** por (Mellado, 2014), se propuso un analizador sintáctico en el cual buscó evaluar la oración de una consulta realizando un análisis sintáctico más profundo usando un mayor número de características gramaticales del lenguaje español. Mediante un estudio extenso de la gramática española se elaboraron estructuras gramaticales reconocidas por la RAE y se le asignaron etiquetas a dichas estructuras gramaticales, con el fin de usarlas como símbolos terminales y no terminales para facilitar la reducción sintáctica. Además de esto se buscó crear un número menor de reglas gramaticales para este analizador. Este trabajo revisó un extracto de la gramática española, el autor verificó minuciosamente la correspondencia de información presentada por la RAE. Tras un amplio bosquejo se observó que para construir estructuras gramaticales en la gramática española como los sintagmas, se requieren de una a tres categorías gramaticales en la mayoría de los casos y que en casos donde se requirieran más elementos, dichos elementos son sintagmas construidos a partir de una base de al menos tres elementos gramaticales. El analizador sintáctico versión Mellado trabaja explorando todas las variaciones que pueda tener el sentido de una oración mediante el desarrollo de la reducción sintáctica. Este analizador sintáctico no verifica la congruencia de los accidentes gramaticales de las oraciones, y puede aceptar como correctas oraciones que sintácticamente están bien pero son incongruentes. Este proyecto busca implementar algoritmos que permitan detectar estos errores.

El trabajo “**Modelo Semánticamente Enriquecido de Bases de Datos para su Explotación por Interfaces de Lenguaje Natural**” por (Aguirre, 2014), tuvo como objetivo desarrollar una

interfaz de lenguaje natural encargada de traducir consultas de lenguaje natural a SQL. La propuesta de Aguirre para la ILNBD fue el desarrollo de una arquitectura basada en capas (**Figura 1.1**) y el diseño de un diccionario de información semántica (DIS). El núcleo de la interfaz propuesta por Aguirre y desarrollada en el ITCM consta de tres capas principales como vemos en la **Figura 1.2**.

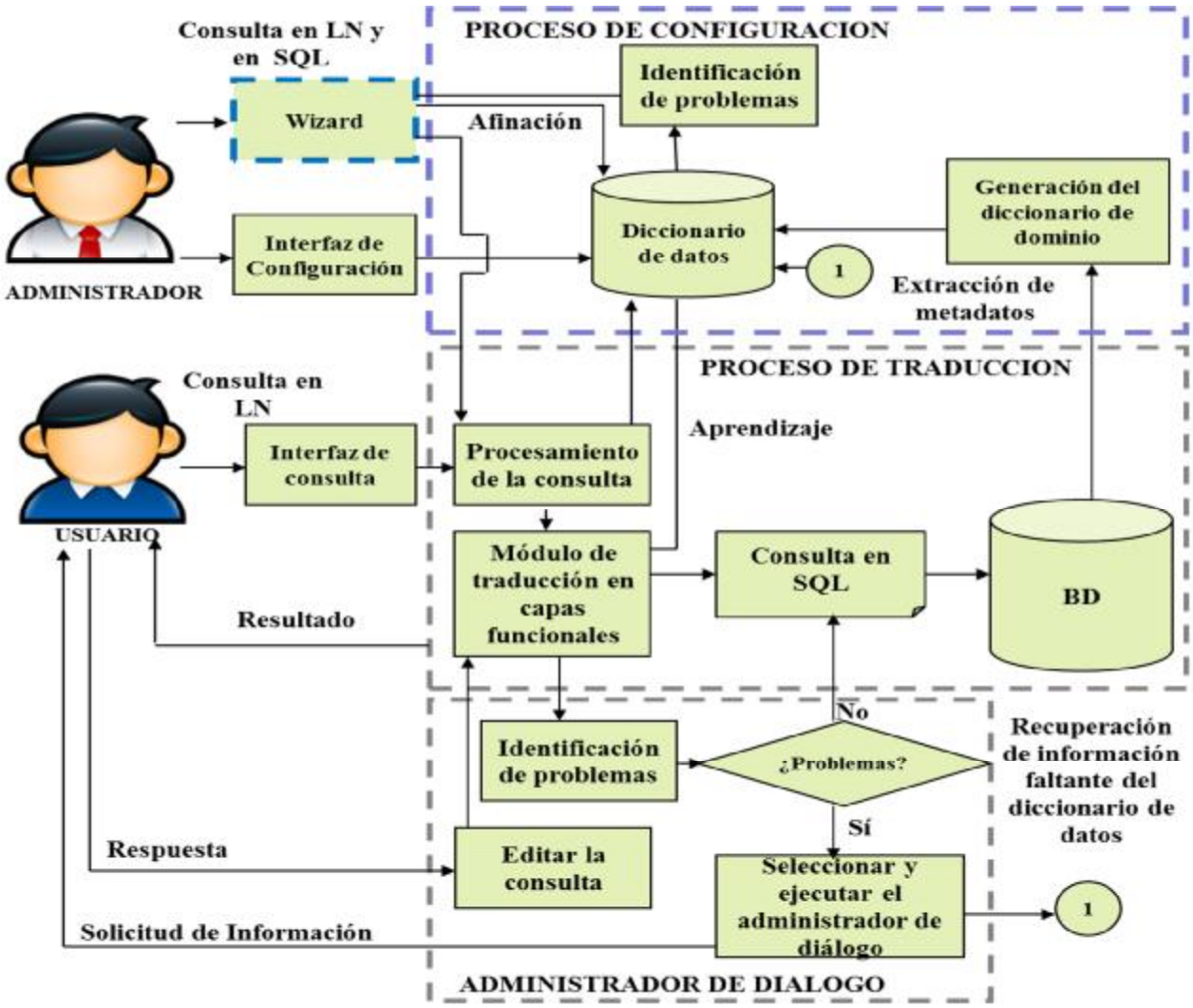


Figura 1.1 Arquitectura general de la ILNBD

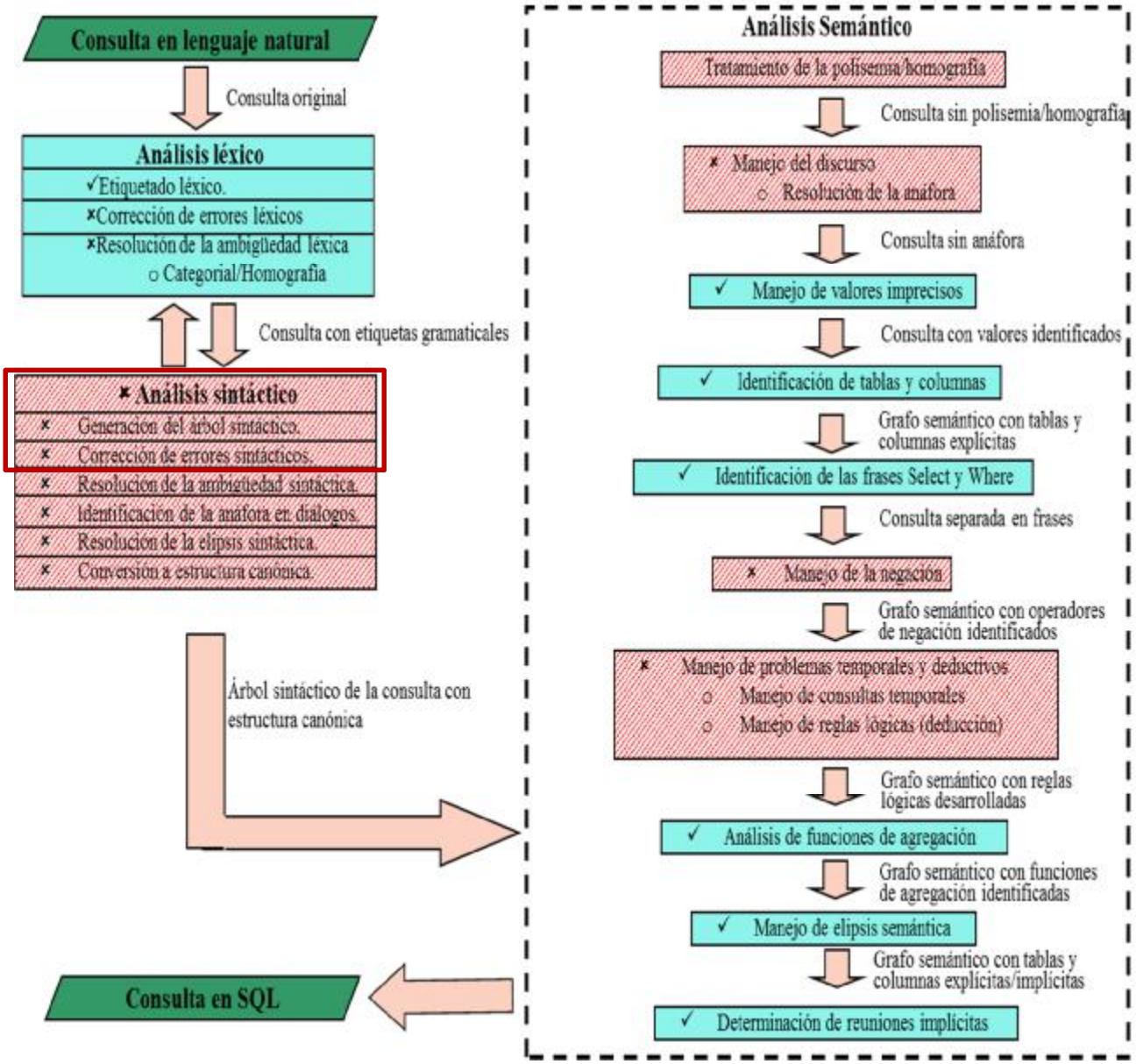


Figura 1.2 Capas de funcionalidad de la ILNBD

La primera capa se denomina análisis léxico y tiene como finalidad etiquetar todas las palabras de una consulta con su categoría gramatical, en caso en que alguna palabra no sea localizada dentro del lexicón, se infiere que es un valor de búsqueda, para lograr lo anterior Aguirre construyó un lexicón, el cual contiene la mayoría de los verbos de la lengua española, incluyendo su conjugación. La siguiente capa, el análisis sintáctico está compuesta por una heurística que realiza un análisis para obtener una sola categoría gramatical por palabra, ya que existen palabras que tienen más de una categoría gramatical. En este trabajo se ignoran palabras irrelevantes. La tercera capa consta del análisis semántico, el cual se encarga de identificar tablas y columnas en base a la información almacenada en el DIS. Dicha ILNBD logra obtener un porcentaje de acierto de 90% de acierto. Uno

de los objetivos de este proyecto fue el enriquecimiento del lexicón desarrollado por [Aguirre, 2014] con accidentes gramaticales de las palabras. El recuadro rojo de la **Figura 1.2** delimita el área del proyecto en la cual se trabajó.

1.4 Alcances y Limitaciones del Proyecto

1.4.1 Alcances

Los alcances que se pretenden lograr en este proyecto se enlistan a continuación.

- El análisis de las oraciones se realiza en lenguaje natural, determinando si la oración posee alguna de las posibles formas gramaticales permitidas para la formación de oraciones en español.
- La verificación de concordancia entre palabras detecta sólo los accidentes gramaticales de una parte del español, debido a la diversidad de palabras que lo conforman, quedando disponible para futuras actualizaciones o mejoras.
- El incremento o modificación de reglas de concordancia es flexible y dinámico al estar independientes del código, lo que facilita la mejora de la verificación de concordancia entre palabras del analizador sintáctico.

1.4.2 Limitaciones

Las limitaciones que se consideraron en este proyecto son las siguientes:

- Debido a que el objetivo de esta tesis es mejorar el funcionamiento de un analizador sintáctico que se integrará a una ILNBD específica, se centra en procesar/analizar oraciones que se encuentran en corpus de bases de datos.
- Este analizador no busca corregir la ortografía; si se recibe una palabra, éste buscar procesar la entrada y generar un análisis de la misma, indicando posibles errores.
- El analizador sintáctico sólo trabajar con el idioma español de México.
- Se pretende trabajar con oraciones con máximo dos verbos.
- No se pretende eliminar por completo la ambigüedad sintáctica.
- No se pretenden abarcar todos los accidentes gramaticales, sólo los relevantes para la información que maneja la ILNBD, específicamente los de género y número.
- El analizador no será integrado a una interfaz de lenguaje natural como parte de este proyecto de tesis.

Capítulo 2 Estado del Arte

2.1 Trabajos enfocados al procesamiento de lenguaje natural

El análisis sintáctico es una parte importante del PLN que verifica las relaciones de concordancia y la jerarquía que deben llevar las palabras dentro de la oración. Actualmente se espera que los analizadores sintácticos alcancen los mayores niveles de precisión y robustez. Muchos lenguajes como el inglés, portugués, francés, danés y noruego han alcanzado los mayores niveles de precisión y robustez (Bick, 2006). En el caso del idioma español dado que éste tiene una amplia variedad de reglas gramaticales, el desempeño de estos analizadores sintácticos es menor. La precisión de los analizadores sintácticos de español varía entre los 70% y 90% (Gelbukh, 2010). Algunos trabajos en el área se describen a continuación

En el CII-IPN se desarrolló el trabajo **"Análisis sintáctico conducido por un diccionario de patrones de manejo sintáctico para lenguaje español"** por (Galicia S. , 2000). En este trabajo se tiene un compendio de reglas sintácticas las cuales están separadas del código fuente del analizador sintáctico. Este trabajo posee una gramática de dependencias MTT con 150 reglas gramaticales, además de un sistema de desambiguación de oraciones. El sistema en pruebas realizadas obtuvo 53 oraciones procesadas correctamente de un total de 100 oraciones con un promedio de colocación del 25 %. Además se hace uso de algunas estrategias para la detección de errores de accidentes gramaticales. Esta detección se hace mediante el etiquetado de algunos accidentes como los de género y número. Estas etiquetas se usan para clasificar las palabras con relación a las variaciones que estas puedan experimentar, por ejemplo, el adjetivo *bajo* se etiqueta de la siguiente manera **bajo <NCMS000>.** La etiqueta contiene las siglas M y S, indicando el género *masculino* y el número *singular* de la palabra bajo.

El trabajo **"Reconocedor de comandos en español"**, por (Del Rosario & Hernández, 2004) tuvo como objetivo el desarrollo de un sistema reconocedor de comandos en lenguaje natural. El reconocedor de comandos contiene como procesos principales las fases de análisis sintáctico y de análisis semántico y la ejecución del comando reconocido. El sistema primero realiza el análisis sintáctico en el cual se identifican comandos atómicos conjuntivos, disyuntivos y condicionales. Una vez que un comando se acepta por el analizador sintáctico, se mapea con el analizador semántico a una función a ejecutar. La interpretación de una oración imperativa o comando involucra en una primera etapa su compilación para verificar si la oración dada es realmente imperativa. Una segunda etapa, consiste en la realización del análisis semántico de los comandos. Según se reporta, el trabajo tiene un módulo de detección de errores morfológicos por medio del cual detecta algunos accidentes gramaticales de género y número.

En el trabajo titulado **“Analizador sintáctico de oraciones en español usando el método dependencias”** por (Cervantes, 2005) donde se implementó un analizador sintáctico flexible y dinámico, que permitía ampliar el conocimiento lingüístico realizando mínimas inclusiones de datos. En el sistema usuarios expertos en PLN pueden agregar o modificar la información sintáctica sin necesidad de editar el código fuente. A nivel de base de datos este trabajo representa una mejora ya que el diseño del conocimiento lingüístico permite que un verbo pueda tener los mismos patrones que otro verbo y además permite que los patrones de un mismo verbo sean diferentes, dependiendo del tiempo en el que se encuentre conjugado. Otro aspecto que contempla el diseño de la base de datos es la existencia de verbos con polisemia, los cuales pueden manejar patrones distintos dependiendo del contexto en el que se usen. El trabajo usó un compendio de reglas sintácticas separado del código fuente, también usó una gramática de dependencias y cuenta con aproximadamente 70 reglas gramaticales y conocimiento lingüístico de verbos, gracias al diseño de la base de datos. Se enfocó a oraciones interrogativas e imperativas, con capacidad analítica de oraciones simples (un verbo). A pesar de todo, el sistema no pudo eliminar la ambigüedad sintáctica. El trabajo no hace uso de ninguna estrategia de detección de errores de accidentes gramaticales.

El trabajo titulado **“FreeLing: From a multilingual open-source analyzer suite to an EBMT platform”** por (Farwell & Padró, 2010) en el cual se desarrolló FreeLing que es una biblioteca de código abierto que proporciona una amplia gama de herramientas de análisis del lenguaje, para varios idiomas diferentes. La herramienta desarrollada es totalmente personalizable y extensible. Los desarrolladores pueden utilizar los recursos lingüístico (diccionarios, léxicos, gramáticas, etc.), por defecto o extenderlas, adaptarse a los dominios particulares, o incluso desarrollar nuevos recursos para los diferentes idiomas. El ser código abierto ha permitido a la aplicación FreeLing crecer más allá de sus capacidades originales, especialmente con respecto a los datos lingüísticos. Algunos de los elementos que reconoce se muestran a continuación:

- Tokenización.
- Separación de sentencia.
- Análisis morfológico, con avanzado manejo de sufijo y prefijo cuando se detectan pronombres, etc.).
- Reconocimiento de expresiones de fecha y tiempo.
- Reconocimiento de las expresiones del lenguaje corriente.
- Reconocimiento de expresión numérica (números, cantidades, porcentajes, etc.).
- Reconocimiento de magnitudes físicas: velocidad (por ejemplo 120 km/h), longitud (por ejemplo 23 cms.), presión (p. ej. 12.3 lb_f/in^2), frecuencia, densidad, energía, etc.
- Analizador de dependencia.
- Desambiguación del sentido de una palabra.

El trabajo no especifica el uso de estrategias de detección de errores de accidentes gramaticales, pero sí realiza el análisis morfológico con el cual se pueden detectar otros errores. Además de esto posibilita un análisis más detallado de la oración y genera un árbol sintáctico para la misma. FreeLing tiene disponible en su página oficial un demo. Tomando como ejemplo la frase "Juan vio un gato con el telescopio", los resultados se ven en las **Figuras 2.1**, **2.2** y **2.3**.

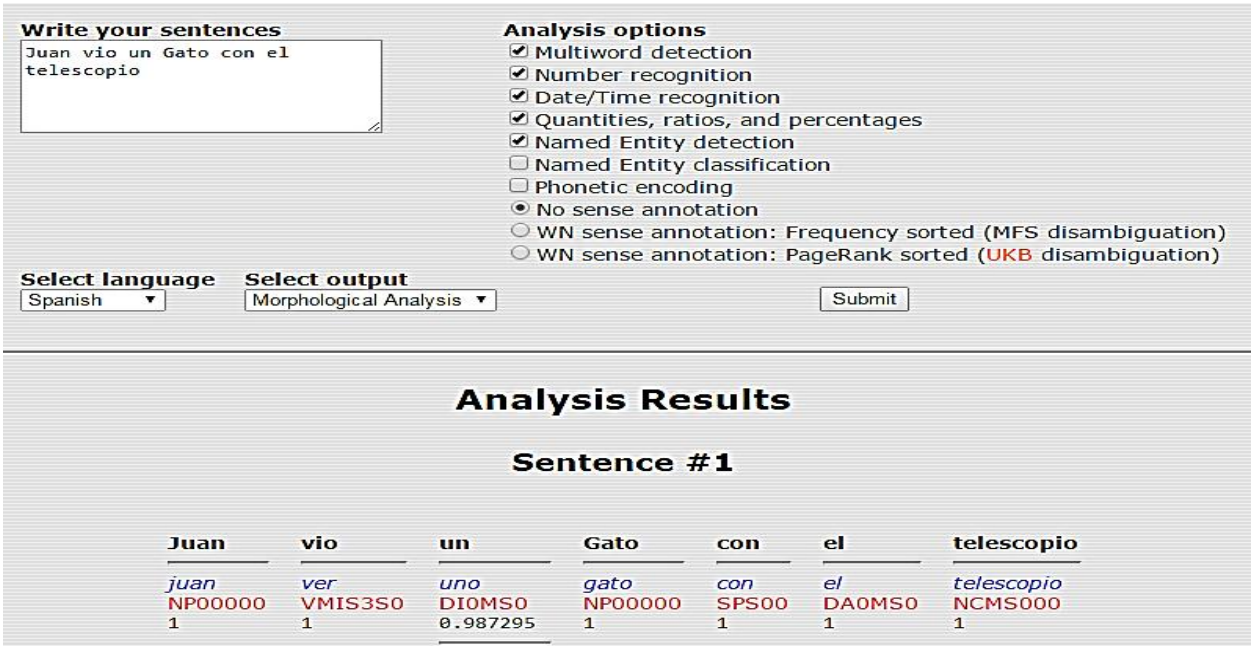


Figura 2.1 Análisis morfológico de la oración

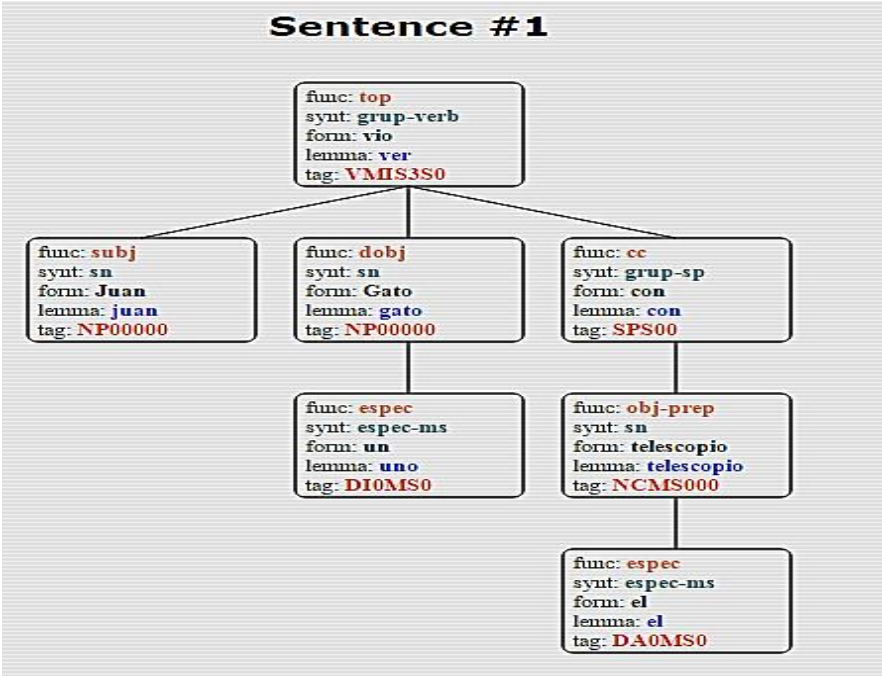


Figura 2.2 Árbol de **dependencias** de la oración.

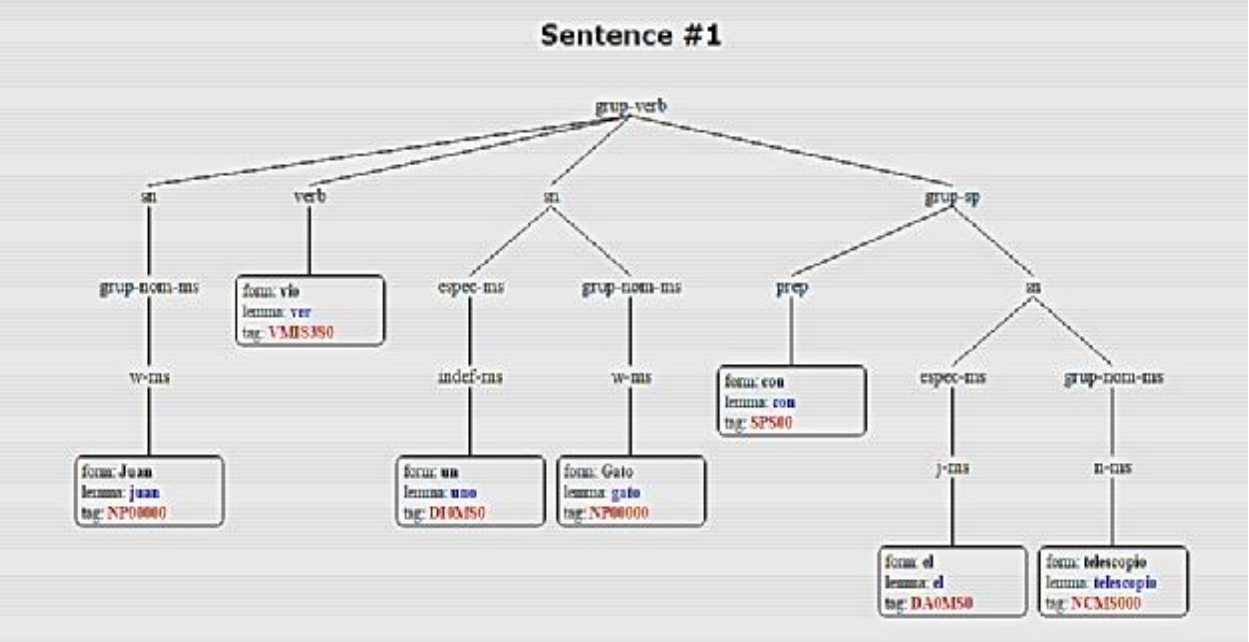


Figura 2.3 Análisis **completo** de la oración.

Otro trabajo, titulado **"ELE-Tutor Inteligente: Un analizador computacional para el tratamiento de errores gramaticales en Español como Lengua Extranjera"** (Ferreira & Kotz, 2010) desarrolló un Sistema Tutor Inteligente (STI) con Inteligencia Artificial, utilizando técnicas de comprensión de lenguaje. STI es un programa para la enseñanza-aprendizaje con el objetivo de facilitar los procesos de aprendizaje personalizados. El sistema analiza gramaticalmente la entrada y luego genera un *feedback* adecuado para los errores introducidos. Se usan técnicas de procesamiento de lenguaje natural que se basan en teorías gramaticales para procesar la entrada. El programa posee la detección de errores tanto sintácticos como morfológicos y despliega un informe de los mismos. El programa funciona haciendo uso de un Lexicón, un analizador morfológico y un analizador sintáctico. Finalmente el programa despliega un árbol sintáctico a partir del cual se genera un aviso del conflicto de accidente gramatical que puede existir dentro de la oración evaluada. En la **Figura 2.4** se puede observar este proceso.

En el trabajo **"Análisis léxico funcional de la sintaxis: propuesta para el procesamiento automático del español"** de (Loáiciga, 2012), se propuso un análisis formal de la frase simple del español según los principios de la Gramática Léxico-Funcional o LFG (del inglés Lexical Functional Grammar). Este es un formalismo de unificación, de carácter lexicalista fuerte y matemáticamente robusto. El autor propone una representación arbórea de los constituyentes de la frase y su organización sintáctica. Además se declaran estructuras de rasgos de acuerdo a las especificaciones del lexicón. Se especifica en el trabajo la estructura argumental de acuerdo a la subcategorización verbal, asegurando la completitud y la coherencia. También crearon un lexicón que contiene las categorías de verbos, sustantivos, las preposiciones *a* y *para*, artículos definidos e indefinidos,

adjetivos y conjunción. La estructura que utilizan probó ser robusta para el análisis de oraciones con hasta tres argumentos, construidas con diferentes tipos de verbos. El autor no especifica la existencia de métodos de detección de accidentes gramaticales. El análisis de la oración **“Carmen come una Manzana”** se muestra en la **Figura 2.5**.

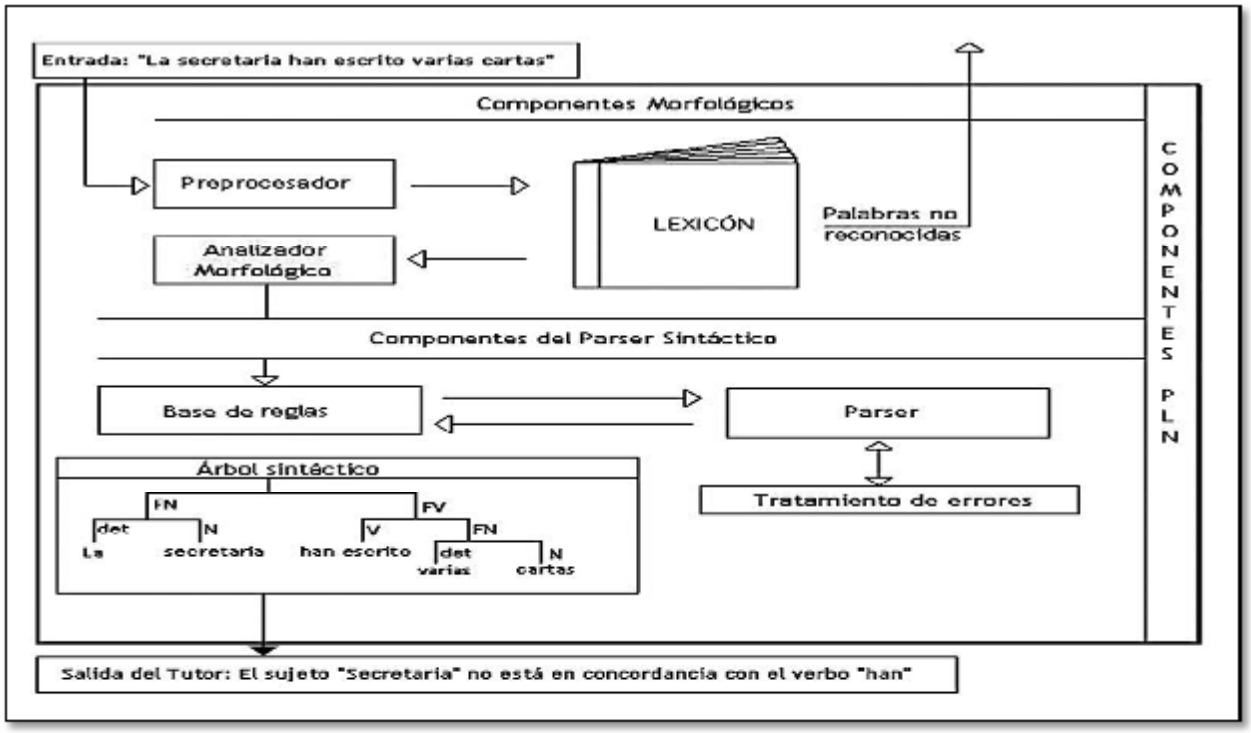


Figura 2.4 Análisis de una oración realizado por ELE-Tutor Inteligente.

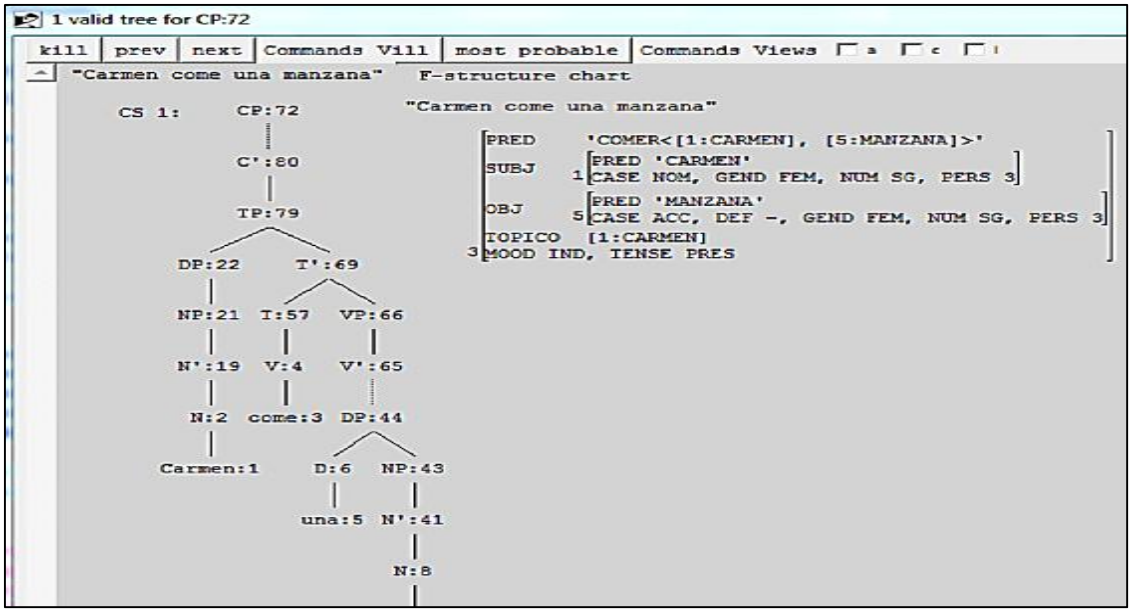


Figura 2.5 Análisis de una oración con la gramática léxico funcional LFG.

2.2 Tablas comparativas del Estado del Arte

Para comprender mejor los trabajos mencionados en el estado del arte se muestran las tablas siguientes. En la **Tabla 2.1** se muestran las características principales de los trabajos revisados en este proyecto. La **Tabla 2.2** muestra una comparativa de los trabajos anteriores.

Tabla 2.1 Características principales de los trabajos del estado del arte.

Trabajo	Año	Características Principales	
Mellado	**2014**	• 59 reglas gramaticales. • Análisis sintáctico exhaustivo.	• Gramática categorial.
Galicia	**2000**	• 150 reglas gramaticales. • Reglas sintácticas separadas del código fuente.	• Gramática MTT. • Método de desambiguación. • 53/100 oraciones resueltas.
Montes and Martínez	**2004**	• 70 reglas gramaticales. • Reglas sintácticas separadas del código fuente. • Gramática de dependencias. • Conocimiento lingüístico de verbos.	• Oraciones interrogativas e imperativas. • Capacidad analítica de oraciones simples.
Cervantes	**2005**	• Análisis sintáctico. • Análisis semántico.	• Manejo de errores morfológicos. • Oraciones imperativas.
Farwell and Padró	**2010**	• Análisis morfológico • Reconocimiento de expresiones fecha /tiempo. • Reconocimiento lenguaje corriente. • Reconocimiento de expresión numérica.	• Reconocimiento de la magnitud física. • Etiquetado de parte de la oración. • Gramática de dependencias. • Desambiguación del sentido de palabra
Ferreira and Kotz	**2010**	• Analizador Sintáctico. • Analizador morfológico. • Teorías gramaticales.	• Estrategia de *feedback* para errores. • Detección de errores sintácticos y morfológicos.
Loáiciga	**2012**	• Gramática LFG. • Representación de constituyentes. • Lexicón incluye las categorías de verbos, sustantivos, las preposiciones *a* y *para*, artículos definidos e indefinidos, adjetivos y la conjunción.	• Oraciones con hasta tres argumentos.

Tabla 2.2 Tabla comparativa de trabajos del estado del Arte.

Trabajo	Año	Detección de		Generación de Árbol de Análisis Sintáctico
		Errores sintácticos	Conflicto de Accidentes Gramaticales	
Galicia	2000	û	ü	û
Montes et al.	2004	ü	ü	û
Cervantes	2005	ü	û	ü
Farwell y Padró	2010	û	û	ü
Ferreira y Kotz	2010	ü	ü	ü
Loáiciga	2012	ü	û	ü
Mellado	2014	ü	û	û
Este trabajo	2017	ü	ü	ü

Capítulo 3 Marco Teórico

3.1 Lenguaje

Un lenguaje es una herramienta de comunicación estructurada por medio de la cual el hombre puede manifestar lo que piensa o siente haciendo uso de símbolos o sonidos articulados. También cabe la pena resaltar que para cada lenguaje existe un contexto de uso y ciertos principios combinatorios formales.

Existen dos clases de lenguajes que podemos diferenciar muy claramente dentro del lenguaje, que son (Rojas, 2009):

- El lenguaje Formal.
- El lenguaje Natural.

Estas clases de lenguaje se describen a continuación.

3.1.1 Lenguaje Formal

Un lenguaje formal es una construcción artificial desarrollada por los humanos. Este tipo de construcciones hacen uso de las matemáticas y otras disciplinas formales, incluyendo lenguajes de programación. Además las mismas tienen estructuras internas las cuales comparten con el lenguaje humano natural, por lo que pueden analizarse y tratarse bajo los mismos conceptos que éste en algunos casos. El lenguaje formal generalmente se usa para representar a través de símbolos algún tipo de conocimiento. Los lenguajes formales pueden ser utilizados para modelar teorías de la mecánica, física, matemática, ingeniería eléctrica, o de otra naturaleza, con la ventaja de que en este tipo de lenguaje no existe la ambigüedad. Algunas de las características presentes en los lenguajes formales son las siguientes:

- Se obtienen a partir de una teoría preestablecida.
- La cantidad de componentes semánticos es mínima.
- Se pueden incrementar los componentes semánticos según requiera la teoría a formalizar.
- La sintaxis no produce ambigüedad en las oraciones.
- El rol de los números es de gran importancia.
- Completa formalización que aporta un potencial a la construcción computacional.

3.1.2 Lenguaje Natural

El Lenguaje Natural (LN) es el lenguaje hablado o escrito por humanos. El LN es el medio que utiliza el ser humano de manera cotidiana para establecer la comunicación con las demás personas. Este tipo de lenguaje nos permite designar toda la actividad humana y razonar a cerca de ella. Este lenguaje fue desarrollado y organizado a partir de la experiencia humana y puede ser utilizado para analizar situaciones altamente complejas. Gracias a la riqueza de sus componentes semánticos el LN posee un gran poder expresivo. Por otro lado, la sintaxis de un LN puede ser modelada fácilmente por un lenguaje formal, similar a los utilizados en las matemáticas y la lógica. Algunas de las características presentes en los lenguajes naturales se muestran a continuación:

- Se desarrolla por enriquecimiento progresivo antes de cualquier intento de formación de una teoría.
- Su carácter expresivo se debe a la gran riqueza del componente semántico (polisemántica), una palabra en una oración puede tener diversos significados.
- Es imposible realizar una formalización completa del lenguaje natural.

3.2 Procesamiento de Lenguaje Natural

El procesamiento de lenguaje natural (PLN) es el campo de las ciencias computacionales que combina tecnologías diversas como la inteligencia artificial, el aprendizaje automático o la inferencia estadística con la lingüística aplicada. Se puede decir que el PLN es un conjunto de técnicas computacionales por medio de las cuales se puede analizar y representar naturalmente textos en uno o más niveles de análisis lingüísticos, con el fin de llevar a cabo el tratamiento del lenguaje como un humano para un rango de tareas y aplicaciones (Liddy, 2001). Su principal objetivo es hacer posible la comprensión y el procesamiento asistidos por ordenador de información expresada en lenguaje humano, para determinadas tareas de entre las cuales podemos tener (Benavides & Rodríguez, 2007):

- Traducción automática.
- Corrección de textos.
- Recuperación de la información.
- Extracción de información y resúmenes.
- Análisis de opiniones.
- Búsqueda de documentos.
- Sistemas de diálogo interactivos.
- Sistemas inteligentes para la educación y el entrenamiento.

3.2.1 Sistemas de Procesamiento de Lenguaje Natural

Los sistemas de procesamiento de lenguaje natural son sistemas mediante los cuales una persona y una máquina pueden establecer una comunicación haciendo uso del lenguaje natural donde los diferentes fenómenos lingüísticos sirven de base para la creación modificación y selección de datos. Una modalidad de sistemas de PLN son las interfaces de lenguaje natural que son mecanismos de comunicación entre una persona y una máquina a través de lenguaje natural, donde los distintos fenómenos lingüísticos hacen la función de control en la creación, modificación y selección de datos.

3.2.1.1 ILNBD

La Interfaz de lenguaje natural hacia bases de datos son un tipo de interfaz de usuario que permite la comunicación entre humanos y máquinas, donde los verbos, frases y cláusulas actúan como controles de la interfaz de usuario para crear, seleccionar y modificar datos en aplicaciones de software. Tienen una gran ventaja que es su velocidad y facilidad pero hay factores como la comprensión, que añaden una dificultad significativa, ya que podemos encontrar entradas ambiguas en el sistema (Hill, 1983). Las interfaces de lenguaje natural componen un área activa de estudio en el campo del procesamiento del lenguaje natural y la lingüística computacional.

3.3 Elementos para el Procesamiento de Lenguaje Natural

En este apartado se describen los elementos que hacen parte del procesamiento del lenguaje natural.

Frase

Grupo de una o más palabras que funciona como unidad, pero que (normalmente) no funciona en su totalidad independientemente, como en el caso de una oración.

Oración

La oración es la máxima unidad lingüística con sentido completo. Esta contiene dos partes principales que conocemos, el sujeto y el predicado.

Gramática

Es la manera característica en que se combinan los elementos básicos (especialmente los elementos léxicos) de una lengua para formar estructuras más complejas que permitan la comunicación de los pensamientos. La gramática incluye la morfología y la sintaxis: algunos analistas incluyen la fonología, la semántica y el léxico también como parte de la gramática, definir desde punto de vista lingüístico la gramática es la ciencia que estudia los elementos de una lengua y sus combinaciones.

Palabra

Es una raíz, junto con los afijos que dependan de ella y posiblemente de otras raíces (en el caso de una raíz compuesta), que puede pronunciarse sola en el uso normal de una lengua, por ejemplo, como respuesta a una pregunta. Frecuentemente las palabras tienen rasgos fonológicos especiales.

3.4 Categoría gramatical

Una categoría gramatical se encuentra definida como cada una de las diferentes funciones que desempeña una palabra dentro de una oración. Según la RAE es la propiedad gramatical que se expresa a través de los morfemas flexivos.

La gramática Tradicional Española suele considerar los siguientes tipos de palabras:

- Sustantivo
- Adjetivo
- Verbo
- Pronombre
- Determinante
- Adverbio
- Preposición
- Conjunción
- Interjección

3.4.1 Etiqueta Gramatical

En lingüística computacional, el etiquetado léxico es el proceso de asignar a cada una de las palabras de un texto su categoría gramatical. Este proceso se puede realizar de acuerdo con la definición de la palabra o el contexto en que aparece, por ejemplo su relación con las palabras adyacentes en una oración. También se puede realizar utilizando un diccionario o lexicón que contenga información y descripción de la palabra. Dentro de un Lexicón cada palabra tiene una

etiqueta. Esta etiqueta funciona como una especie de identidad de cada palabra, dado que la misma contiene información respecto a la categoría gramatical, los tiempos de la palabra y sus accidentes gramaticales. En la **Figura 3.1** podemos ver un ejemplo de la etiqueta para la palabra *número*.

número
NC00MS0

Figura 3.1 Ejemplo de Etiqueta

En la **Figura 3.1** podemos ver que las dos primeras letras de esta etiqueta muestran respectivamente la categoría y la descripción de la palabra, la antepenúltima y penúltima el género y el número. Es importante mencionar que el etiquetado es relevante para que se realice un análisis sintáctico efectivo. En el **Anexo B.2.1** se describen las etiquetas de las palabras con mayor detalle.

3.4.2 Ambigüedad

Término que hace referencia a aquellas estructuras gramaticales que pueden entenderse de varios modos o admitir distintas interpretaciones y dar, por consiguiente, motivo a dudas, incertidumbre o confusión.

3.4.3 Variación

Una variación es una interpretación para un conjunto de palabras que conforman una oración. Dado que una palabra puede tener más de una categoría gramatical, una oración puede tener más de una variación. Para determinar el número de variaciones totales que una consulta tiene, se multiplica el número de categorías que tiene cada palabra que hay en una oración. Un Ejemplo lo podemos ver en la **Figura 3.2**.

Consulta	***Lista***	***el***	***número***	***de***	***pasajeros***	***de***	***cada***	***vuelo***
Categorías gramaticales	5, 3, 2	1	2	7	2	7	4, 3	5, 2
Nº de categorías	3	1	1	1	1	1	2	2

Figura 3.2 *Variaciones construidas para la consulta*

Para calcular el número total de variaciones que tiene una consulta, se multiplica la cardinalidad del conjunto de categorías gramaticales de cada una de las palabras que constituyen la consulta a analizar; es decir, los valores que aparecen en la fila ***Nº de categorías*** de la **Figura 3.2** Este cálculo se realiza como se ve en la **Figura 3.3**.

Total de variaciones = 3 × 1 × 1 × 1 × 1 × 1 × 2 × 2 = 12

Figura 3.3 *Calculo de las variaciones de la oración de la* **Figura 3.2**

La **Figura 3.4** muestra todas las variaciones de la oración de la **Figura 3.2**, que son 12.

i	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*
0	5	1	2	7	2	7	4	5
1	5	1	2	7	2	7	4	2
2	5	1	2	7	2	7	3	5
3	5	1	2	7	2	7	3	2
4	2	1	2	7	2	7	4	5
5	2	1	2	7	2	7	4	2
6	2	1	2	7	2	7	3	5
7	2	1	2	7	2	7	3	2
8	3	1	2	7	2	7	4	5
9	3	1	2	7	2	7	4	2
10	3	1	2	7	2	7	3	5
11	3	1	2	7	2	7	3	2

Figura 3.4 *Variaciones de la oración de la* **Figura 3.2**

3.5 Representación de árboles sintácticos

Para representar los árboles sintácticos se definió un formato que se construye a partir de los elementos de la reducción sintáctica. Los Arboles Sintácticos son la estructura que permite guardar la información obtenida de una reducción en la fase de reducción sintáctica de un Analizador Sintáctico. El formato mediante el cual se pueden representar los árboles en forma de cadena consiste en una secuencia de caracteres ordenados, siendo el primero, más a la izquierda el nodo raíz y los que le siguen subsecuentemente representan las siguientes ramas del árbol. De esa manera se puede representar fácilmente un árbol sintáctico. El formato nos dice que el árbol presentado en la **Figura 3.5** se representa de la siguiente manera (***A***, (***B***, ***E***, ***F***, ***G***), ***C***, (***D***, ***H***)).

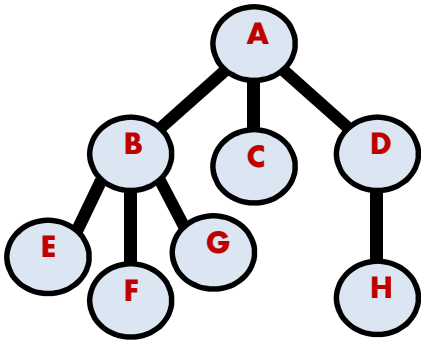


Figura 3.5 Ejemplo de árbol sintáctico

3.6 Accidentes Gramaticales

Según la Real Academia Española (RAE) el accidente gramatical es la propiedad gramatical que se expresa a través de los morfemas flexivos como el género y el número, es decir, son los cambios que las palabras presentan en su terminación para darle sentido a la oración.

En español, las palabras que presentan variaciones son (LLorach, 1999):

- sustantivo,
- adjetivo,
- pronombre y
- verbo.

El accidente gramatical se define por la estructura de la palabra, dado que toda palabra se constituye por: lexema (raíz) y morfema (terminación). Los cambios que existen en los morfemas son los que configuran un accidente gramatical como vemos en la **Figura 3.6**.

Ejemplo	*Lexema*	*Morfema*
Alumna	**alumn**	**a**
Alumno	**alumn**	**o**
Alumnas	**alumn**	**as**
Alumnos	**alumn**	**os**

Figura 3.6 Ejemplo de Accidente Gramatical

En el ejemplo de la **Figura 3.6**, la raíz de la palabra es Alumno y sus desinencias de género y número son: Alumna, Alumnos, Alumnas, es decir, son sus Accidentes Gramaticales. En algunos sustantivos no se tiene accidente gramatical de género. Existen 4 tipos de accidentes gramaticales conocidos en la lengua española (GramaticasNet, 2014):

- Accidentes Gramaticales del Sustantivo
- Accidentes Gramaticales del Adjetivo
- Accidentes Gramaticales del Pronombre
- Accidentes Gramaticales del Verbo

También se tienen en cuenta otras clasificaciones atendiendo a diferentes criterios (punto de vista utilizado para discriminar los elementos). Estos criterios son según (Mejia, Mira, & Mercado, 2009).

- **Semántico**. Cuando sirve para nombrar cosas o personas. Atiende a lo que significan.
- **Morfológico**. Cuando se define al sustantivo como *palabra variable*. Se fija en los monemas y morfemas que las integran.
- **Sintáctico**. Cuando puede ser acompañado por determinantes y adjetivos. Según si acompañan y complementan o cuando no lo hacen.
- **Funcional**. Cuando desempeñan funciones de sujeto y complemento directo. Se tiene en cuenta el trabajo que realiza en la oración o mensaje.

3.6.1 Accidentes gramaticales del sustantivo

Los accidentes gramaticales del Sustantivo son los nominales: El género y el número.

Género: Morfema que indica masculinidad o femineidad. Los géneros son masculinos y femeninos.

Número: Morfema que indica singularidad o pluralidad.

En la **Tabla 3.1** se presentan los diferentes tipos de accidentes gramaticales que puede llegar a experimentar el sustantivo.

3.6.2 Accidentes gramaticales del adjetivo

Los adjetivos no tienen género y número por sí mismos, sino que lo adoptan del sustantivo que modifican. El adjetivo también puede tener Accidentes de Grado. En la **Tabla 3.2** se presentan los diferentes tipos de accidentes gramaticales que puede experimentar el Adjetivo.

3.6.3 Accidentes gramaticales del pronombre

Los pronombres tienen accidentes gramaticales de Género (masculino y femenino) y Número (plural y singular). La **Tabla 3.3** presenta los diferentes tipos de accidentes gramaticales que puede llegar a experimentar el pronombre.

3.6.4 Accidentes gramaticales del verbo

Los accidentes gramaticales del verbo son las desinencias que los verbos pueden tener y los diferentes significados que los mismos pueden expresar. Los accidentes gramaticales del verbo puedes ser de: número, persona, tiempo y modo. En la **Tabla 3.4** se pueden observar los diferentes tipos de accidentes gramaticales que pueden llegar a experimentar el verbo.

Tabla 3.1 Clasificación de accidentes gramaticales del Sustantivo.

Accidente	Clase	Descripción	Ejemplo
Género	**En Forma Fija**	El artículo cambia el género (Significado cambia)	**la** capital / **el** capital **la** cólera / **el** cólera
	En Doble Forma	Cambia terminación del sustantivo.	profes-**or** / profes-**ora** poet-**a** / poet-**isa**
	Comunes de Dos	Sustantivo no varía, El artículo marca el género.	**el** amigo / **la** amiga **el** artista / **la** artista
	Epiceno	Artículo y el sustantivo no cambian el adjetivo marca el género.	La avestruz **macho** / la avestruz **hembra**. El cóndor **macho** / el cóndor **hembra**.
	Heterónimo	Se designa el género con palabras diferentes	Caballo / **yegua** hombre / **mujer**
	Ambiguo	Indistintamente, se podrá usar 'el' o 'la' para marcar el género.	**el/la** mar **el/la** azúcar
Número	**Expresa número plural**	Sustantivo termina en vocal no acentuada se agrega "s". Sustantivo termina en consonante o en vocal acentuada se agrega "es".	carpeta-**s** ; perro-**s** ; vida-**s**; libro-**s** ají-**es** ; amor-**es**; ñandú-**es**; árbol-**es**
	Plural sustantivo compuesto		Guardabosque /guardabosqu**es** Gentilhombre / gentil**es**hombr**es**
	Sustantivos sin singular		Anales, bienes (capital), comicios, cosquillas,
	Sustantivos sin plural		Cuclillas, ínfulas, nupcias. Salud, caos, oro, plata.
	Particularidades	Sustantivos graves, esdrújulos terminados en "s" o "x", expresan el número por el artículo.	**La** crisis / **las** crisis **la** tesis / **las** tesis **el** tórax / **los** tórax

Tabla 3.2 Accidentes gramaticales del Adjetivo.

Accidente	Clase	Descripción	Ejemplo
Género	**Adjetivos de una terminación**	Invariables, misma forma para los dos géneros	La niña/El niño **alegre** El/La mar **azul** El hombre/ La mujer **inteligente**
	Adjetivos de dos terminaciones	Variables en género. Presentan terminación para cada género : -o/-a	Bonit**o** / Bonit**a**, Pequeñ**o**/Pequeñ**a** Atencios**o**/Atencios**a**
Número	**Adjetivos en singular, plural**	Se añade -s si terminan en vocal no acentuada. Si terminan en vocal acentuada o en consonante se añade –es.	Grande/Grande**s** Feo/Feo**s** Civil/ Civil**es**, Azul/ Azul**es**.
Grado	**Grado positivo**	Los adjetivos en grado positivo expresan cualidad sin especificar ningún grado.	Juan es **inteligente**.
	Grado comparativo	El adjetivo se cuantifica por adverbios de cantidad: menos, tan, más.	Juan es **más** inteligente que pablo.
	Grado superlativo	El adjetivo se expresa en su forma superlativa añadiendo adverbio de cantidad o por el sufijo-isimo, errimo.	Pobr**ísimo**, alt**ísimo**, paup**érrimo**.

Tabla 3.3 Clasificación de accidentes gramaticales del Pronombre.

Accidente	Ejemplo
Género	Ést**e** / ést**a** Ése / ésa aqu**él** / aqu**élla**
Número	Ést**os** / ést**as** és**os** / és**as** aqu**éllos** / aqu**éllas**

3.7 Concordancia Gramatical

Se entiende por concordancia gramatical a la relación sintagmática entre miembros de la misma frase o sintagma, es decir, es la regla o recurso gramatical que tiene como función marcar las relaciones que puedan tener las palabras que la constituyen.

Tabla 3.4 Clasificación de accidentes gramaticales del verbo.

Accidente	Clase	Referencia	Ejemplo
Número	**Singular / Plural**	un sólo sujeto / varios sujetos	Tú estudiaste Vosotros estudiáis
Persona	**Primera / Segunda / Tercera**	persona(s)que habla persona(s)que escucha quien(es)se habla	Yo escribo Tú lees Él estudia.
Tiempo	**Pasado/Pretérito** **Presente** **Futuro**	persona(s)que habla persona(s)que escucha de quien(es)se habla	Él pintó la pared Tú pintas la pared Nosotros la pintaremos
Modo	**Indicativo** **Subjuntivo** **Imperativo**	Hechos reales(seguros) Expresar deseo(duda) Expresión de mandato	Acertó una quiniela Quisiera acertar ¡Adivina el resultado!

Otro concepto de concordancia es la conformidad de accidentes gramaticales, la lógica entre los distintos elementos que debe regir para que un texto pueda interpretarse adecuadamente. El sujeto y el predicado deben concordar en número y persona.

- **El número** es el accidente gramatical que clasifica a los seres de acuerdo a la cantidad (uno o varios), y que se llama singular si se trata de un solo ser y plural si se trata de dos o más.
- **La persona** es el pronombre que se puede asignar a cada verbo conjugado. Según la RAE la concordancia es la congruencia que se establece entre las informaciones flexivas de dos o más palabras relacionadas sintácticamente (Peña, 2013).

Se puede entonces decir que los accidentes gramaticales a través de las coincidencias forman la concordancia (RAE). En la lengua española puede existir concordancia de:

- Género
- Número
- Grado
- Persona
- Tiempo
- Modo

3.7.1 Verificación de Concordancia Gramatical

La verificación de concordancia gramatical comprueba la correcta aplicación de las palabras dentro de una oración para que la oración tenga sentido. La concordancia gramatical es la que estipula las relaciones que pueden tener las palabras que constituyen las oraciones. También se puede decir que es la conformidad de los accidentes gramaticales de las palabras que se relacionan dentro de la oración. En el desarrollo de este proyecto se buscó generar un modelo que se pudiera seguir para evaluar las oraciones. Según la RAE (RAE, 2010) en el español existen dos tipos de concordancia gramatical la nominal y la verbal, luego existe una serie excepciones que se presentan debido al orden que tomen las palabras dentro de una oración. Algunos ejemplos de estos dos tipos de concordancia se revisaron con el fin de buscar un modelo para el desarrollo de las reglas de concordancia que se usaron en el proyecto con el fin de verificar la concordancia de los elementos dentro de las oraciones. Algunos ejemplos de los tipos de concordancia se muestran a continuación para darnos una idea más clara del tema que se está abordando.

3.7.1.1 Concordancia nominal (coincidencia de género y número)

La concordancia nominal es la que se establece entre el sustantivo con el artículo o los adjetivos que lo acompañan (RAE, 2010). Tomemos los siguientes ejemplos de la **Figura 3.7**.

lA	***blancA***	***palomA***
Art	**Sust**	**Adj**
Sing. Fem.	***Sing. Fem.***	***Sing. Fem.***

a

EsOS	***librOS***	***viejOS***
Pron	**Sust**	**Adj**
Plu. Masc.	***Plu. Masc.***	***Plu. Masc.***

b

Figura 3.7 Coincidencia de género y número

A partir de la concordancia de estos tres elementos podemos definir una regla la cual dicta que siempre que estos tres elementos se encuentren dentro de una oración y estos se encuentren consecutivamente, deben de coincidir en género y número.

3.7.1.2 Concordancia verbal (coincidencia de número y persona)

Existen algunos tipos de concordancia verbal de entre los cuales una es la que se establece entre el pronombre y el verbo (RAE, 2010). Tomemos el ejemplo de la **Figura 3.8**.

La RAE establece que siempre que el pronombre y el verbo se encuentren consecutivamente en una oración, éstos deben de coincidir en número y persona.

ESOS	***cantAN***	***muy bien.***
Pron	**Verb**	
Plu. 3ra Persona.	***Plu. 3ra Persona.***	

Figura 3.8 Coincidencia de número y persona

Existen otras reglas más avanzadas las cuales requieren que más elementos coincidan dentro de la oración. Por cuestiones del alcance del proyecto de tesis no se analizaron éstas.

3.7.1.3 Coordinación de dos o más sustantivos o pronombres en singular (entes distintos)

Los artículos y sustantivos que se encuentren separados por la conjunción ***y***, deben concordar con el adjetivo, el pronombre, o con el verbo en plural del que son sujetos. Veamos los ejemplos de la **Figura 3.9**. Tomaremos como referencia los siguientes significados.

- *SM = Singular Masculino*
- *SF = Singular Femenino*
- *SN = Singular Neutro*
- *PM =Plural Masculino*
- *PF = Plural Femenino*
- *PN = Plural Neutro*

Rehogar	***la***	***cebolla***	***y***	***la***	***zanahoria***	***PICADAS***	***durante quince minutos***
	Art	***Sust***	***Conj***	***Art***	***Sust***	***Adj***	
	SF	***SF***	***-***	***SF***	***SF***	***PF***	

a

El	***oxígeno,***	***el***	***hidrógeno***	***y***	***el***	***carbono***	***LOS***	***proporciona el medio***
Art	***Sust***	***Art***	***Sust***	***Conj***	***Art***	***Sust***	***Pron***	
SM	***SM***	***SM***	***SM***	***-***	***SM***	***SM***	***PM***	

b

«La	*sal*	*y*	*el*	*agua*	*SON*	*gratis»*
Art	*Sust*	*Conj*	*Art*	*Sust*	*Verb*	
SF	*SF*	*-*	*SM*	*SM*	*PN*	

c

Figura 3.9 Coordinación de dos o más sustantivos o pronombres

Los artículos y sustantivos coordinados deben de coincidir con el pronombre en género como vemos en la **Figura 3.9 a y b**. En la **Figura 3.9 c** tenemos artículos y sustantivos coordinados pero con géneros diferentes, en estos casos la regla especifica que el verbo esté en masculino para que exista concordancia con éstos.

3.8 Taxonomía de los Accidentes Gramaticales

En este proyecto se realizó una Taxonomía de los Accidentes Gramaticales donde se clasificaron todos los tipos de accidentes gramaticales que se presentan en la lengua española. Durante el desarrollo de la taxonomía se descubrió que algunos sintagmas en las oraciones también poseen accidentes gramaticales. La taxonomía de los accidentes gramaticales propuesta en este trabajo se muestra en la **Tabla 3.5**.

Según (Mejia, Mira, & Mercado, 2009) la preposición, la conjunción y el adverbio no poseen ningún tipo de accidente gramatical. De la taxonomía propuesta en este trabajo se abordaron sólo los accidentes gramaticales de género y número, por limitaciones de tiempo y de complejidad del estudio de estos accidentes gramaticales. Otros conceptos teóricos relacionados de PLN se pueden ver en el **ANEXO A**.

Tabla 3.5 Taxonomía de Accidentes Gramaticales

Accidente Gramatical	*Categoría Gramatical*	*Tipo de Accidente*
Género	***Sustantivo***	***En Forma Fija***
		En Doble Forma
		Comunes de Dos
		Epiceno
		Heterónimo
		Ambiguo
	Adjetivo	***De una terminación***
		De dos terminaciones
	Pronombre	***Masculino / Femenino***
	Articulo	***Masculino / Femenino***
	Sig Nom - Sig Adj	***Masculino / Femenino***
	SigAdj - Sig Nom	***Masculino / Femenino***
Número	***Sustantivo***	***Expresa número plural***
		Plural sustantivo compuesto
		Sustantivos sin singular
		Sustantivos sin plural
		Particularidades
	Adjetivo	***Singular / Plural***
	Pronombre	***Singular / Plural***
	Articulo	***Singular / Plural***
	Verbo	*Singular / Plural*
	Sig Nom - Sig Adj	***Singular / Plural***
	Sig Nom - Sig Prep	***Singular / Plural***
	SigAdj - Sig Nom	***Singular / Plural***
	Sig Verb - Sig Nom - SigAdj	***Singular / Plural***
	Sig Verb - Sig Nom	***Singular / Plural***
	Sig Verb - Sig Prep	***Singular / Plural***
	Sig Adv - Sig Nom	***Singular / Plural***
	Sig Adv - Sig Prep	***Singular / Plural***
	SigPrep - Sig Nom	***Singular / Plural***
	SigPrep - Sig Verb	***Singular / Plural***
Grado	*Adjetivo*	*Positivo/ Comparativo / Superlativo*
Persona	*Verbo*	*Primera / Segunda /Tercera*
Tiempo	*Verbo*	*Pasado / Presente / Futuro*
Modo	*Verbo*	*Indicativo / Subjuntivo / Imperativo*

Capítulo 4 Análisis Sintáctico y Verificación de Concordancia

En este capítulo se explican las estructuras y métodos utilizados para verificar la concordancia entre palabras; es decir, reglas de producción con información de accidentes gramaticales, la estructura de los arboles sintácticos para representar oraciones y los algoritmos implementados para verificar la congruencia de accidentes gramaticales entre palabras de una oración.

4.1 Reglas de Producción

El estudio realizado por (Mellado 2014) basado en los conocimientos de la Real Academia Española (RAE), dio lugar a un conjunto de reglas de producción. Las cuales están compuestas por estructuras que comprenden las nueve categorías gramaticales y otros elementos como lo son los sintagmas y los complementos.

Mellado optó por asignar identificadores a dichas estructuras gramaticales, los cuales se usan para facilitar el proceso de análisis sintáctico. Los elementos se clasificaron como símbolos terminales y no terminales, dichas etiquetas e identificadores se muestran en las **Tablas 4.1** y **4.2**.

Tabla 4.1 Identificadores de los símbolos terminales

Símbolo	Significado	Identificador
CV	Cadena vacía	0
art	Artículo	1
sus	Sustantivo	2
adj	Adjetivo	3
pro	Pronombre	4
ver	Verbo	5
adv	Adverbio	6
pre	Preposición	7
con	Conjunción	8
int	Interjección	9

En el estudio realizado por (Mellado 2014) en el cual se revisó un extracto de la gramática española de la RAE, se construyeron las reglas de producción para implementar el analizador sintáctico. Otro detalle importante es el de que Mellado en su estudio observó que para construir estructuras gramaticales como los sintagmas en la gramática española, se necesitaban de una a tres categorías gramaticales en la mayoría de los casos y que en casos donde se requirieran más elementos, esos elementos eran sintagmas que se derivaban de una base de entre una y tres categorías gramaticales.

Tabla 4.2 Identificadores de los símbolos no terminales

Símbolo	Significado	Identificador
SNom	Sintagma nominal	10
SAdj	Sintagma adjetival	11
SVer	Sintagma verbal	12
SAdv	Sintagma adverbial	13
Spre	Sintagma preposicional	14
CC	Complemento circunstancial	15
CD	Complemento directo	16
CI	Complemento indirecto	17
O	Oración	-
S	Sujeto	-
FV	Frase verbal	-

Por lo tanto, todas las reglas de producción propuestas por (Mellado 2014) poseen las siguientes características:

- Tienen máximo tres elementos (símbolos terminales o no terminales),
- Todos los elementos se reducen a un símbolo no terminal,
- Son genéricas para analizar un mayor número de consultas,
- Se puede incrementar o reducir el número de reglas según sea conveniente

Cabe mencionar que en este proyecto se percibió que algunas de las reglas presentadas por Mellado afectaban la reducción sintáctica del analizador sintáctico e impedían a éste llegar a una reducción correcta de la oración, cuando ésta sí la tiene. Por esta causa se decidió eliminar las reglas 33 a 37 del conjunto que originalmente propuso Mellado, las cuales se enlistan a continuación.

4.1.1 Listado de reglas de producción

1. SNom = art sus adj
31. SAdv = adv SPre

2. SNom = art sus SNom
3. SNom = art sus SAdj
4. SNom = art sus SPre
5. SNom = art sus
6. SNom = sus adj
7. SNom = sus con
8. SNom = sus SNom
9. SNom = sus SAdj
10. SNom = sus SPre
11. SNom = sus
12. SNom = adj sus
13. SNom = adj SNom
14. SAdj = adj SPre
15. SNom = pro sus
16. SNom = pro SNom
17. SNom = pro SAdj
18. SNom = pro SPre
19. SNom = pro
20. SVer = ver art sus
21. SVer = ver adj sus
22. SVer = ver adv
23. SVer = ver SNom
24. SVer = ver SPre
25. SVer = ver
26. SAdj = adv adj SPre
27. SAdv = adv sus
28. SAdj = adv adj
29. SVer = adv ver
30. SAdv = adv SNom
32. SAdv = adv
33. SPre = pre art sus
34. SPre = pre sus adj
35. SPre = pre adj sus
36. SPre = pre sus
37. SPre = pre adj
38. SPre = pre SNom
39. SPre = pre SAdj
40. SPre = pre SVer
41.SPre = pre SAdv
42.SPre = pre SPre
43.SNom = con art sus
44.SNom = con sus
45.SVer = con ver adv
46.SVer = con ver
47.SVer = con adv ver
48.SPre = con pre
49.SNom = con SNom
50.SNom = SNom SAdj
51.SNom = SNom SPre
52.SAdj = SAdj SNom
53.SVer = SVer SNom SAdj
54.SVer = SVer SNom
55.SVer = SVer SPre
56.SAdv = SAdv SNom
57.SAdv = SAdv SPre
58.SPre = SPre SNom
59.SVer = SPre SVer

4.2 Reglas de Producción con Concordancia a Verificar

Para establecer las reglas de concordancia a utilizar en este proyecto se añadió a las reglas de producción (lado derecho) información correspondiente a los accidentes gramaticales a verificar, en este caso los accidentes de género y número. Un ejemplo de lo anterior se muestra en la **Figura 4.1.**

Regla			Género	Número
art	sust	-	gen	núm

Figura 4.1 Ejemplo de regla de producción con información de accidentes gramaticales.

A continuación se enlistan las reglas de producción con información de accidentes gramaticales.

1. art sus adj - gen núm
2. art sus SNom - gen núm
3. art sus SAdj - gen núm
4. art sus SPre - núm.
5. art sus - gen núm
6. sus adj - gen núm
7. sus con -
8. sus SNom - gen núm
9. sus SAdj - gen núm
10. sus SPre - núm
11. sus -
12. adj sus - gen núm
13. adj SNom - gen núm
14. adj SPre - núm
15. pro sus - gen núm
16. pro SNom - gen núm
17. pro SAdj - gen núm
18. pro SPre - núm
19. pro -
20. ver art sus - núm
21. ver adj sus - núm
22. ver adv -
23. ver SNom -
24. ver SPre - núm
25. ver -
26. adv adj SPre
27. adv sus -
28. adv adj -
29. adv ver -
30. adv SNom - núm
31. adv SPre - núm
32. adv -
38. pre SNom -
39. pre SAdj - núm
40. pre SVer - gen
41. pre SAdv - núm
42. pre SPre - núm
43. con art sus -
44. con sus -
45. con ver adv
46. con ver -
47. con adv ver -
48. con pre -
49. con SNom -
50. SNom SAdj - gen núm
51. SNom SPre - núm
52. SAdj SNom - gen núm
53. SVer SNom SAdj - núm
54. SVer SNom - núm
55. SVer SPre - núm
56. SAdv SNom - núm
57. SAdv SPre - núm
58. SPre SNom - núm
59. SPre SVer - núm

4.3 Algoritmos implementados

Los algoritmos implementados en este trabajo principalmente son dos: la construcción de arboles sintácticos y la verificación de concordancia entre palabras. El primero se implementó debido a que en el trabajo realizado por (Mellado 2014) no se consideró crear arboles sintácticos de una oración. En el segundo algoritmo se utilizaron los arboles sintácticos como guía para verificar la congruencia de los *Accidentes Gramaticales* entre palabras de una oración.

Antes de comenzar a explicar cómo se implementó el algoritmo de Construcción de árboles sintácticos, se describirá la información utilizada por este algoritmo y el de verificación de la concordancia; dicha información es generada en el análisis léxico, está la podemos ver en la **Tabla 4.3**.

Tabla 4.3 Etiquetado léxico de la oración *Lista el número de personas en cada vuelo*

Elemento	1	2	3	4	5	6	7	8
Oración	Lista	el	número	de	gente	en	cada	vuelo
Categoría Gramatical	Verbo	Articulo	Sustantivo	Preposición	Sustantivo	Preposición	Pronombre	Sustantivo
Identificador	5	1	2	7	2	7	4	2
Etiquetas lexicón	VMIP3S0	DA00MS0	NC00MS0	SPS0000	NC00FS0	SPS0000	PI00NS0	NC00MS0

La estructura de la **Tabla 4.3** de describe a continuación:

- La primer fila indica el número de cada uno de los elementos de la oración.
- La segunda fila muestra las palabras de la oración.
- La tercer fila muestra la categoría gramatical de la palabra.
- La cuarta fila muestra el identificador numérico de la categoría gramatical de la palabra.
- La quinta fila muestra la etiqueta de la palabra localizada en el lexicón.

A continuación se describen los principales algoritmos desarrollados en este proyecto de tesis.

4.3.1 Construcción de árboles sintácticos

Al *Analizador Sintáctico* implementado por Mellado, se le agregó un código para permitirle representar la salida (reducción valida sintácticamente) en forma de *Árbol Sintáctico*, cabe

mencionar que una oración puede tener más de uno. El formato para representar los arboles sintácticos se muestra en el **Subcapítulo 3.5.** En la **Figura 4.2** se muestra sólo una reducción de la oración de la **Tabla 4.3**, sabemos que una oración puede tener diferentes reducciones y por consiguiente árboles como reducciones tenga. La descripción completa de cómo se realiza el proceso de reducción se puede observar en el Capítulo 4 de la tesis de (Mellado, 2014).

En la **Tabla 4.4 se** muestra el ejemplo de un árbol sintáctico**.** La primera columna (**Fila**) muestra el número de fila. En la segunda columna (**Regla de Producción**) se muestran las reglas de producción, éstas se conforman por dos partes, parte izquierda y parte derecha. La parte izquierda es un símbolo no terminal que corresponde a una regla, la parte derecha se conforma por los elementos a reducir (símbolo(s) terminal(es) y/o símbolo(s) no terminal(s) separados por coma), los que generan la parte izquierda de la regla. En la tercera columna (**Árbol)** se muestra la construcción del árbol sintáctico (en formato de cadena).

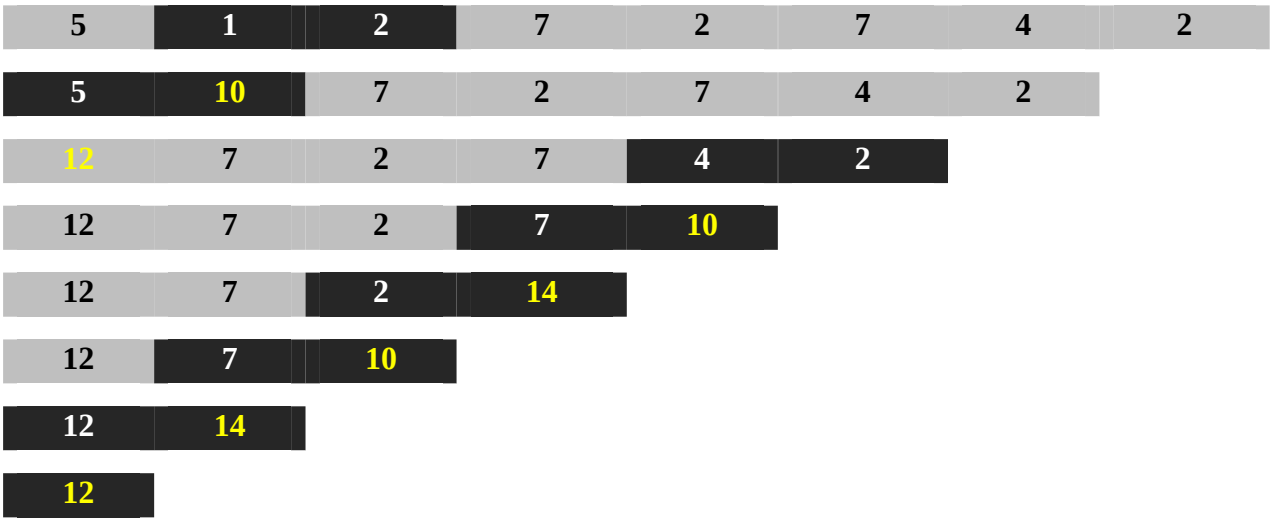


Figura 4.2 Reducción sintáctica de la oración de la **Tabla 4.3**

Tabla 4.4 Árbol sintáctico de la reducción de la **Figura 4.2**

Fila	Regla de Producción	Árbol
1	10 (1 , 2)	**(10 , 1 , 2)**
2	12 (5 , 10)	**(12 , 5,** (**10** , 1 , 2) **)**
3	10 (4 , 2)	(12, 5 , (10 , 1 ,2)) , **(10 , 4 , 2)**
4	14 (7 , 10)	(12, 5 , (10 , 1 , 2)) , **(14 , 7 ,** (**10** , 4 , 2)**)**
5	10 (2, 14)	(12, 5 , (10 , 1 , 2)) , **(10, 2,** (**14** , 7 , (10 , 4 , 2)))**)**
6	14 (7 , 10)	(12, 5 , (10 , 1 , 2)) , **(14 , 7 ,** (**10**, 2, (14 , 7 , (10 , 4 , 2))))**)**
7	12 (12 , 14)	**(12,** (**12**, 5 , (10 , 1 , 2)) , (**14** , 7 , (10, 2, (14 , 7 , (10 ,4 , 2)))))**)**

El procedimiento para construir el árbol es sencillo, la regla de producción de la Fila 1

(**Tabla 4.4**), pasa a conformar la estructura inicial del árbol sintáctico. A partir de la fila 2 se necesita realizar una pequeña verificación, para saber si la regla anterior (fila 1) es un elemento (subrama del árbol) de la regla actual; en este ejemplo se puede observar que la regla de la fila 1, es un elemento de la regla de la fila 2, es decir, la reducción (lado izquierdo) de la regla de la fila 1, es 10, este *símbolo* no terminal, a su vez es un elemento del lado derecho de la regla de la fila 2. La regla de la fila 3, como se puede ver no se enlaza con la regla anterior, la de la fila 2, por lo que se buscará qué otra regla de filas posteriores se relacione con ésta; para este caso, es justo la fila 4, sucediendo lo mismo que ocurre con la fila 1 y 2. Se continúa realizando este proceso hasta finalizar con cada una de las filas. El proceso concluye satisfactoriamente cuando se forma un árbol sintáctico, en caso contrario, es decir, cuando no se forma ésto denota que la oración no se redujo, lo cual indica que la oración es sintácticamente incorrecta. Para este ejemplo cabe mencionar que la oración se redujo satisfactoriamente.

La **Figura 4.3** muestra el árbol de forma gráfica de la reducción de la **Figura 4.2**, es otra manera de ver un árbol, este formato es más utilizado debido a que facilita su interpretación, este no se implementó por cuestiones de tiempo además de que no se comprometió en los alcances de este proyecto.

El formato de cadena usado en este trabajo para representar un árbol sintáctico permite identificar las relaciones entre las palabras, el algoritmo para construir los aboles sintácticos se muestra en la **Figura 4.4**. El ejemplo de la **Tabla 4.4** contiene reglas de producción de dos elementos, cabe mencionar que existen reglas de 1, 2 y 3 elementos, tal como se muestra al inicio de este capítulo. El algoritmo de la **Figura 4.4** es capaz de construir el árbol sintáctico independiente del tamaño de las reglas que se presenten.

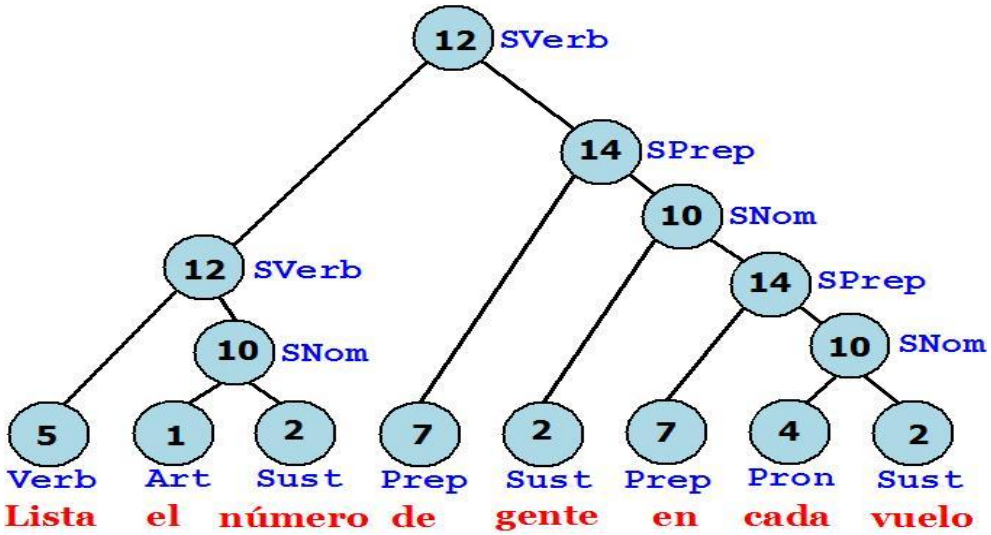


Figura 4.3 Árbol sintáctico de la oración de la **Tabla 4.3**

```
ConstruirArbol ( )
        for( i ←0 i  to elementosArbol)
              if (elementosArbol(i) == 4)                    // Regla de 3 elementos
                    if (condicion)
                          arbol ←generarRama(3)
                    else
                          arbol ←generarRamaconSubrama(3)
              if (elementosArbol(i) == 3)                    // Regla de 2 elementos
                    if (condicion)
                          arbol ← generarRama(2)
                    else if (elemento final)
                          arbol ← generardosRamas(2)
                    else
                          arbol ←generarRamaconSubrama(2)

           if (elementosArbol(i) == 2)                       // Regla de 1 elementos
                    if (condicion)
                          arbol ← generarRama(1)
                    else
                          arbol ← generarRamaconSubrama(1)
      end for
    return   árbol
```

Figura 4.4 Algoritmo para crear arboles sintacticos

4.3.2 Evaluación de concordancia entre palabras

El proceso para evaluar la concordancia (congruencia de los accidentes gramaticales) entre palabras de una oración se realiza utilizando su árbol sintáctico, éste como se mostró en el **Subcapítulo 4.3.1** está constituido de reglas de producción, con información de los accidentes gramaticales a evaluar de esos elementos. Los accidentes gramaticales de cada palabra están almacenados en el Lexicón, ver **Anexo B.2.1**. Los elementos del árbol (palabras) dependiendo de la regla de producción se verifican con relación al género o número, o a ambos (género y número). Si el género y/o número de cada elemento (palabra) es igual, indica que los elementos son congruentes con relación a sus accidentes gramaticales, en caso contrario, significa que existe una discrepancia en sus accidentes gramaticales.

Durante el proceso de evaluación de una oración se presentan tres tipos, las cuales son:

- Evaluación de símbolos terminales
- Evaluación de símbolos terminales con no terminales
- Evaluación entre símbolos no terminales

Los tipos de evaluación arriba mencionados se explicarán realizando la verificación de concordancia de la oración que se encuentra en la **Tabla 4.5**.

La evaluación de concordancia de una oración inicia evaluando los primeros elementos de un árbol sintáctico; utilizando la **Tabla 4.5** como ejemplo, los elementos de la **Fila 1**, que corresponde a la evaluación de **símbolos terminales**, nodos de los niveles más bajos del árbol (nodos hoja). Los elementos (símbolos ***1*** y ***2***) del lado derecho de la regla de la Fila 1, los que están entre paréntesis, se deben validar para verificar que tengan concordancia estos elementos. Cabe mencionar que para las demás validaciones será igual. La **Regla *5*** indica que para estos símbolos se debe verificar su género y número; de acuerdo a las etiquetas de estas palabras se observa que no existe incongruencia en los accidentes gramaticales de ambos elementos (**Figura 4.5**), la etiqueta para el elemento generado de la reducción hereda la información de género y número de estos dos elementos. Estos elementos se pueden ver en el árbol sintáctico de la **Figura 4.8 a**.

Tabla 4.5 Etiquetas de los elementos de las reglas de producción del árbol sintáctico de la **Tabla 4.4**

Fila	Regla de Producción	Etiquetas de los elementos	Árbol
1	10 (1 , 2)	**Z100MS0, (DA00MS0, NC00MS0)**	**(10 , 1 , 2)**
2	12 (5 , 10)	**Z1200S0, (VMIP3S0 , Z100MS0)**	**(12 , 5,** (**10** , 1 , 2) **)**
3	10 (4 , 2)	**Z1000S0, (PI00NS0, NC00MS0)**	(12, 5 , (10 , 1 ,2)) , **(10 , 4 , 2)**
4	14 (7 , 10)	**Z1400S0, (SPS0000, Z1000S0)**	(12, 5 , (10 , 1 , 2)) , **(14 , 7 ,** (**10** , 4 , 2)**)**
5	10 (2, 14)	**Z1000S0, (NC00FS0, Z1400S0**)	(12, 5 , (10 , 1 , 2)) , **(10, 2,** (**14** , 7 , (10 , 4 , 2)))**)**
6	14 (7 , 10)	**Z1400S0 (SPS0000 , Z1000S0**)	(12, 5 , (10 , 1 , 2)) , **(14 , 7 ,** (**10**, 2, (14 , 7 , (10 , 4 , 2))))**)**
7	12 (12 , 14)	12 (**Z1200S0 , Z1400S0**)	**(12,** (**12**, 5 , (10 , 1 , 2)) , (**14** , 7 , (10, 2, (14 , 7 , (10 ,4 , 2)))))**)**

el	número
1	2
DA00MS0	NC00MS0
10	
Z120MS0	

Figura 4.5 Verificación de concordancia entre los símbolos terminales (1, 2).

El proceso continúa evaluando las restantes filas, para este ejemplo (**Tabla 4.5**), se continua con la evaluación de la **Fila 2,** en esta parte del proceso se verá otro tipo de evaluación, la que se presenta entre **símbolos terminales y no terminales**, ya que el elemento número ***1*** es un símbolos terminal, mientras que el elemento número 2 es un símbolo no terminal; el identificador correspondiente a estos elementos son los símbolos ***5*** y ***10***, la regla de producción correspondiente es la **Regla *23***, no indica qué accidente(s) se debe(n) verificar en estos elementos; para estos casos, la etiqueta del elemento generado a partir de los elementos evaluados, hereda del **elemento no terminal** la información de los accidentes a verificar (**Figura 4.6**). No obstante a lo anterior, que la

Regla no indique qué accidente se debe evaluar en los elementos, no significa que pueda existir o no concordancia entre éstos. Por cuestiones de tiempo no se analizaron los accidentes relacionados con los verbos debido a su complejidad. Sin embargo, esto no impide que en futuras mejoras se pueda agregar la verificación de estos accidentes. En la **Figura 4.10 b** se puede ver la reducción de los símbolos ***5*** y ***10*** en el árbol sintáctico.

Lista	el número
5	10
VMIP3S0	Z100MS0
12	
Z1200S0	

Figura 4.6 Verificación de Concordancia entre los símbolos terminales (5, 10).

La siguiente evaluación es la de la **Fila 3,** la cual es semejante a la evaluación realizada a la **Fila 1**, es decir, también se debe verificar el género y número de los elementos, en este caso, de los símbolos terminales ***4*** y ***2*** (**Regla 15**). Al evaluar las etiquetas de los elementos (palabras) anteriores se observa que existe concordancia entre éstos. En la **Figura 4.10 c** se pueden ver estos elementos en el árbol sintáctico.

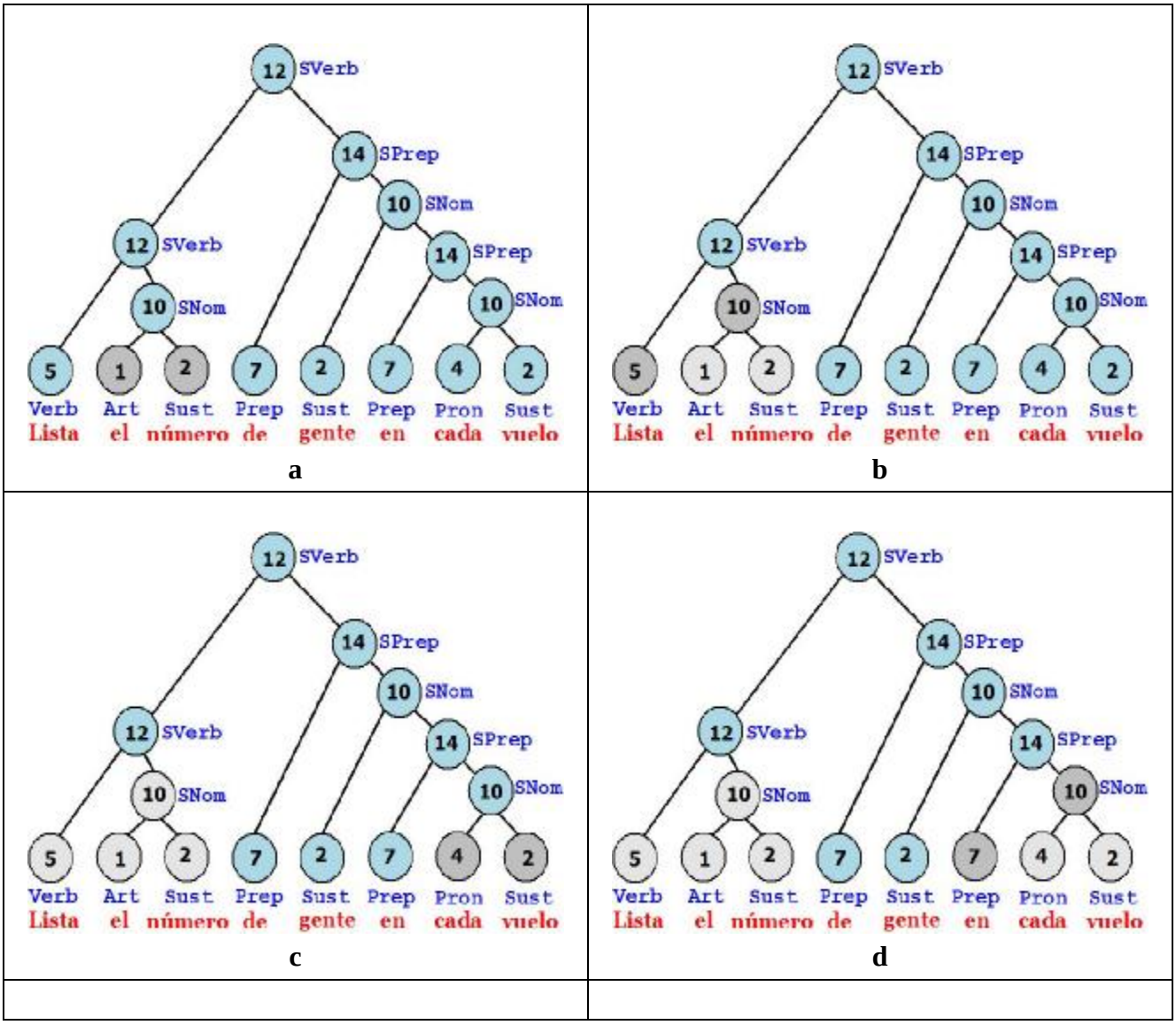
La evaluación de la **Fila 4** a la **Fila 6** es similar a la efectuada en la **Fila 2**, la que se presenta entre **símbolos terminales y no terminales.** La regla de producción de la **Fila 4** y **6** son idénticas, ambas se forman con los símbolos ***7*** y ***10***, la **Regla *38*** correspondiente no indica qué accidente se debe evaluar. Estos elementos se observan en el árbol sintáctico (**Figura 4.10 d y Figura 4.10 f)**. Al verificar la congruencia de los accidentes gramaticales de los elementos de la **Fila 5**, se identifica que para los símbolos ***2*** y ***14*** se utiliza la **Regla 10**, ésta indica que se debe verificar el accidente gramatical número. Al evaluar las etiquetas de los elementos (palabras) anteriores se observa que existe concordancia entre éstos. Se pueden ver estos elementos en el árbol sintáctico de la **Figura 4.10 e**.

La evaluación finaliza con la verificación de concordancia entre los elementos de la **Fila 7** de la **Tabla 4.5**, aquí se ve el último tipo de evaluación, la que se presenta entre **símbolos no terminales**. En la **Figura 4.7** se ven los símbolos terminales (***12*** y ***14***), en este caso la **Regla 55** indica que se debe evaluar el accidente gramatical *número* entre estos elementos. En la **Figura 4.10 g** se ven estos elementos en el árbol sintáctico.

Lista el número	de gente en cada vuelo
Sintagma verbal	Sintagma Preposicional
12	14
Z1200S0	Z1400S0
12	

Figura 4.7 Estado de la oración de la **Tabla 4.3** después de seis reducciones

Debido a que no restan más elementos a evaluar, el proceso termina, tal como se ve en la **Figura 4.10 h**.



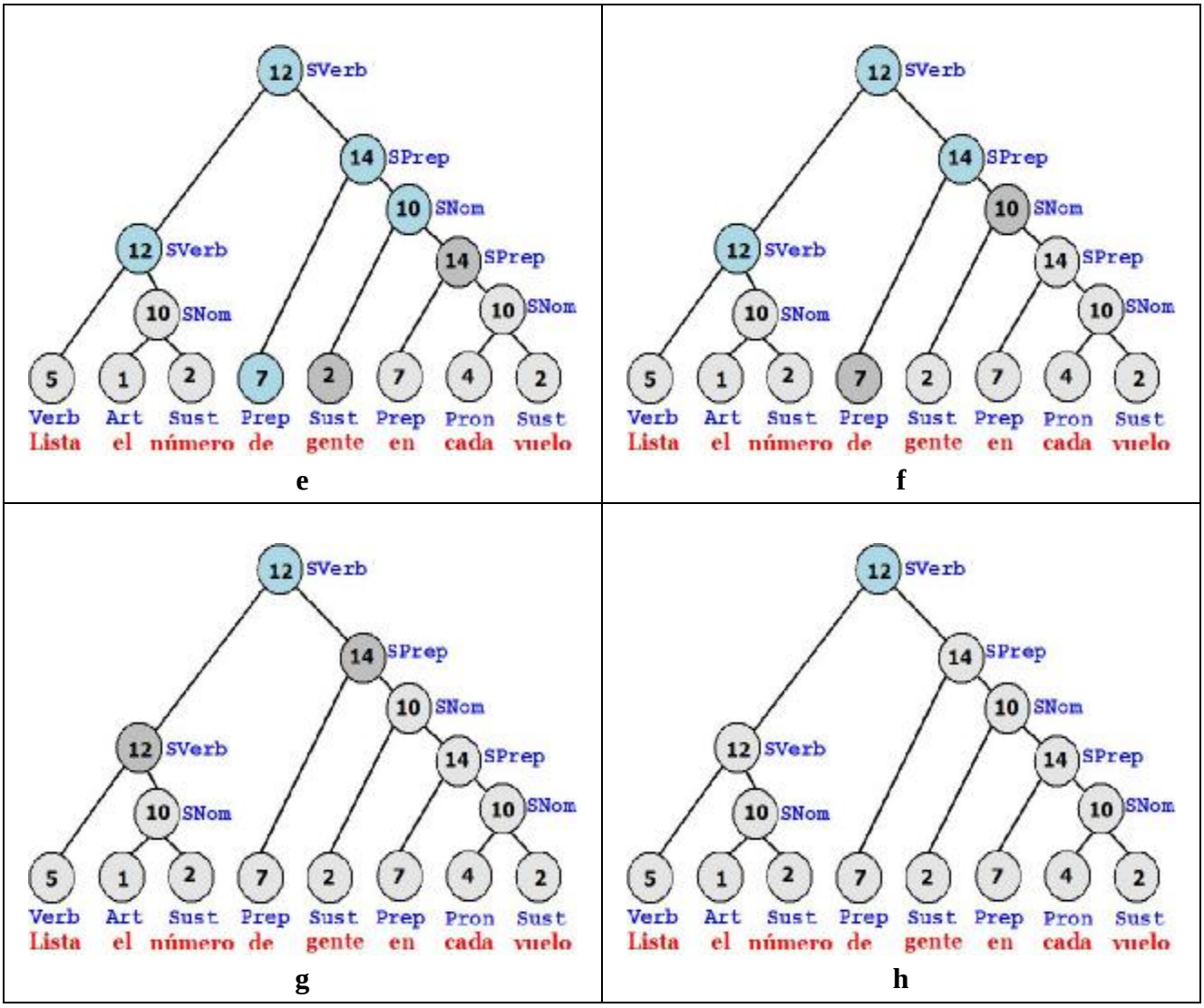


Figura 4.8 Secuencia de verificación de concordancia de la oración de la **Tabla 4.3**

El algoritmo de verificación de concordancia se puede observar en la **Figura 4.9**

```
verificarConcordancia ( oracion etiqueta  punt  red tipo)
elemento
Reglas_Concordancia
   if (tipo == 3)
     eti1 eti2 eti3  etiex = "ZZ"
     eti1 ← etiqueta.get(0 + punt)
     eti2 ← etiqueta.get(0 + punt+1)
     eti3 ←etiqueta.get(0 + punt+2)
     for (i ← 0; i to m
          if (Reglas_Concordancia[i][7] == 3)
             if (oracion.get(0 + punt) == Reglas_Concordancia[i][0] &&
                 oracion.get(0 + punt + 1) == Reglas_Concordancia[i][1])
                                                                    //Accidente de Género y Número
                if (Reglas_Concordancia[i][3] == 1 && Reglas_Concordancia[i][4] == 1)
                eti1 ←eti1.substring(4 6)
                eti2 ←eti2.substring(4 6)
                     if (eti1.equals(eti2)&& eti2.equals(eti3) )
                     etiex ← etiex.concat(Integer.toString(red).concat(eti2))
                     elemento.add(Integer.toString(red))
                     elemento.add(Integer.toString(oracion.get(punt)))
                     elemento.add(Integer.toString(oracion.get(punt+ 1)))
                     elemento.add(etiex)
                              else if (eti2.equals("FN") || eti2.equals("MN")) //Casos con palabras neutras
                     etiex ←etiex.concat(Integer.toString(red).concat(eti1))
                     elemento.add(Integer.toString(red))
                     elemento.add(Integer.toString(oracion.get(punt)))
                     elemento.add(Integer.toString(oracion.get(punt + 1)))
                     elemento.add(etiex)                    //No coincidencia de Accidentes Gramaticales
                  else
                     etiex ← Integer.toString(-1)
                     elemento.add(Integer.toString(red))
                     elemento.add(Integer.toString(oracion.get(puntero)))
                     elemento.add(Integer.toString(oracion.get(puntero + 1)))
                     elemento.add(etiex)
                else
                   if (Reglas_Concordancia[i][4] == 1)                    //Accidente de Número
                   eti1 ← eti1.substring(5 6)
                   eti2 ← eti2.substring(5 6)
                   if (eti1.equals(eti2)&& eti2.equals(eti3))
                      etiex ←etiex.concat(Integer.toString(red).concat(eti2.concat("0")))
                      elemento.add(Integer.toString(red))
                      elemento.add(Integer.toString(oracion.get(puntero)))
                      elemento.add(Integer.toString(oracion.get(puntero + 1)))
                      elemento.add(etiex)
   if (tipo == 2)
   if (tipo == 1)
 return elementosArboletiqueta.get(elementosArboletiqueta.size()-1)
```

Figura 4.9 Algoritmo para realizar la Verificación de Concordancia.

Capítulo 5 Pruebas y Resultados

5.1 Configuración del Equipo

Las pruebas realizadas en este proyecto se ejecutaron en una laptop con las especificaciones mostradas en la **Tabla 5.1**. Mientras que el software usado tanto para la implementación como para las pruebas, se encuentra descrito en la **Tabla 5.2**.

Tabla 5.1 Especificaciones del equipo

Sistema	Descripción
Procesador	Intel® Core™ i7-6500U Procesador 2.50.Ghz
Memoria	8Gb
Sistema operativo	Windows 10

Tabla 5.2 Especificaciones del software

Software	Descripción
Entorno	Netbeans ver. 8.1
Lenguaje	Java
Java	Java 8

5.2 Pruebas del Analizador Sintáctico con Concordancia entre Palabras

Las pruebas realizadas al algoritmo de verificación de la concordancia de las oraciones para verificar el funcionamiento de concordancia en éste consistió en evaluar aleatoriamente consultas escogidas de los corpus siguientes: ATIS, PUBS y CFA. El conjunto de estas consultas están en el Anexo C.

Sabemos que una oración puede tener varias formas de interpretarse y sintácticamente cada forma se determina a través de un árbol sintáctico. El algoritmo de (Mellado, 2014) realiza la reducción sintáctica y genera los diferentes arboles de la oración como podemos ver en el **Caso 5.1**. Para la oración del **Caso 5.1**, el algoritmo de reducción es capaz de encontrar diferentes árboles, cuatro de éstos se muestran de la **Figuras 5.1** a la **Figuras 5.4**.

Caso 5.1 Oración: Dame una lista de todos los tamaños de equipo y velocidad.

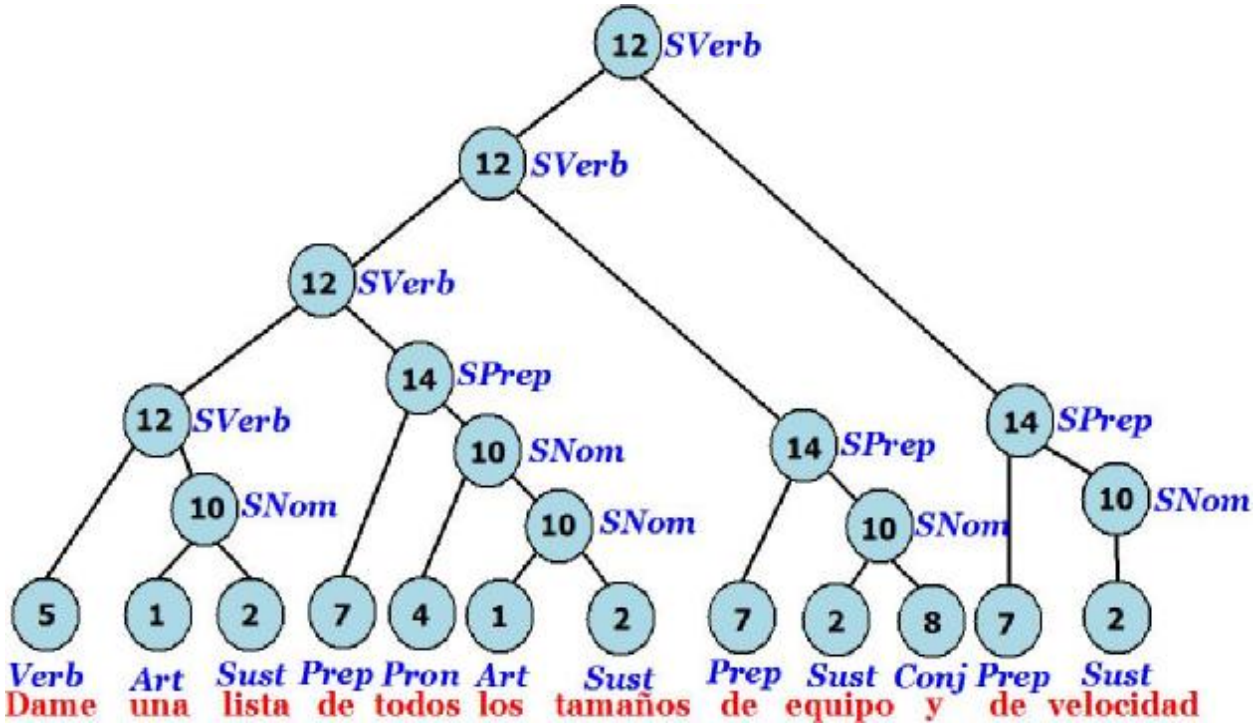


Figura 5.1 Árbol Sintáctico 1 para la oración del **Caso 5.1**

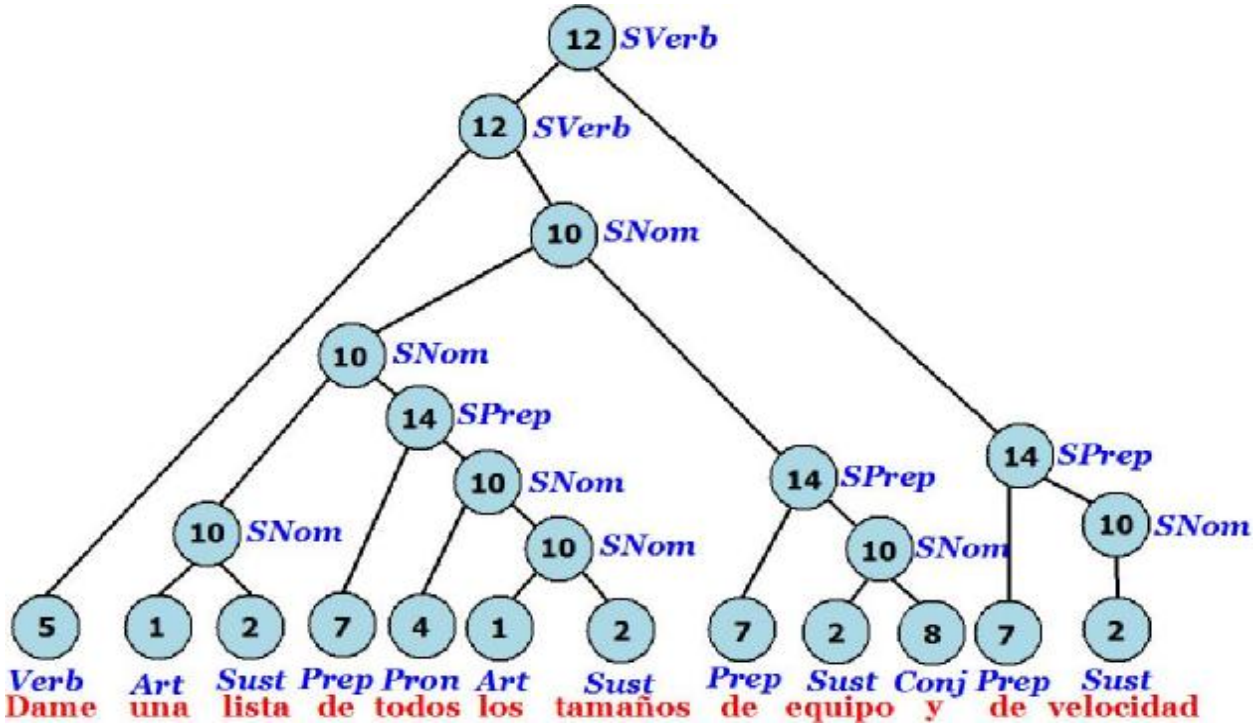


Figura 5.2 Árbol Sintáctico 2 para la oración del **Caso 5.1**

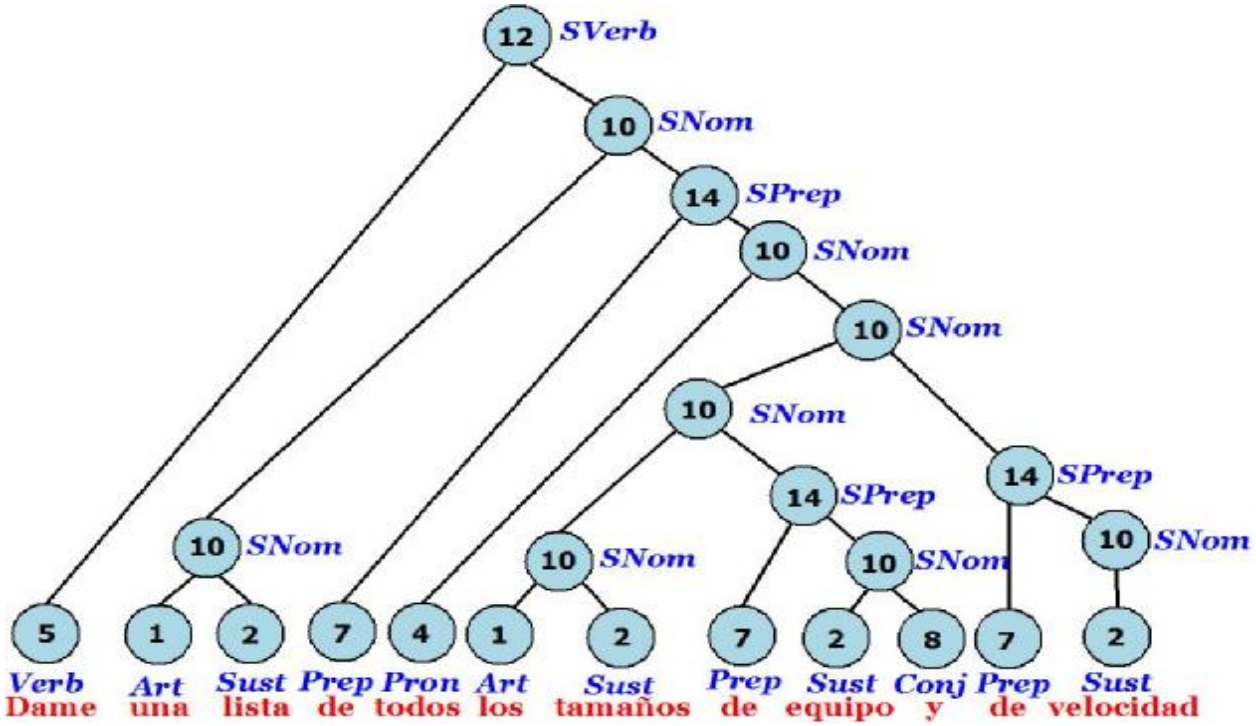


Figura 5.3 Árbol Sintáctico 3 para la oración del **Caso 5.1**

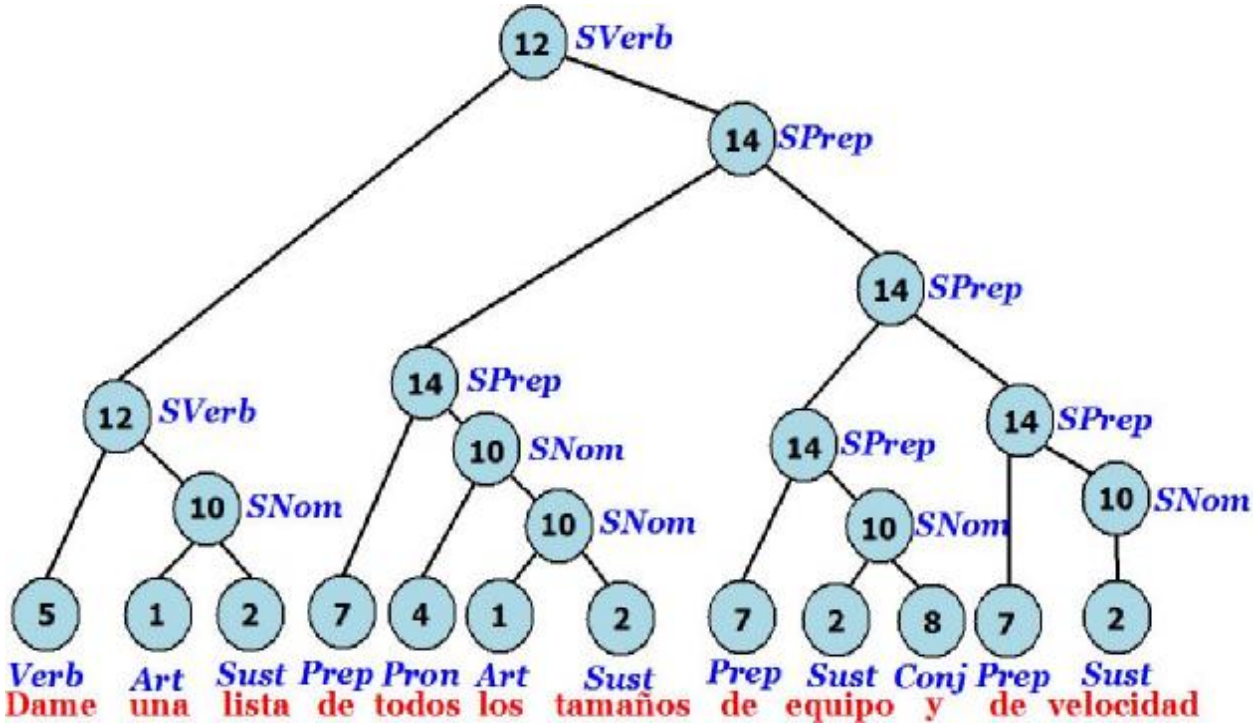


Figura 5.4 Árbol Sintáctico 4 para la oración del **Caso 5.1**

A nivel del análisis sintáctico no es posible detectar cuál o cuáles de estos árboles serían los correctos, dado que este tipo de detección se realiza a nivel del análisis semántico.

5.3 Pruebas Negativas

Según (Balkan, Netter, & Arnold, 1994) los corpus de prueba son extraídos de manera natural de los textos y por lo general estos corpus no presentan errores. Por eso es necesario someter el algoritmo a pruebas donde se consideren ejemplos negativos en los cuales existen incongruencias que permitan probar la eficacia del algoritmo. Balkan explica que este tipo de ejemplos no ocurre naturalmente en un corpus; sin embargo, someter a este tipo de ejemplos en verificadores gramaticales o correctores de lenguaje puede evidenciar información de gran importancia. Se realizan pruebas negativas para obtener mayor seguridad con relación al desempeño del algoritmo. En estas pruebas se optó por introducir de manera arbitraria palabras que causaran una incongruencia de accidentes gramaticales en las oraciones.

La **Figura 5.5** muestra la oración del **Caso 5.2** con cambios de accidentes gramaticales de *número* en algunos de sus elementos.

Caso 5.2 Oración**:** Lista el número de gente en cada vuelo.

Lista	los	número	de	gente	en	cada	vuelos
Verbo	Articulo	Sustantivo	Preposición	Sustantivo	Preposición	Pronombre	Sustantivo
5	1	2	7	2	7	4	2
VMIP3S0	DA00MP0	NC00MS0	SPS0000	NC00FS0	SPS0000	PI00NS0	NC00MP0

Figura 5.5 Oración del **Caso 5.2** con errores de concordancia introducidos.

Un error de concordancia en la oración del **Caso 5.2**, se ve en la **Figura 5.6**.

los	número
Articulo	Sustantivo
1	2
DA00MP0	NC00MS0
10	
-1	

Figura 5.6 Verificación de Concordancia en símbolos terminales (1, 2).

En la **Figura 5.6** el artículo y el sustantivo coinciden en *género*, pero no en *número*, dado que sus etiquetas en la posición donde guardan la información de número no coinciden. En este caso se apunta el accidente gramatical con un -1 en la etiqueta resultante. Otros dos elementos que no

coinciden en número los podemos ver en la **Figura 5.7**, son el pronombre y el sustantivo como lo indican sus etiquetas. En estos casos la etiqueta resultante de estos dos elementos será -1 indicando que no existe concordancia entre estos elementos (palabras). El proceso completo de verificación de concordancia de la oración del **Caso 5.2** se muestra en la **Figura 5.8**.

cada	vuelos
Pronombre	Sustantivo
4	2
PI00NS0	NC00MP0
10	
-1	

Figura 5.7 Verificación de Concordancia en símbolos terminales (4, 2).

La **Figura 5.9** muestra la oración del **Caso 5.3** con cambios de accidentes gramaticales de *género* de uno de sus elementos.

Caso 5.3 Oración: Despliega los tiempos de salida

El proceso completo de verificación de concordancia de la oración del **Caso 5.3** se muestra en la **Figura 5.10**.

Lista	los	número	de	gente	en	cada	vuelos
5	1	2	7	2	7	4	2
Verb	Art	Sust	Prep	Sust	Prep	Pron	Sust
VMIP3S0	DA00MP0	NC00MS0	SPS0000	NC00MS0	SPS0000	PI00NS0	NC00MP0
Verb	SNom		Prep	Sust	Prep	Pron	Sust
VMIP3S0	-1		SPS0000	NC00MS0	SPS0000	PI00NS0	NC00MP0
SVerb			Prep	Sust	Prep	Pron	Sust
-1			SPS0000	NC00MS0	SPS0000	PI00NS0	NC00MP0
SVerb			Prep	Sust	Prep	SNom	
-1			SPS0000	NC00MS0	SPS0000	-1	
SVerb			Prep	Sust	SPrep		
-1			SPS0000	NC00MS0	-1		
SVerb			Prep	SNom			
-1			SPS0000	-1			
SVerb			SPrep				
-1			-1				
SVerb							
-1							

Figura 5.8 Proceso de verificación de concordancia de la oración del **Caso 5.2**

Despliega	las	tiempos	de	salida

Verbo	Articulo	Sustantivo	Preposición	Sustantivo
5	1	2	7	2
VSIP3S0	DA00FP0	NC00MP0	SPS0000	NC00FS0

Figura 5.9 Oración del **Caso 5.3** con errores de concordancia introducidos.

Despliega	las	tiempos	de	salida
5	1	2	7	2
Verb	Art	Sust	Prep	Sust
VSIP3S0	DA00FP0	NC00MP0	SPS0000	NC00FS0
Verb	SNom		Prep	Sust
VSIP3S0	-1		SPS0000	NC00FS0
SVerb			Prep	Sust
-1			SPS0000	NC00FS0
SVerb			Prep	SNom
-1			SPS0000	Z1200S0
SVerb			SPrep	
-1			Z1400S0	
SVerb				
-1				

Figura 5.10 Proceso de verificación de concordancia de la oración del **Caso 5.3**

La **Figura 5.11** muestra la oración del **Caso 5.4** con cambios de accidentes gramaticales de *género* y *número* de algunos de sus elementos.

Caso 5.4 Oración: ¿Cuantos pasajeros tiene cada vuelo de Boston a San Francisco?

Cuantas	pasajero	tiene	cada	vuelo	de	Boston	a	San_Fra
Pronombre	Sustantivo	Verbo	Pronombre	Sustantivo	Preposición	Sustantivo	Preposición	Sustantivo
4	2	5	4	2	7	2	7	2
PT30FP0	NC00MP0	VMIP3S0	PI00NS0	NC00MS0	SPS0000	NC00MS0	SPS0000	NC00MS0

Figura 5.11 Oración del **Caso 5.4** con errores de concordancia introducidos.

El proceso completo de verificación de concordancia de la oración del **Caso 5.4** se muestra en la **Figura 5.12**.

Cuantas	pasajero	tiene	cada	vuelo	de	Boston	a	San_Fra
4	2	5	4	2	7	2	7	2
Pron	Sust	Verb	Pron	Sust	Prep	Sust	Prep	Sust
PT30FP0	NC00MS0	VMIP3S0	PI00NS0	NC00MS0	SPS0000	NC00MS0	SPS0000	NC00MS0
SNom		Verb	Pron	Sust	Prep	Sust	Prep	Sust
-1		VMIP3S0	PI00NS0	NC00MS0	SPS0000	NC00MS0	SPS0000	NC00MS0

SNom	Verb	SNom	Prep	Sust	Prep	Sust
-1	VMIP3S0	Z1000S0	SPS0000	NC00MS0	SPS0000	NC00MS0
SNom	SVerb		Prep	Sust	Prep	Sust
-1	Z1200S0		SPS0000	NC00MS0	SPS0000	NC00MS0
SNom	SVerb		Prep	SNom	Prep	Sust
-1	Z1200S0		SPS0000	Z1000S0	SPS0000	NC00MS0
SNom	SVerb		SPrep		Prep	Sust
-1	Z1200S0		Z1400S0		SPS0000	NC00MS0
SNom	SVerb				Prep	Sust
-1	Z1200S0				SPS0000	NC00MS0
SNom	SVerb				Prep	SNom
-1					SPS0000	Z1000S0
SNom	SVerb				SPrep	
-1	Z1200S0				Z1400S0	
SNom	SVerb					
-1	Z1200S0					

Figura 5.11 Proceso de verificación de concordancia de la oración del **Caso 5.4**

En la **Figura 5.13** se muestra el árbol sintáctico creado por *Freeling* 4.0 (en su demo online) de la oración del **Caso 5.4** y uno de los arboles creado por el algoritmo presentado en este trabajo se muestra en la **Figura 5.14.** Para generar el árbol de la **Figura 5.13**, se seleccionó la opción *Full Parsing* de Freeling, que etiquetó la oración y un árbol, en el cual se ve cómo agrupa los conjuntos de elementos. Freeling 4.0 no nos muestra gráficamente una reducción completa de la oración por lo que no queda clara la manera en cómo construye el árbol. En la **Figura 5.13** se ve cómo Freeling realiza la agrupación de los elementos en Sintagmas nominales y otros grupos de elementos. Lo que no se visualiza claramente en los resultados que muestra Freeling es la manera en que a partir de estos elementos llega a la conclusión de que esta oración es un *grupo verbal*, dado que no se visualizan algunos niveles del árbol que permitirían comprender la relación entre los sintagmas nominales, el sintagma preposicional y el verbo de la oración. El resultado final *group-verb* (*grupo verbal*) indica que la oración posee al menos un verbo. En la **Figura 5.14** se muestra que nuestro algoritmo realiza una reducción completa de la oración mostrando todas las relaciones entre las palabras.

Algunas semejanzas y diferencias se perciben entre los árboles de las **Figuras 5.13** y **5.14**. Entre las semejanzas están la identificación correcta de los sintagmas preposicionales ***de Boston*** y ***a San Francisco***, además de identificar el sintagma nominal ***Cuántos pasajeros***. Sin embargo, una de las diferencias de nuestro árbol con relación al que presenta Freeling 4.0 es que este último no es capaz de establecer la reducción del verbo y de mostrar claramente la relación de éste con los sintagmas de la oración, a pesar de llegar a una raíz correcta. Sin embargo, no se puede realizar una comparativa satisfactoria entre nuestro algoritmo y Freeling 4.0, dado que no son herramientas semejantes. Freeling 4.0 no realiza la detección de incongruencias de accidentes gramaticales, no

muestra correctamente la reducción de los elementos de la oración y la causa principal por la cual no podemos establecer una comparativa se mostrará en el análisis del **Caso 5.5**.

Si tomamos la misma oración del **Caso 5.4** y se ordenan aleatoriamente sus elementos, tenemos ahora una oración que no tiene sentido, como se observa en el **Caso 5.5**.

Caso 5.5 Oración: ¿San Francisco cuantos pasajeros cada de Boston tiene vuelo a?

La oración del **Caso 5.5** al ser evaluada, al inicio el analizador sintáctico versión Mellado encuentra reglas que reducen los elementos de ésta, sin embargo, se llega a un punto donde no encuentra reglas que reduzcan más elementos y esto ocurre sin llegar a la condición de término de la reducción. Por lo tanto no se encuentra una reducción válida para la oración como nos muestra la **Figura 5.15** y consecuentemente no se puede construir ningún árbol sintáctico. Freeling por su parte sí nos muestra un árbol para la misma oración como se observa en la **Figura 5.16**.

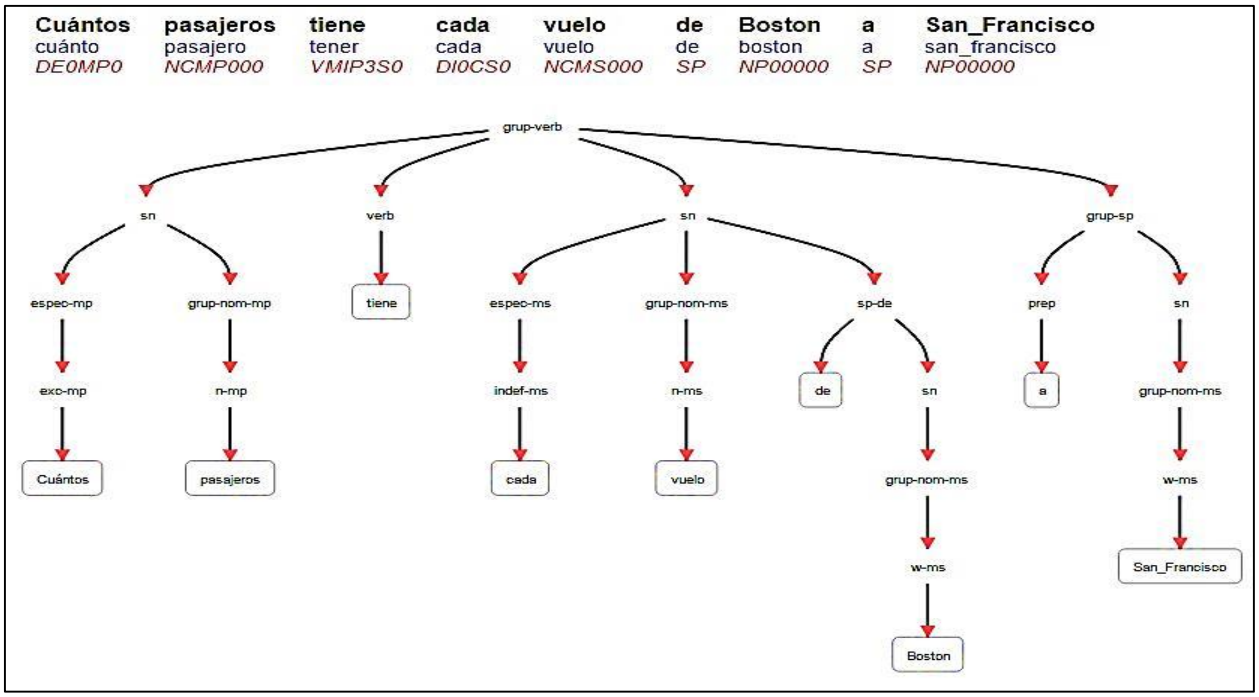


Figura 5.12 Árbol generado por Freeling 4.0 de la oración del **Caso 5.3**.

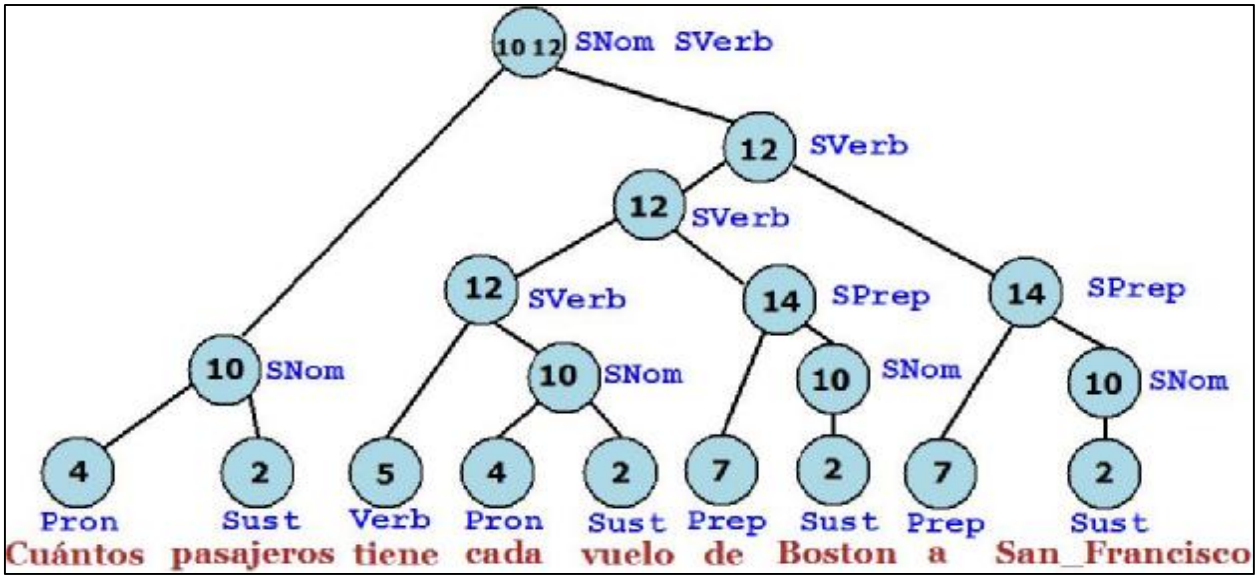


Figura 5.13 Árbol generado por nuestro algoritmo de la oración del **Caso 5.3**.

San_Fra	cuantos	pasajeros	cada	de	Boston	tiene	vuelo	a
2	4	2	4	7	2	5	2	7
Sust	Pron	Sust	Pron	Prep	Sust	Verb	Sust	Prep
Sust	SNom		Pron	Prep	Sust	Verb	Sust	Prep
SNom			Pron	Prep	Sust	Verb	Sust	Prep
SNom			SNom	Prep	Sust	Verb	Sust	Prep
SNom				Prep	Sust	Verb	Sust	Prep
SNom				Prep	SNom	Verb	Sust	Prep
SNom				SPrep		Verb	Sust	Prep
SNom						Verb	Sust	Prep
SNom						SVerb	Sust	Prep
SNom						SVerb	SNom	Prep
SNom						SVerb		Prep
No Reduce !!!								

Figura 5.15 Proceso de reducción de concordancia de la oración del **Caso 5.5**

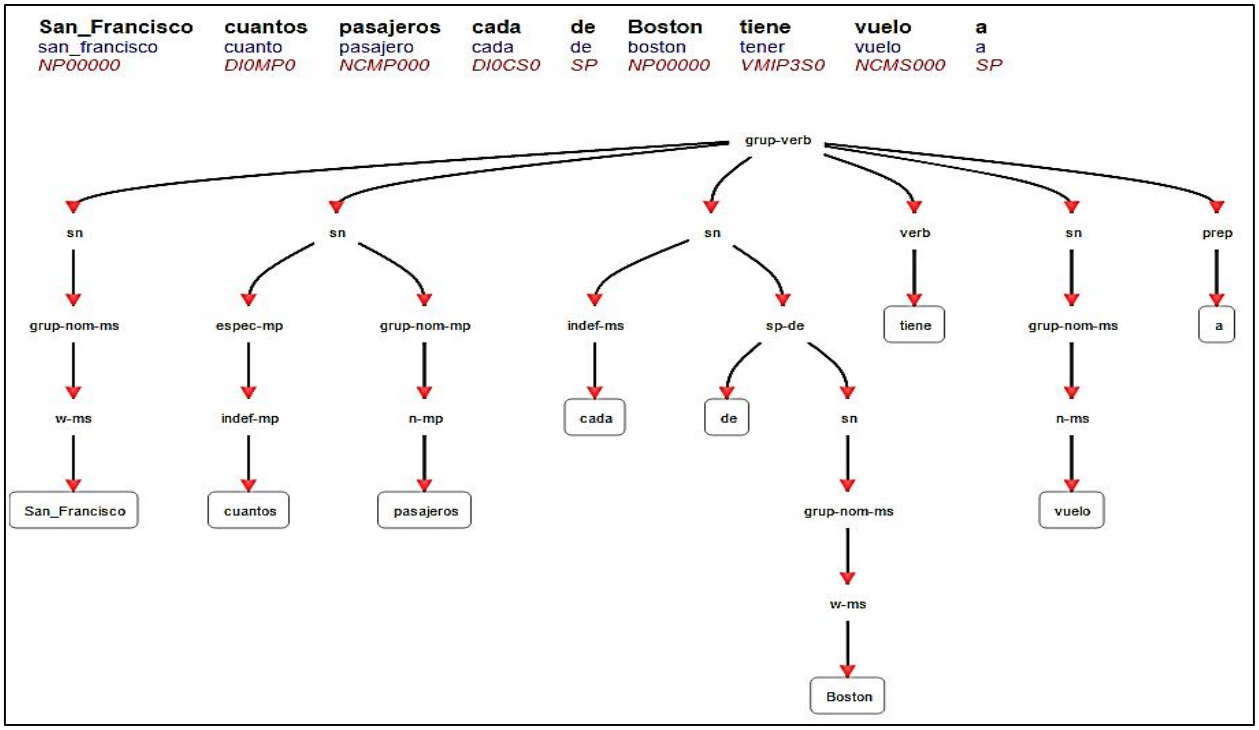


Figura 5.16 Árbol generado por Freeling 4.0 de la oración del **Caso 5.5**.

Capítulo 6 Conclusiones y Trabajos Futuros

En este capítulo se presentan las conclusiones a las cuales se llegó este proyecto después de haber realizado la complementación del analizador sintáctico con la verificación de congruencia de los accidentes gramaticales. Además se mencionan algunas áreas de oportunidad en el tema descubiertas en el desarrollo de este proyecto con relación al procesamiento del lenguaje natural del idioma español.

6.1 Conclusiones

En este trabajo se diseñó e implementó un analizador sintáctico mejorado con la verificación de congruencia de los accidentes gramaticales con lo que se pudo cumplir el objetivo general del proyecto.

Como resultado del proyecto, se concluye lo siguiente:

1. Se complementó un analizador sintáctico con la verificación de concordancia entre accidentes gramaticales.
2. Se optimizó el lexicón de (Aguirre, 2014) al definir un formato homogéneo de las etiquetas de las palabras que contenía, además de agregarle al mismo cerca de 200,000 nuevas palabras, obtenidas de documentos de la RAE. Esta mejora permite que el lexicón sea más eficiente en las próximas fases del análisis sintáctico.
3. Se mejoró el etiquetado del análisis léxico, permitiendo el correcto etiquetado de claves, fechas y horas.
4. Se construyó un método para representar los Arboles Sintácticos generados a partir de la reducción sintáctica de las oraciones.
5. Se logró diseñar el método de verificación de concordancia y un compendio de reglas de concordancia las cuales indican cómo se deben relacionar las palabras dentro de la oración.

6.2 Trabajos Futuros

Dada la magnitud que posee el área de lenguaje natural del español. Aún en los días actuales muchas incógnitas son lanzadas cada día. Muchas de estas incógnitas son áreas de oportunidad a partir de las cuales se pueden derivar otros proyectos. Proyectos que pueden ayudar a complementar este trabajo, son:

- Reducir el tiempo de ejecución de la reducción sintáctica. El algoritmo que propuso (Mellado, 2014) realiza una búsqueda exhaustiva de todos los árboles sintácticos posibles de una oración, esto implica un tiempo de ejecución muy largo, el uso de programación dinámica con estrategias de *branch and bound* ayudarían mucho en la reducción del tiempo de procesamiento y permitiría encontrar los arboles sintácticos de la oración con más eficiencia.
- Actualización del lexicón actual, dado que el lenguaje dentro de un idioma tiende a crecer cada día con nuevas palabras que surgen fruto de la experiencia humana, desarrollar un lexicón que contenga todas las palabras de la lengua española puede ser un propósito difícil de alcanzar, pero haciendo uso del diseño actual del lexicón este siempre se puede mejorar. Esto permitiría contar con un correcto etiquetado léxico de las oraciones que son introducidas en el analizador sintáctico.
- Implementar un módulo que permita al analizador sintáctico ser capaz de procesar locuciones.
- Ampliar la funcionalidad del analizador sintáctico para que permita verificar todas las incongruencias de accidentes gramaticales que se presentan en el español y procesar excepciones de las reglas gramaticales.
- Realizar un estudio más profundo de la concordancia gramatical del español y ampliar las reglas de concordancia ya existentes con reglas que contemplen las excepciones del español, además de implementar heurísticos para la verificación de concordancia.

Anexo A Fases del Procesamiento de Lenguaje Natural

En este anexo se detallan conceptos específicos que ayudarán a comprender mejor los temas abordados en este trabajo y conocer elementos que componen el área de procesamiento de lenguaje natural.

A.1 Análisis Léxico

El análisis léxico es el primer paso del PLN en el traductor. El proceso comienza con la información léxica, está contenida en el lexicón. El lexicón se puede considerar como el conjunto de unidades léxicas pertenecientes a un sistema lingüístico. Dicha información consta de la etiqueta relativa a la categoría gramatical de cada unidad lingüística (sustantivo, verbo, pronombre, etc.) y una serie de otras etiquetas correspondientes a los diferentes rasgos de subcategorización que hacen posible que cada unidad lingüística seleccione otra u otras a la hora de combinarse, formando los distintos sentidos que una oración pueda tener en la lengua (Aho, Seit, & Ullman, 1998).

A.1.1 Lexema

Unidad léxica abstracta que no puede descomponerse en otras menores, aunque si combinarse con otras para formar compuestos, y que posee un significado definible por el diccionario, no por la gramática. Por ejemplo: fácil es el lexema básico de facilidad, facilitar, fácilmente.

A.1.2 Léxico

El conjunto de los morfemas de una lengua, junto con raíces complejas o palabras pre-formadas (o sea que no se arman en forma productiva), modismos y otras frases establecidas. Éstas son las estructuras lingüísticas que un hablante sabe cómo unidades completas y que puede usar sin tener que determinar sus significados a base de sus partes integrantes. Un diccionario (que a veces también se llama léxico) es un libro que exhibe elementos del léxico dc una lengua, especialmente palabras, con una indicación breve de sus significados y usos. Contrástese con gramática: sintaxis, morfología, fonología, semántica.

A.1.3 Lexicón

Serie ordenada de palabras de una lengua, una persona, una región, una materia o una época determinadas.

A.1.4 Ambigüedad Léxica

La ambigüedad léxica es aquella que se presenta en la categoría gramatical de un vocablo. Es decir, un vocablo puede tener más de un rol gramatical en diferentes contextos.

A.2 Análisis Sintáctico

Teniendo en cuenta todos los niveles de análisis conocidos para el tratamiento del lenguaje natural, la sintaxis ha sido durante mucho tiempo y aún sigue siendo el nivel al que la lingüística ha centrado más su atención. Por ejemplo: El procesamiento semántico funciona sobre los constituyentes de la oración. Si en el proceso no existe un paso de análisis sintáctico, el sistema semántico debe identificar sus propios constituyentes. Con el análisis sintáctico, se restringe enormemente el número de constituyentes a considerar por el análisis semántico, mucho más complejo y menos fiable. Podemos decir que el análisis sintáctico es mucho menos costoso computacionalmente hablando que el análisis semántico (que requiere inferencias importantes). Por lo que el uso de éste conlleva a un considerable ahorro de recursos y una disminución de la complejidad del sistema. Aunque con frecuencia se puede extraer el significado de una oración sin usar hechos gramaticales, no siempre es posible hacerlo (Rich & Knight, 1994).

El análisis sintáctico es un análisis a nivel de palabras, y es mucho más complejo que el análisis léxico. Tiene como función subdividir una oración en forma de *tokens*, que recibe de un analizador léxico (denominados **no terminales** en la definición de la gramática) y determinar si la estructura de la oración es correcta o no. Este proceso determina la clase gramatical de cada palabra y reduce la oración haciendo uso de las reglas gramaticales. A través de este proceso es posible determinar dentro de una oración el sujeto, el predicado, el verbo y los complementos. El análisis sintáctico agrupa a los tokens en estructuras sintácticas (denominadas no terminales en la definición de la gramática). Después de realizar este proceso el analizador sintáctico obtiene un árbol sintáctico (u otra estructura equivalente) en la cual las hojas son los tokens y cualquier nodo, que no sea una hoja, representa un tipo de clase sintáctica. En la **Figura A.1** se muestra claramente la representación de un árbol sintáctico (Fernández, 2000) .

Otra forma de representar el análisis sintáctico (tomando el ejemplo anterior) se puede apreciar en la **Figura A.2**.

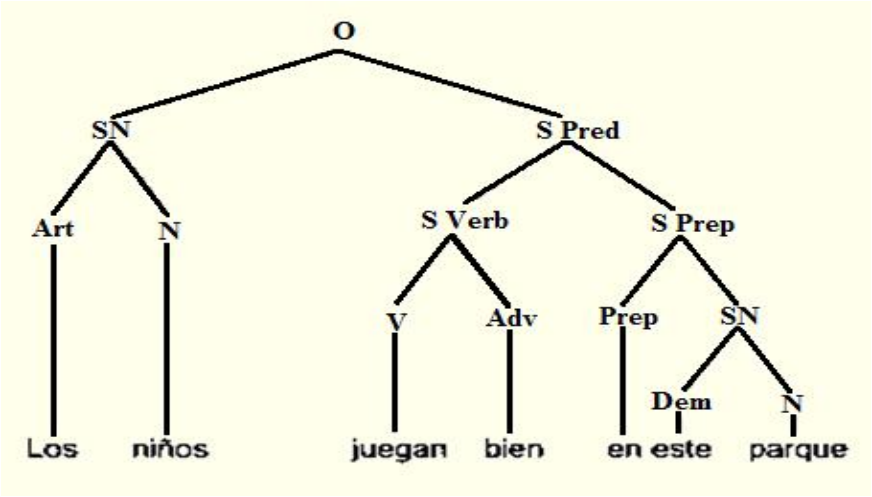


Figura A.1 Representación de un árbol sintáctico según el enfoque de constituyentes.

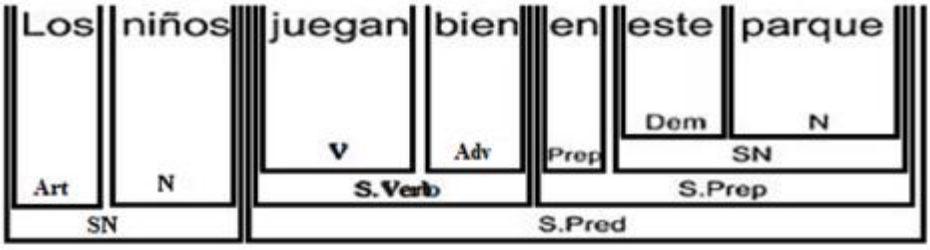


Figura A.2 Representación de un análisis sintáctico según el método distribucional.

Algunas de las funciones que el analizador sintáctico realiza son las siguientes:

- Generación del árbol sintáctico.
- Corrección de errores sintácticos.
- Resolución de la ambigüedad sintáctica.
- Identificación de los diálogos.
- Conversión a la estructura canónica.

El análisis sintáctico se puede clasificar en dos grandes grupos, el análisis sintáctico ascendente (Bottom-Up) y el análisis sintáctico descendente (Top-Down). Ambos métodos de análisis sintáctico realizan el análisis tomando un símbolo a la vez, de izquierda a derecha, haciendo uso de las reglas de la gramática formal.

A.2.1 Análisis Sintáctico Ascendente (Bottom-Up)

En el análisis sintáctico ascendente se construye el árbol sintáctico a partir de las hojas, paso a paso hasta llegar a la raíz. Es decir a partir de los distintos tokens de una sentencia a analizar y gracias a reducciones se llega al símbolo inicial de la gramática. Las reducciones que se efectúan en sentido contrario a las producciones de la gramática. El principal problema que se plantea en el análisis ascendente es el del retroceso, que se traduce en la elección de un pivote para realizar la reducción. Por tanto el análisis sintáctico ascendente comienza tomando cada palabra que constituye la oración de entrada y la etiqueta como terminación del árbol a construir. Este método conoce la parte derecha de las reglas de producción y trata de sustituirla por la parte izquierda que denota la regla que produce (Cervantes, 2005).

A.2.2 Análisis Sintáctico Descendente (Top-Down)

En el análisis sintáctico descendente se construye el árbol sintáctico a partir del símbolo inicial de la gramática, hasta llegar a los distintos tokens, que constituyen la sentencia a analizar. Es decir, se parte del símbolo inicial de la gramática y se van aplicando las distintas producciones, hasta llegar a formar la sentencia. El árbol sintáctico se obtiene a partir de las reglas de derivación de <*expresión*> hasta llegar a la expresión. El orden de derivación es importante, siempre se deriva primero el no terminal situado más a la izquierda según se mira el árbol (derivaciones más a la izquierda). En el ascendente, partiendo de las hojas hasta llegar al axioma obtenemos la inversa de una derivación por la derecha (Cervantes, 2005).

A.2.3 Sintaxis

La sintaxis es el estudio en el cual se determina la correcta combinación de las palabras dentro de las oraciones. La sintaxis se encarga de los problemas relacionados con el orden que las palabras deben llevar en la oración y las variaciones que éstas pueden sufrir (sean de género, numero, entre otras) además de las diferentes funciones que pueden desarrollar estas palabras en la oración.

A.2.4 Ambigüedad Sintáctica

La ambigüedad sintáctica, también conocida corno estructural, es aquella que se presenta en oraciones, de tal manera que éstas puedan ser representadas por más de una estructura sintáctica.

A.2.5 Árbol de Constituyentes

Un árbol dc constituyentes es una estructura de datos que permite categorizar una oración en sus partes dc oración. En el llamado sistema o método de constituyentes la principal operación lógica es la inclusión de elementos en conjuntos, así éstos pertenecen a una oración o a una categoría. Según esta aproximación, una oración es segmentada en constituyentes, cada uno de los cuales es consecuentemente segmentado. Así, esto favorece un punto de vista analítico.

A.2.6 Árbol de Dependencias

Un árbol de dependencias es una estructura dc datos que permite obtener las relaciones dc dependencia sintáctica entre un núcleo y conjunto de modificadores. La aproximación de dependencias se centra en las relaciones entre las unidades sintácticas últimas, es decir, en las palabras. La principal operación aquí consiste en establecer relaciones binarias. Según esta idea, una oración se construye de palabras, unidas por dependencias.

A.3 Morfología

La Morfología es la rama de la lingüística que estudia la estructura interna de las palabras para delimitar, definir y clasificar sus unidades, las clases de palabras a las que da lugar (morfología flexiva) y la formación de nuevas palabras (morfología léxica). La palabra 'morfología' trataba originalmente de la forma de las palabras, aunque en su acepción más moderna estudia fenómenos más complejos que la forma en sí. La descripción gramatical de todas las lenguas del mundo se divide, por convención, en dos secciones: morfología y sintaxis. La relación entre las dos es la siguiente: La morfología explica la estructura interna de las palabras mientras que la ***sintaxis*** describe cómo las palabras se combinan para formar sintagmas, oraciones y frases (Mendoza & Palma , 2010). La unidad mínima de la morfología es el morfema y la unidad máxima la palabra o pieza léxica. Dentro de la morfología podemos tener:

- Palabras mono morfemáticas: *mar.*
- Palabras poli morfemáticas: *mar –es*, *mar-in-o-s.*

Por ejemplo, la Morfología analiza la palabra jardineros descomponiéndola en raíz y morfemas como vemos en la **Figura A.3**.

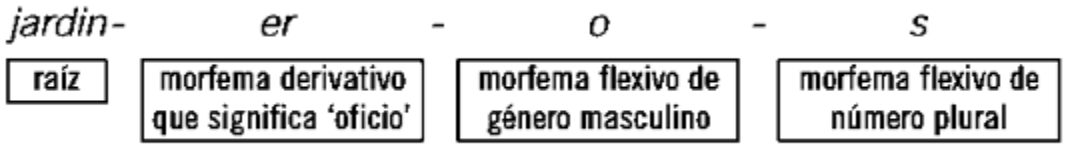


Figura A.3 Ejemplo de descomposición Morfológica.

A.3.1 Morfología Flexiva

La morfología flexiva es la que estudia las variaciones de las palabras que tienen consecuencias en la concordancia y en otros aspectos de las construcciones sintácticas. Las palabras flexionadas (escribo, escribió, escribiré) constituyen variantes de una misma unidad léxica (escribir).

La menor unidad de la morfología es el morfema. El morfema no puede segmentarse en unidades menores a las que corresponda algún significado. Son también llamados de bases o lexemas y contienen información léxica, la que proporciona el diccionario para el verbo cantar o el sustantivo mesa.

Ejemplo: *cant-**é**; mesa-**s***

A.3.2 Morfología Léxica o Derivativa

La morfología léxica también llamada morfología derivativa, tradicionalmente recibe el nombre de *formación de palabras*. La morfología léxica estudia la estructura de las palabras y las pautas que permiten formarlas o derivarlas a partir de otras.

Ejemplo: *escri**bir;** escri**to;** escri**tor;** escri**tura,***

A diferencia de las palabras flexionadas ***(escribo, escribió, escribiré)*** que constituyen variantes de una misma unidad léxica, las palabras derivadas ***(escritor, escritura)*** no constituyen variantes de las formas de las que proceden, sino nuevas palabras obtenidas de ellas.

A.3.3 Análisis Morfológico

El Análisis Morfológico consiste en determinar la forma, clase o categoría gramatical de cada palabra de una oración. No se debe confundir con el análisis sintáctico en el que se determinan las funciones de las palabras o grupos de palabras dentro de la oración. Más bien el Análisis Morfológico es el que detecta las relaciones que se establece entre las unidades mínimas que forman una palabra

(sufijos, prefijos) y la relación con el léxico, siendo este un conjunto de información sobre cada palabra que el sistema utiliza para el procesamiento.

A.4 Análisis Semántico

La tarea de interpretar el significado de lo que se está diciendo puede ser costosa en procesamiento y complicada. Es decir, la semántica hace referencia a lo que significan las palabras por sí mismas sin considerar el uso en un tema.

A.4.1 Analizador Semántico

Se trata de determinar el tipo de los resultados intermedios, comprobar que los argumentos que tiene un operador pertenecen al conjunto de los operadores posibles, y si son compatibles entre sí, etc. En definitiva, comprobará que el significado de lo que se va leyendo es válido.

A.4.2 Ambigüedad Semántica

La ambigüedad semántica es aquella que se presenta en una estructura gramatical, de tal manera que ésta puede expresar diferentes sentidos dependiendo del contexto local, el tópico global y el mundo pragmático en el que se manifiesta.

Anexo B Mejora del Analizador Léxico y Lexicón

En este anexo se puntualiza el desarrollo de dos actividades que ayudan a realizar de manera correcta y eficiente la verificación de congruencia de accidentes gramaticales entre palabras de una oración. Las actividades que se detallan a continuación se enfocaron en la mejora del analizado Léxico desarrollado por (Aguirre, 2014) y homogenización y la mejora de su lexicón para aprovecharlo en este trabajo.

B.1 Mejora del Analizador Léxico

Para obtener un mejor rendimiento en el funcionamiento del analizador sintáctico se procedió a mejorar la etapa anterior, el análizador léxico. Dentro del analizador léxico (versión Aguirre, 2014) se realizó un análisis exhaustivo del código buscando comprender como se realizaba el etiquetado de las palabras contenidas en las oraciones de los corpus. Algunos problemas que se encontraron fueron problemas con el etiquetado de claves, nombres propios, fechas y números en forma de datos. Realizando investigaciones con el motor de búsqueda de palabras que la rae provee en su página web se pudo determinar la categoría gramatical a la que pertenecen estas palabras, y quedaron de la siguiente manera:

- Claves (Sustantivos)
- Nombres propios (Sustantivos)
- Fechas (Sustantivos)
- Horas (Sustantivos)
- Números (Adjetivos Numerales)

Otros problemas se descubrieron como por ejemplo el hecho de que algunas palabras presentes en el lexicón poseían más de una categoría gramatical y se procedió a actualizar esta información dentro del Lexicón. Esto se reflejó en un análisis léxico más coherente y consecuentemente una mejor respuesta del analizador sintáctico en su fase de reducción sintáctica.

Ej.: ¿**Cuántas** ventas se realizaron en el año de 1992?

En el ejemplo anterior se detectó que la palabra **Cuántas** que es un pronombre se puede comportar como adjetivo.

Para que se produjera un etiquetado léxico más flexible que generara resultados que mejorarían la reducción sintáctica de oraciones se arreglaron los problemas que se generaban y dificultaban la reducción de la oración. El analizador léxico anterior no era capaz de identificar estos elementos debido a que se desconocía información respecto a ellos, principalmente su categoría gramatical. Realizando investigaciones con el motor de búsqueda de palabras que la RAE provee en su página web se pudo determinar la categoría gramatical a la que pertenecen este tipo de palabras. A continuación se explica cómo se trataron los problemas dentro del analizador léxico.

B.1.1 Claves

Ej.:

¿Cuál es el título del libro con identificador **TC4203**?

¿Qué empleado tiene como identificador **H-B39728F**?

Según la RAE las claves son sustantivos y por lo tanto deben etiquetarse con esta categoría gramatical. Para resolver el problema se desarrolló dentro del analizador léxico un código capaz de filtrar las palabras que son claves y si esta tiene el formato de cadena como por ejemplo **TC4203, EKC6T23** o **345RT**, su categoría debe de ser sustantivo. El código en si hace un análisis carácter a carácter para identificar secuencias de letras y números y con esto identificar claves.

B.1.2 Nombres propios

Ej.:

¿Qué autores viven en la ciudad de **Oakland**?

¿A qué almacén pertenece la siguiente dirección 679 **Carson st.** ?

Con relación a los nombres propios se desarrolló un código capaz de identificar los nombres propios dentro de las oraciones. Este código consiste en identificar palabras que contengan su letra inicial en mayúscula denotando así los nombres propios como por ejemplo **Oakland, Barcelona, México.** En este caso se agregó al código un filtro con el fin de evitar que palabras clave como órdenes se etiqueten como sustantivos. A continuación se muestra una lista de órdenes que están presentes dentro de los corpus las cuales no se deben etiquetar como sustantivos.

Cuántos	Muestra	Puedo
Cuántas	Mostrar	Primera
Cuál	Quién	Me
Dame	En	Sólo
Lista	Obtener	Muéstrame
Qué	Nombra	A
Puedes	Encuentra	
Total	Desde	
Despliega	Visualiza	

Generalmente estas palabras aparecen al inicio de las oraciones.

B.1.3 Fechas

Ej.:

¿Qué trabajador tiene su fecha de contratación como **miércoles 14 de septiembre de 1994**?
¿Cuantos ejemplares del libro **The Busy** se vendieron el **14 de Septiembre**?

Para el problema con fechas se desarrolló un código capaz de identificar Fechas dentro de las oraciones teniendo formatos como por ejemplo **13/02/1991** o **14 de Septiembre de 1994.** Actualmente el código es capaz de evaluar esta secuencia y determinarla como un sustantivo. Sin embargo la fecha se puede presentar de otras formas, con otros formatos y con el uso de varias palabras. Como podemos ver en la **Tabla B.1** un conjunto de palabras consecutivas puede representar una fecha. Realizando un estudio del corpus de consultas y de los principales formatos de fecha se determinaron y propusieron varios formatos de fecha los cuales podemos ver en la **Tabla B.1**. Para validar el trabajo se probaron estos formatos con el demo online de Freelig y se obtuvieron los siguientes resultados. La gran mayoría de los formatos propuestos tienen una correspondencia en el demo Freelig. La gran mayoría de ellos fueron aceptados por Freelig como formatos de fecha. Colocando uno de estos ejemplos en Freelig se obtuvo resultado de la **Figura B.1**.

En el caso del analizador léxico y con el fin de mejorar el rendimiento del análisis es necesario agrupar todas estas palabras como podemos ver en el demo de Freelig y conformar la fecha para posteriormente etiquetarlas como sustantivo. Buscando resolver este problema y teniendo en cuenta los formatos anteriores se hizo un análisis de las palabras contenidas en estos formatos buscando obtener su categoría gramatical y con esto identificar posteriormente cada uno de los elementos que conforma el formato. En las **Tablas B.2, B.3** y **B.4** se muestran los formatos desglosados elemento por elemento junto con su categoría gramatical.

Tabla B.1 Formatos propuestos para las fechas y comprobación con Freeling.

Formato	Correspondencia con Freelig
14/09/1994	**SI**
14 de Septiembre	**SI**
14 de Septiembre de 1994	**SI**
Septiembre de 1994	**SI**
Miércoles 14 de Septiembre de 1994	**SI**
mes de Septiembre	**SI**
mes de Septiembre de 2014	**SI**
año 2014	**SI**
lunes	**SI**
primavera, verano, otoño, invierno	**NO**

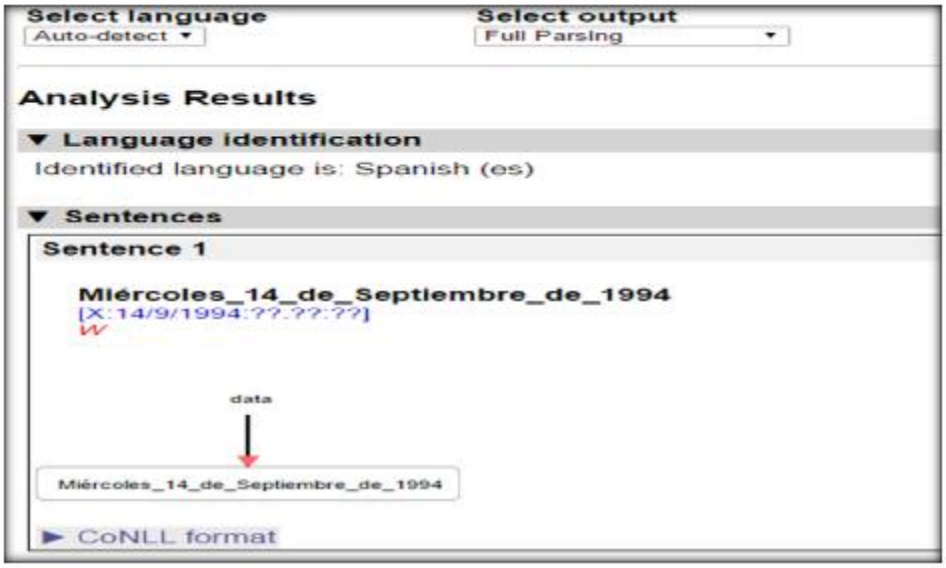


Figura B.1 Verificación del formato de fecha con Freeling 4.0.

Tabla B.2 Formatos de Fechas 1 al 6.

Formato	*Sust*	*Adj*	*Prep*	*Sust*	*Prep*	*Adj*
1	primavera					
2	miércoles					
3	miércoles	14	de	septiembre	de	1994
4		14	de	septiembre	de	1994
5		14	de	septiembre		
6				septiembre	de	1994

Tabla B.3 Formatos de Fechas 7 y 8.

Formato	*Sust*	*Prep*	*Sust*	*Prep*	*Adj*
7	mes	de	septiembre	de	1994
8	mes	de	septiembre		

Tabla B.4 Formato de Fecha 9.

Formato	*Sust*	*Adj*
9	año	1994

Teniendo en cuenta este análisis que se realizó. Se elaboró el diagrama que muestra en la **Figura B.2** En este diagrama se puede apreciar el procedimiento a seguir para identificar los anteriores formatos.

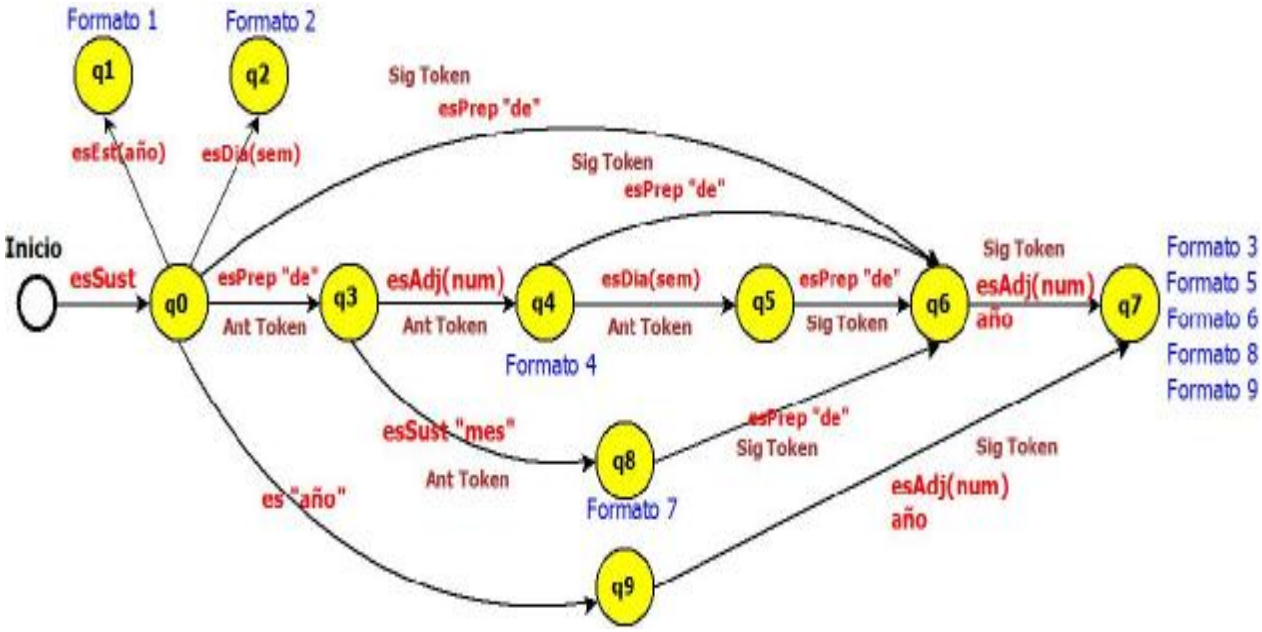


Figura B.2 Procedimiento para identificar formatos de fechas.

B.1.4 Horas

Ej.: Dame una lista de todos los vuelos que lleguen antes de **7:00 am.**

Con relación a las horas se realizó prácticamente el mismo procedimiento que en las fechas y se contrastaron algunos formatos con Freelig. El resultado se muestra en la **Tabla B.5.**

.

Tabla B.5 Formatos de Hora Propuestos.

Formato	Correspondencia con Freelig
mediodía, medianoche	**SI**
12:00pm	**SI**
07:00am	**SI**

En este caso se desarrolló realizar un código para identificas las secuencias anteriores.

B.1.5 Números

Ej.: ¿Qué trabajador tiene como nivel de trabajo **227**?

Dentro del Análisis Léxico existe un código capaz de identificar los números dentro de las oraciones. Este código consiste en identificar secuencias de números y clasificarlos gramaticalmente como adjetivos. Algunos ejemplos son **19989**, **227**, **34**.

B.2 Proceso de Mejora del Lexicón

Dentro de la actividad de Ampliación del Lexicón se estudió el lexicón propuesto por [Aguirre, 2014]. Dicho lexicón contenía 637,282 palabras validadas por la Real Academia Española (RAE). La mejora se realizó con el fin de comprender su estructura y realizar las adaptaciones necesarias que se pudiera usar dicho lexicón en este trabajo. Para mejorar el funcionamiento del programa se desarrolló una actualización del anterior lexicón con palabras obtenidas de la RAE. En la **Tabla B.6** se muestra una comparativa del número de palabras dentro de los Lexicones, el anterior y el nuevo.

Para actualizar el Lexicón se usaron un conjunto de 80000 archivos obtenidos de la RAE que contenían palabras y su información respectiva como su categoría gramatical y su género. Estos archivos se encontraban en HTML. En la **Figura B.3** se muestra uno de estos archivos.

Tabla B.6 Comparativa entre los elementos de los Lexicones.

ELEMENTO	CANTIDAD DE PALABRAS Lexicón Anterior	CANTIDAD DE PALABRAS Nuevo Lexicón
Adverbios	185	85
Adjetivos	31,451	88,256
Artículos	9	9
Conjunciones	18	18
Preposiciones	27	27
Interjecciones	200	200
Pronombres	343	175
Sustantivos	107,638	156,607
Verbos	497,681	497,681
TOTAL:	**637,282**	**743,058**

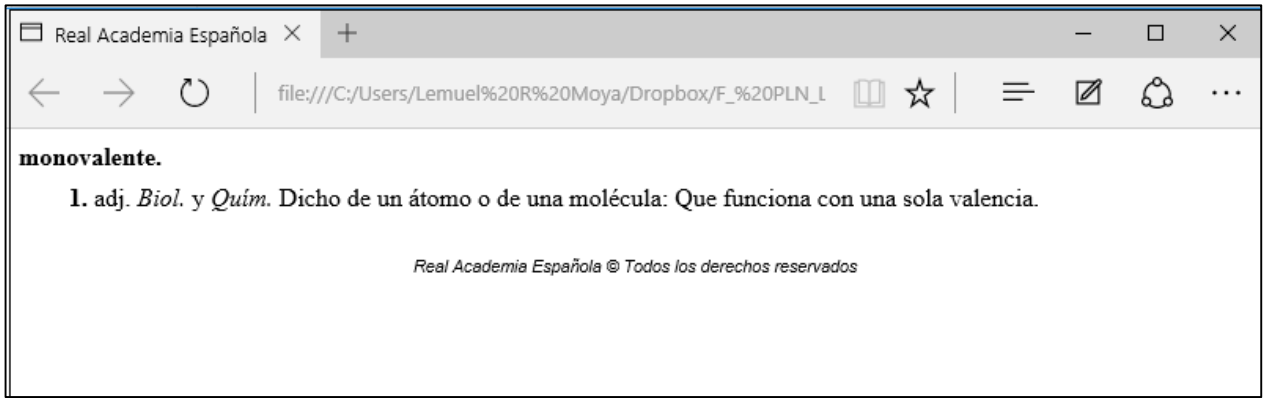


Figura B.3 Archivo HTML de la palabra monovalente.

Para obtener esta información se realizó un código capaz de extraer esta información de cada uno de estos archivos. Este código busco principalmente obtener la palabra su categoría gramatical y dado que en algunos archivos no se encontraba toda la información necesaria en algunas palabras se pasó a definir el número con la aplicación de reglas de pluralización y también el género de la palabra caso esta lo requiriera. También por medio de la información obtenida de las etiquetas y otra definida dentro del programa se pudo construir las etiquetas asociadas a cada una de las palabras las cuales se pudieron ingresar posteriormente al nuevo lexicón. En la **Figura B.4** se muestra un diagrama con el procedimiento usado para evaluar los archivos.

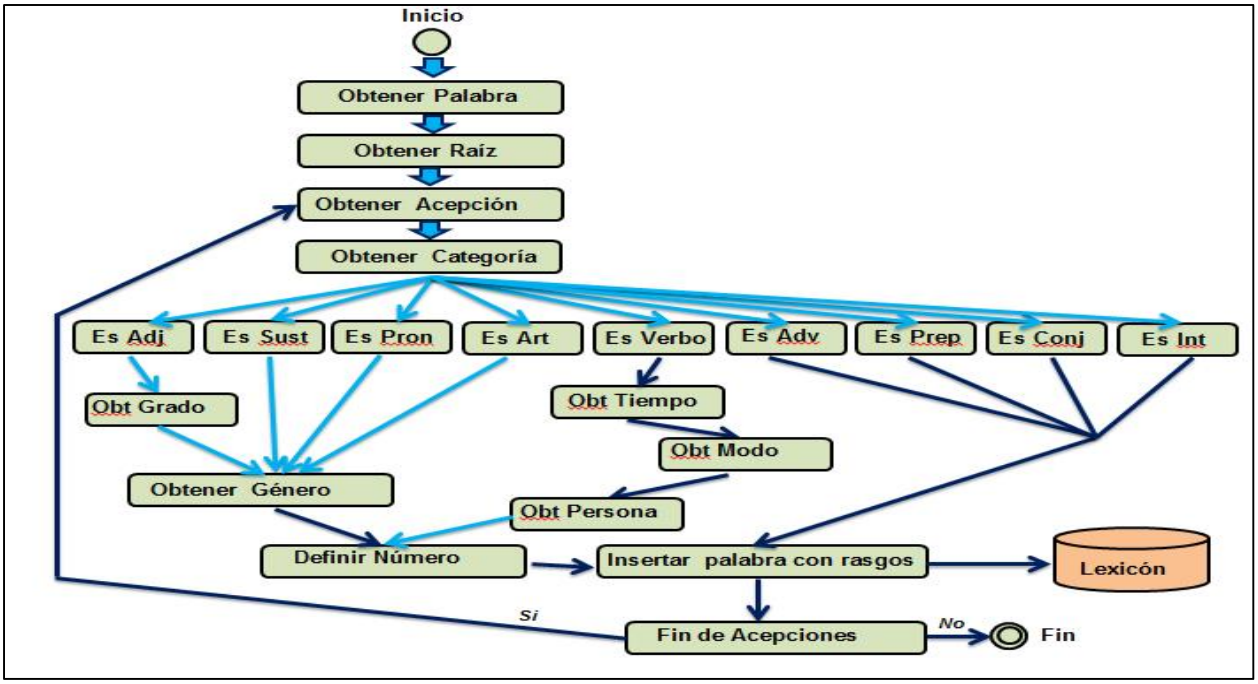


Figura B.4 Procedimiento para evaluar los archivos HTML.

Posteriormente una vez terminado el programa se pasó a analizar la información contenida en el nuevo lexicón y se contrasto con las bases en la base de datos de la RAE en internet a través del módulo de consultas online de la RAE, además de usar información obtenida en libros específicos y manuales de la RAE que respaldan los elementos del nuevo lexicón (Bosque & Demonte, 1999). Este trabajo más minucioso se realizó para Pronombres, Artículos, Adverbios, Preposiciones Conjunciones e Interjecciones (RAE, Nueva Gramatica de la Lengua Española (Manual), 2009). Los verbos se tomaron del trabajo realizado por Aguirre en 2014. Los Sustantivos y Adjetivos se tomaron del Lexicón anterior y se unificaron con las nuevas palabras obtenidas de los archivos evaluados de la RAE.

Cabe resaltar que se redujo el número de tablas del Lexicón de 12 a 10 tablas dado que se unificaron 4 tablas en 2. Anteriormente tanto los sustantivos como los adjetivos se separaban en dos tablas con respecto al número que estos tenían, o sea existía una tabla para el singular y otra para el plural en ambos casos. Esta acción fue posible dado que el número de una palabra se puede identificar por medio de su etiqueta En la **Tabla B.7** se muestra algo se información relevante del nuevo Lexicón y la nueva información agregada:

Tabla B.7 Cantidad de palabras Nuevas del Lexicón.

Sustantivo	Adjetivos
Total Viejo Lexicón: **107,638**	**Total Viejo Lexicón:** 62,922
Nuevas Palabras: **48,969**	**Nuevas Palabras:** 25,334
Total Actual Lexicón: **156,607**	**Total Actual Lexicón:** 88,256

B.2.1 Homogenización de etiquetas del lexicón

Además de ampliar el lexicón, también se homogenizaron las etiquetas de éste, lo cual consistió en colocar el mismo formato de etiqueta a cada una de las palabras del lexicón, con la finalidad de reducir el esfuerzo computacional en futuras fases del proyecto. En la **Tabla B.8** se muestran los nuevos formatos.

El formato de la etiqueta consiste en una secuencia de caracteres que tienen la finalidad de almacenar información de cada una de las palabras del lexicón. Este nuevo formato consta de 7 caracteres, donde el primero identifica el tipo de palabra, el segundo y tercero almacenan información sobre el tipo de palabra, ya el penúltimo y antepenúltimo nos muestran las variación de género y numero de cada palabra.

Tabla B.8 Formato de Etiquetas del Lexicón.

Elemento	Formato de Etiqueta en el Lexicón
Adverbios	RG_ _ _ _ _ o RN_ _ _ _ _
Adjetivos	AQ _ _ _ _ _
Artículos	DA_ _ M/F S/P _
Conjunciones	CC_ _ _ _ _
Preposiciones	I_ _ _ _ _ _
Interjecciones	SPS_ _ _ _
Pronombres	PR_ _ _ _ _
Sustantivos	NC_ _ M/F S/P _
Verbos	VMII1S0

Las Etiquetas Gramaticales son las que permiten determinar la categoría gramatical de cada una de las palabras que se encuentran en el lexicón, además con la etiqueta gramatical se pueden identificar cuáles son los accidentes gramaticales (género, número, modo del verbal, tiempo del verbo, etc.) de la palabra asociada a esa etiqueta. En las etiquetas gramaticales se incluyen una serie de rasgos recurrentes que están presentes en la mayor parte de las categorías gramaticales de las palabras (Cervantes, 2005). Las etiquetas también presentan algunos rasgos particulares que presentan las palabras los cuales no son tan importantes para el desarrollo de este trabajo.

El nuevo formato que se tienen las etiquetas del lexicón las primera posición hay un indicativo de la categoría gramatical de la palabra. En la segunda posición se encuentra una descripción del tipo de categoría gramatical. En la quinta posición se encuentra el género masculino **M,** femenino **F** y **C** cuando

la palabra es neutra cuanto al género. En la sexta posición se encuentra el número de la palabra y para esto usa **P** para plural, **S** para singular y **N** para palabras que son neutras cuanto al número.

En el caso de los verbos en las primeras dos posiciones de encuentra el indicativo de verbo y su forma. En la tercera posición se encuentra el tipo de verbo. En la cuarta posición el tiempo del verbo y en la quinta la persona. En la sexta posición se encuentra el número del verbo y se usa **P** para plural, **S** para singular y **N** para palabras que son neutras cuanto al número. Finalmente en la séptima posición se representa el género masculino **M,** femenino **F** y **0** cuando la palabra es neutra cuanto al género. En la **Tabla B.9** se ven algunos ejemplos de etiquetas presentes en el lexicón**.**

Tabla B.9 Formato de Etiquetas del Lexicón.

Categoría Gramatical	Etiquetas			
Adjetivo	AQ00CN0	AQ00FP0	AQ00MP0	
	AQ00CS0	AQ00FS0	AQ00MS0	
Articulo	DA00FS0	DA00NS0	DI00FP0	DI00MP0
	DA00MP0		DI00FS0	DI00MS0
Pronombre	PI00CN0	PI00FS0	PI00MP0	PD00FP0
	PI00CS0	PI00FP0	PD00MS0	PD00MP0
	PI00CP0	PI00MS0	PD00FS0	PD00CS0
Sustantivo	NCCN000	NCFN000	NCMN000	
	NCCP000	NCFP000	NCMP000	
	NCCS000	NCFS000	NCMS000	

Anexo C Corpus de Consultas

Para este proyecto se usó un extracto de 69 consultas de los corpus de consultas propuestos por Mellado en su proyecto. Mellado Propuso 3 corpus en tesis, el primer corpus de consulta de CFA que es un extracto de los corpus de consultas de las bases de datos de Northiwnd (Base de datos de control de inventarios, contiene elementos como: artículos, órdenes, empleados, zonas de trabajo, etc), Pubs (Base de datos de registro de publicaciones, contiene datos como: nombres de libros, ISBNs, autores, fechas de publicaciones, editoriales, etc) y Geobase (Base de datos que contiene información geográfica sobre Estados Unidos de América, contiene datos como: estados, ciudades, número de habitantes, carreteras, ríos, lagos,et). El segundo corpus es un extracto de ATIS, traducido al español a partir del original que se encuentra en idioma inglés. La base de datos de ATIS maneja información acerca de aerolíneas, contiene datos como: Ciudades, aeropuertos, aviones, escalas, vuelos, tarifas, itinerarios, etc. El último corpus de consulta Pubs contiene información sobre el registro de publicaciones, tiene datos como: nombres de libros, ISBNs, autores, fechas de publicaciones, editoriales, etc.

C.1 - Corpus de consulta

1. ¿Cuántos vuelos tiene cada aerolínea?
2. ¿Cuántos pasajeros tiene cada vuelo de Boston a San Francisco?
3. Lista el número de pasajeros de cada vuelo.
4. ¿Cuantos ejemplares del libro The Busy se vendieron el 14 de Septiembre?
5. ¿Cuántos autores son de la ciudad de Berkeley?
6. ¿Cuál es el número de ventas realizadas el 14/09/1994?
7. ¿Cuál es el libro más barato de tipo Business?
8. ¿Cuál es el promedio de población por estado?
9. ¿Cuántos estados hay en Colorado?
10. ¿Cuál es la ciudad más grande en un estado con río?
11. ¿Cuál es el estado más grande?
12. ¿Cuál es la ciudad más grande en Kansas?
13. ¿Qué estado tiene la mayor población?

14. Dame una lista de todos los tipos de aeronaves.
15. Lista todas las restricciones de vuelos.
16. Lista tarifas de viaje redondo.
17. Lista las categorías de aeroplanos.
18. Lista las descripciones de transporte.
19. Lista los tipos de aeroplanos para vuelos desde Fort Worth a Washington.
20. Nombra todos los aeropuertos.

21. Muéstrame el costo del vuelo 9.
22. Lista todos los vuelos.
23. Muéstrame los vuelos desde Oakland a Baltimore llegando después del mediodía.
24. Lista las tarifas para todos los vuelos saliendo después de las 1200 desde Boston a Baltimore.
25. Dame una lista de todos los vuelos desde Dallas a Boston que lleguen antes de 7:00 am.
26. ¿Cuánto cuestan los vuelos número 1, 2, 3, 4, 5?
27. ¿Cuánto cuestan los vuelos desde Atlanta a San Francisco?
28. ¿Cuánto cuesta un viaje redondo desde Boston a Dallas?
29. Necesito vuelos que lleguen antes del mediodía.
30. Lista sólo vuelos llegando antes de las 7:00 pm.
31. Lista sólo vuelos saliendo de San Francisco.
32. Muéstrame vuelos que salgan después del mediodía.
33. Muéstrame las aerolíneas que vuelan desde Dallas a Denver.
34. Dame una lista de los vuelos desde Denver a San Francisco.
35. Muestra todos los vuelos y tarifas desde Fort Worth a Denver.
36. Da todos los vuelos desde Dallas a Boston a Denver.
37. Dame una tarifa de viaje redondo desde Atlanta a Baltimore.
38. Lista todos los vuelos desde Oakland a San Francisco mostrando los precios.
39. Lista vuelos desde Atlanta a San Francisco.

40. Dame los títulos de los libros.
41. ¿Cuál es la dirección de la editorial y su ciudad?
42. ¿Cuál es el título del libro con identificador TC4203?
43. ¿A qué almacén pertenece la siguiente dirección 679 Carson st.?
44. ¿Qué empleado tiene como identificador H-B39728F?
45. ¿Qué autores viven en la ciudad de Oakland?
46. ¿Qué trabajador tiene su fecha de contratación como 13/02/1991?
47. ¿Qué libros son del tipo bussines?
48. ¿Quién es el autor del título The Busy?
49. Mostrar los libros cuyo precio es mayor a $19.99 y son de tipo bussines.
50. ¿Qué editorial se encuentra en Alemania?
51. Selecciona todos los libros del autor Smith.
52. Dame los títulos del autor Green.
53. ¿Qué puesto tiene el empleado Francisco?
54. Selecciona el descuento para el almacén 8042.
55. ¿Cuál es el número de teléfono del autor Cheryl?
56. ¿Cuál es la ciudad de la editorial New Moon Books?
57. ¿Cuál es el identificador del empleado Paolo Accort?
58. ¿Cuáles son los títulos de la editorial GGG&G?

59. ¿Cuál es la clave y el precio del libro You Can?
60. ¿Qué apellido tiene el empleado Pedro?
61. ¿En qué ciudad se encuentra el almacén Bookbeat?
62. ¿En qué fecha se realizó el contrato del empleado PTC11962M?
63. ¿Cuál es el número de empleados de la editorial Scotney Book?
64. ¿A qué ciudad pertenece el código postal 89076?
65. ¿A qué ciudad corresponde la dirección 567 Pasadena Ave?
66. Dame la fecha de contratación de Pedro.
67. ¿Cuántos números de ejemplares tiene el libro The Busy?
68. Selecciona todas las editoriales del mismo país.
69. ¿Quién es el empleado que tiene más tiempo trabajando?

Referencias

Farwell, D., & Padró, L. (2010). *FreeLing: From a multilingual open-source analyzer suite to an EBMT platform.* Universitat Politècnica de Catalunya , Dept. Llenguatges i Sistemes Informàtics, TALP Research Centre, Barcelona, España.

LLorach, E. (1999). *Nueva Gramatica de la Lengua Española* (Espasa Calpe, S. A. ed.). España.

Aguirre, M. L. (2014). *Modelo semánticamente enriquecido de bases de datos para su explotación por interfaces de lenguaje natural.* Tesis de Doctorado, Instituto Tecnológico de Tijuana., Tijuana, B.C., México.

Aho, A., Seit, R., & Ullman, J. (1998). *Compiladores Principios, Técnicas y Herramientas.* Addison Wesley Longman.

Apostol, S. (9 de 10 de 2006). *Iessapostol*. Recuperado el 1 de 10 de 2015, de http://iessapostol.juntaextremadura.net/latin/gramatica/ESQUEMAS/la_concordancia.htm

Balkan, L., Netter, K., & Arnold, D. (1994). Test suites for natural language processing. *Proceedings of Language Engineering Convention.*

Benavides, P. C., & Rodríguez, S. C. (2007). *Procesamiento del Lenguaje Natural en la recuperación de Información.* Universidad de la Salle, Colombia.

Bick, E. (2006). *A constraint grammar parser for spanish.* Institute of Language and Communication, University of Southern Denmark.

Bosque, I., & Demonte, V. (1999). *Gramática Descriptiva de la Lengua Española* (Vol. 1). (J. P. Rada, Ed.) Madrid, España: Epasa Calpe.

Cervantes, J. A. (2005). *Analizador sintáctico de oraciones en español usando el método de dependencias.* Tesis de Maestria, CENIDET, Departamento de Ingeniería en Sistemas Computacionales, Cuernavaca, Morelos, México.

Comelles, E. (2010). Constituency and dependency parsers evaluation. *Sociedad Española para el Procesamiento del Lenguaje Natural.*

Del Rosario, R. M., & Hernández, P. M. (2004). *Reconocedor de comandos en español.* Instituto de Investigaciones Eléctricas, Cuernavaca, Morelos, Mexico.

Fernández, M. P. (2000). *Introducción a la lingüística.* Barcelona: Ariel.

Ferreira, A., & Kotz, G. (2010). *ELE-Tutor Inteligente: Un analizador computacional para el tratamiento de errores gramaticales en Español como Lengua Extranjera*.* Universidad de Concepción, Concepción, Chile.

Galicia, S. (2000). *Análisis sintáctico conducido por un diccionario de patrones de manejo sintáctico para lenguaje español.* Tesis Doctoral, IPN, CII, D.F,Mexico.

Galicia, S. H., Gelbukh, A., & A, I. B. (2002). Análisis sintáctico para el español basado en el formalismo de la teoría significado-texto.

Gaya,S, G. (1991). Curso Superior de Sintaxis Española. *Bilograf S.A*, 103-153.

Gelbukh, A. (2010). *Procesamiento de lenguaje natural y sus aplicaciones* (Vol. vol. 1). Komputer Sapiens.

GramaticasNet. (27 de 07 de 2014). Recuperado el 1 de 10 de 2015, de http://www.gramaticas.net/2012/05/accidentes-gramaticales-del-verbo.html

Hill, I. (1983). *Natural language versus computer language.* Academic Press.

Hutchins, J. (2005). The history of machine translation in a nutshell.

Liddy, E. D. (2001). *Natural Language Processing for Information Retrieval & Knowledge Discovery*. School of Information Studies, New York, U.S.A.

Loáiciga, S. S. (2012). *Análisis léxico funcional de la sintaxis: propuesta para el procesamiento automático del español.* Tesis de Maestria, Universidad de Costa Rica.

Mejia, A., Mira, D., & Mercado, G. (s.f.). *Las categorías gramaticales. Iniciación al análisis morfológico elemental*. Recuperado el 8 de 10 de 2015, de Angarmegia: Ciencia, Cultura y Educación, Portal de Investigación y docencia: http://angarmegia.com.

Mellado, O. C. (2014). *Implementación de un Analizador Sintáctico del Idioma Español para una Interfaz de lenguaje natural a Base de Datos.* Tesis de Maestria, Instituto Tecnológico de Cd. Madero, Depto. de Sistemas Computacionales, Tamaulipas, México.

Mendoza , J. P., & Palma , A. A. (2010). *La morfologia en el desarrollo del proceso de enseñanzaaprendizaje del idioma inglés en los estudiantes de bachillerato del colegio nacional olmedo de la ciudad de portoviejo.* Tesis de Grado, Universidad Tecnica de Manabi, Facultad de Filosofia, Letras y Ciencias de la Educacion, Portoviejo,Manabi,Ecuador.

Meza, I., & Pineda, L. (2002). The Spanish auxiliari verb system in HPSG. *Computer Science 2276*, 200-209.

Peña, L. (2013). *Larousse Gramatica Lengua Española.* Paris: Ediciones Larousse.

RAE. (s.f.). Recuperado el Julio de 2016, de (http://dle.rae.es/?id=7wJdjdR)

RAE. (s.f.). Recuperado el Agosto de 2016, de (http://dle.rae.es/id=A9sjzXK)

RAE. (2009). *Nueva Gramatica de la Lengua Española (Manual)* (1 ed., Vol. 1). S.L.U. Espasa Libros.

RAE. (2010). *Concordancia - DPD 1.ª edición, 2.ª tirada - Real Academia Española.* Recuperado el 24 de 11 de 2016, de http://lema.rae.es/dpd/srv/search?id=XEVeLzVZaD6CG25cW5

Rich, E., & Knight, K. (1994). *Inteligencia Artificial.* Madrid: McGraw-Hill/Interamericana de España S. A.

Rojas, J. (2009). *Administrador de Diálogo para una Interfaz de Lenguaje Natural a Bases de Datos.* Tesis de Doctorado., Centro Nacional de Investigación y Desarrollo Tecnológico., Depto. de Ciencias Computacionales,, Cuernavaca, México.

Turing, A. (1950). Computing machinery and intelligence. *Mind, 59*, 433-460.

Vaan the Koot, H. (1992). Word Grammar Recognition is NP-hard. *UCL Working Papers in Linguistics, 4*, 406-416.