

PROYECTO DE GRADO

Presentado ante la ilustre UNIVERSIDAD DE LOS ANDES como requisito parcial para
obtener el Título de INGENIERO DE SISTEMAS

DESARROLLO DE UN MOTOR DE TRADUCCIÓN DEL LENGUAJE HABLADO A LENGUAJE DE SEÑAS VENEZOLANO. (CASO: INGENIERÍA DE SISTEMAS)

Por

www.bdigital.ula.ve

Br. María Fernanda Blanco Rodríguez

Tutor: Prof. Alejandro Mujica

Julio 2018



©2018 Universidad de Los Andes Mérida, Venezuela

C.C. Reconocimiento

Desarrollo de un motor de traducción del lenguaje hablado a lenguaje de señas venezolano. (Caso: Ingeniería de Sistemas)

Br. María Fernanda Blanco Rodríguez

Proyecto de Grado — Sistemas Computacionales, 68 páginas

Resumen: La humanidad ha sido testigo de numerosas injusticias contra aquellos a los que una mayoría considera diferentes o que en efecto poseen alguna diferencia, aún cuando nos consideramos una sociedad desarrollada, siguen existiendo abusos contra las minorías. Es innegable que existen grupos, asociaciones y organizaciones realizando trabajo tecnológico para superar estos problemas, sin embargo y con mucho pesar, aún existen limitaciones que la tecnología y la capacidad humana no han podido resolver. Poco a poco aparecen instituciones que buscan minimizar la segregación de grupos minoritarios, como lo es la comunidad sorda. En esta incansable búsqueda de mejorar la calidad de vida de grupos discriminados se puede encontrar una amplia cantidad de ejemplos donde la tecnología busca dar la mano al más necesitado.

Este motor de traducción forma parte de este compendio de tecnologías emergentes orientadas a la accesibilidad. En este trabajo se presenta el desarrollo de un motor de traducción de texto español venezolano al lenguaje de señas venezolano (LSV). El proyecto tiene como población objetivo a los estudiantes Sordos de la facultad de ingeniería en la escuela de sistemas de la Universidad de Los Andes (ULA), Mérida, Venezuela y la jerga utilizada en los salones de clase por sus profesores.

La necesidad principal a solventar es la escasez de intérpretes con la que cuenta la ULA. Debido a la creciente demanda y a la poca existencia de estos, los estudiantes que necesitan de su ayuda, deben recurrir a métodos alternativos para entender las clases, lo que implica muchas veces un sobre esfuerzo, que se traduce en más tiempo tratando de entender una clase y menos tiempo reforzando en casa con estudio lo aprendido. Esta situación claramente implica una desventaja, que puede ser solventada gracias a este traductor. Para el desarrollo de este producto de software se utilizó Freeling como biblioteca de procesamiento de lenguaje con la API para Python, además se

desarrollaron reglas sintácticas de la LSV actual basados en la metodología lingüística de la elicitación y se desarrolló un sistema de pruebas que utiliza reconocimiento de voz con la API de Google y crea clips de película con señas resultado de la traducción.

Palabras clave: Traductor, Lenguaje de Señas Venezolano, LSV, Freeling, NLP, Lenguaje hablado, Procesamiento de Lenguaje, Accesibilidad, Tecnología Orientada al Servicio.

www.bdigital.ula.ve

C.C. Reconocimiento

Índice

Índice de Tablas	vii
Índice de Figuras	viii
Agradecimientos	x
1 Introducción	1
1.1 Antecedentes	6
1.2 Justificación	8
1.3 Objetivos	8
1.3.1 Objetivo General	8
1.3.2 Objetivos Específicos	8
2 El procesamiento del lenguaje natural, Freeling y la LSV	10
2.1 Procesamiento de lenguaje	10
2.1.1 Un poco de historia	10
2.1.2 ¿Qué es el Procesamiento del Lenguaje Natural?	11
2.1.3 Aplicaciones del PLN	14
2.2 Freeling	15
2.3 La LSV	20
2.3.1 Sintaxis de la LSV	21
2.4 Las señas utilizadas en la Ingeniería de Sistemas	24
3 Un traductor de lenguaje natural a LSV (TANtALUm)	30
3.1 El proceso de elicitación	31

3.2	Reglas de sintaxis a partir de la elicitación	34
3.3	Diseño TAMtALUm	34
4	Implementación y pruebas	42
4.1	Creando un traductor de texto a LSV	42
4.2	Pruebas	46
4.3	Configuración del entorno de trabajo	51
4.4	Creación de un módulo de pruebas	53
5	Conclusiones y recomendaciones	60
5.1	Conclusiones	61
5.2	Recomendaciones a futuro	63
	Bibliografía	65

www.bdigital.ula.ve

Índice de Tablas

2.1	Análisis disponibles en Freeling y lenguajes	17
2.2	Vocabulario documentado de la ingeniería de sistemas	29
3.1	Reglas LSV por elicitación	35
3.2	(continuación) Reglas LSV por elicitación	36
3.3	Carácteres sustituibles en el proceso de limpieza	41
4.1	Catálogo de señas extras	48
4.2	Catálogo de señas extras (continuación)	57
4.3	Algunas pruebas unitarias sobre TAMtALUm con la lista de elicitación	58
4.4	Muestra de algunas de las pruebas unitarias realizadas sobre TAMtALUm sobre vocabulario EISULA	58
4.5	Muestra de algunas de las pruebas unitarias realizadas sobre TAMtALUm sobre las palabras con flexiones	59

Índice de Figuras

1.1	La lengua de señas dentro del lenguaje.	3
1.2	Sistema traductor de voz a LSV.	5
2.1	Los niveles de PLN	12
2.2	Clases de datos lingüísticos Freeling [1]	18
2.3	Clases de procesamiento Freeling [1]	19
2.4	Abecedario dactilológico LSV	22
2.5	Seña cálculo	25
2.6	Seña derivar	25
2.7	Seña despejar	25
2.8	Seña ecuación	26
2.9	Seña inecuación	26
2.10	Seña integrar	27
2.11	Seña sumar	27
2.12	Seña restar	27
2.13	Seña multiplicar	28
2.14	Seña dividir	28
3.1	El proceso de traducción de 3 pasos.	37
3.2	Diagrama de procesos, TANtALUm	39
4.1	Diagrama de actividades para el análisis de nombres	44
4.2	Diagrama de actividades para el análisis de pronombres y determinantes determinados	45
4.3	Diagrama de actividades para el análisis verbos	47

4.4	Secuencia de señas para: Juan comió tarde	49
4.5	Secuencia de señas para: abres el computador	49
4.6	Secuencia de señas para: el teorema demuestra que 0 es el límite de X .	50
4.7	Módulo de pruebas, TAMtALUm	54

www.bdigital.ula.ve

Capítulo 1

Introducción

*“Si nosotros no tuviésemos ni
voz ni lengua y quisiéramos,
sin embargo, designarnos las
cosas los unos a los otros,
¿no recurriríamos, como los
sordos, a los signos de las
manos, la cabeza y el cuerpo?”*

Platón, Cratilo (siglo V AC)

Entender el mundo es una tarea ardua. El mundo se muestra al hombre como un caos que debe ordenar para poder entender, interpretar y reconstruir [2]. Nuestra percepción de este depende de los cinco sentidos, los cuales son: gusto, tacto, olfato, vista y oído; pero cuando uno de estos se ve disminuido o simplemente no funciona, el individuo con esta carencia no cuenta con las herramientas habituales para entender su entorno. En casos como el que se plantea en el presente proyecto, el lenguaje y la tecnología colaboran como herramientas para leer la realidad y simbolizar de otra manera las señales que el sentido que se ve disminuido no puede captar.

Noam Chomsky en una entrevista de 1965 da una definición de lengua, diciendo que es “la correspondencia específica entre sonidos y significados”, más adelante se pregunta a Chomsky cómo dar cabida dentro de esta definición centrada en el sonido a los sistemas lingüísticos usados por los sordos y el mismo replica que habría que

reformular esta definición diciendo que “una lengua es la correspondencia específica entre señales (sonoras o manuales) y sus significados”, de aquí se puede decir que la lengua es independiente del medio, no es necesaria la existencia de sonidos para que un lenguaje se desarrolle cuando existe una correspondencia específica entre señales y significados [3].

El aprendizaje de una lengua es un proceso que depende muchas variables, como la cultura y la sociedad en la que se desarrolla en individuo: El aprendizaje de nuestra lengua materna implica una serie de variables culturales, sociales y lingüísticas que influyen en nuestra percepción del mundo, en nuestra forma de pensar y en la manera de vivir el presente, reconstruir el pasado y de imaginar el futuro [2], el lenguaje es uno de los elementos que distinguen a una sociedad.

La lengua materna de una persona sorda es la lengua de señas (LS), “La lengua de señas emergió como un aspecto medular en la interpretación de su realidad como Sordos.” [2] La LS es la primera lengua y en muchos casos la única lengua de millones de Sordos a nivel mundial, los sordos constituyen una comunidad minoritaria y única, caracterizada por la existencia de una lengua propia y una cultura singular. La LS es el elemento que confiere una identidad distintiva. En la obra “The Linguistic Structure of Sign Languages” de W. Stokoe se demuestra que la Lengua de Señas Americana (ASL por sus siglas en inglés) y por consiguiente todas las LS están organizadas lingüísticamente como cualquier lengua oral, y puede, por lo tanto analizarse de acuerdo con los mismos criterios que se emplean para analizar éstas últimas [3].

Desde esta definición se deduce entonces que las LS constituyen un subconjunto de las lenguas naturales y se pueden representar como se muestra en la figura 1.1, donde L es todo el lenguaje, es decir la estructura común a todas las lenguas humanas y tanto las lenguas orales (LO) como las LS de son subconjuntos de las lenguas naturales, que utilizando distintos canales cumplen las mismas características de esta estructura universal llamada lenguaje [3].

A diferencia de los llamados sistemas gestuales, las lenguas de señas pertenecen a la categoría de las lenguas naturales (LN), porque son ilimitadas, permanentes, con una gramática específica, agrupan principalmente a una comunidad y constituyen el principal medio de comunicación entre sus habitantes [3], por lo tanto en el presente

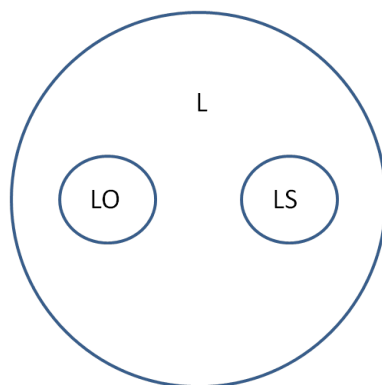


Figura 1.1: La lengua de señas dentro del lenguaje.

proyecto se plantea la traducción de una lengua natural a otra, es decir de LO a LSV.

La LS es el pilar de una comunicación insesgada en la comunidad sorda, en palabras de Chela-Flores “Somos lo que somos, no por los genes que nos formaron, sino por la visión del mundo que tenemos. Y la visión del mundo nos la da la lengua, nos la conforma la lengua y la transmitimos por la lengua” [2]. La lengua moldea, da identidad y define la interacción que se tiene con el mundo. El Sordo se comunica y habla con sus manos. La LS se define como un código que cumple con todas las funciones que las lenguas orales cumplen en las comunidades de oyentes. Las lenguas de señas son las lenguas naturales de las personas sordas. Estos sistemas se adquieren de manera natural, y además permiten a sus usuarios desarrollar el pensamiento de manera espontánea y cumplir con las funciones comunicativas propias de un conglomerado social [2].

En Venezuela no se cuenta con un número preciso de sordos, se puede tener un estimado de acuerdo a los controles llevados en ambulatorios y hospitales estatales del territorio nacional. Para el 2014 se estimaba que el 0,4% de la población contaba con una deficiencia auditiva [4]. La comunidad sorda en nuestro país se comunica principalmente a través del LSV; término acuñado por la lingüista Lourdes de Pietrosémoli en el I Seminario de Lingüística de LS realizado en 1989. Su propuesta se fundamentó en la tendencia seguida por otras naciones donde las lenguas se denominan de acuerdo al nombre del país de origen [5]. La LSV es la lengua oficial de ese 0,4% de habitantes sordos de nuestro país y en función de atender las necesidades de esta

comunidad debe despertar el interés general en la búsqueda de medios que faciliten la comunicación entre todos los individuos de una misma sociedad.

Por otra parte, para que en un determinado contexto se dé una comunicación exitosa, las partes involucradas deben seguir cierto protocolo. Un ente receptor, un emisor y un mensaje figuran como parte de los elementos principales del proceso de comunicación, dentro del protocolo previamente mencionado se establece que el emisor y el receptor deben entender el mismo lenguaje y con esto se puede asegurar que la comunicación fue exitosa. Sin embargo, afirmar que esta es la única forma de comunicación que tiene un final positivo es una mentira. Si el emisor y el receptor manejan lenguajes diferentes el mensaje que envía el emisor se puede decodificar al lenguaje del receptor y así esta comunicación también sería exitosa.

En el caso particular de este proyecto de desarrollo se plantea el comienzo de una estrategia, que busca la solución a un grave problema de comunicación existente entre un estudiante Sordo de la Universidad de los Andes, Mérida, de la facultad de Ingeniería en la escuela de Sistemas y sus profesores. Esta persona, que de ahora es adelante se denominará Product Owner (PO), es un estudiante del 5to semestre cuya primera lengua es la LS y como lengua complementaria el español escrito. Con este producto se espera crear un canal de ida de información entre los emisores y el receptor. La creación de su contraparte es una tarea que no se aborda ni se menciona en este proyecto, pero que se espera pueda llevarse a cabo en un lapso prudencial de tiempo.

Los argumentos que se han desarrollado hasta ahora llevan a definir un problema crucial en el proceso de aprendizaje del PO. Según la Declaración Universal de Derechos Humanos, las personas que nacen con una discapacidad son libres, como cualquier persona, pero se reconoce que corren mayor riesgo de que sus derechos sean vulnerados [6]. Esto sucede en un acto de discriminación involuntario en el que se viola el derecho al acceso a la información especificado en el artículo 19 de la Declaración Universal de los Derechos Humanos. Además en [Art. 81] de la constitución de la República Bolivariana de Venezuela dice: "Toda persona con discapacidad o necesidades especiales tiene derecho al ejercicio pleno y autónomo de sus capacidades y a su integración familiar y comunitaria. El Estado, con la participación solidaria de las familias y la sociedad, le garantizará el respeto a su dignidad humana, la equiparación de oportunidades,

condiciones laborales satisfactorias, y promoverá su formación, capacitación y acceso al empleo acorde con sus condiciones, de conformidad con la ley. Se les reconoce a las personas sordas o mudas el derecho a expresarse y comunicarse a través de la lengua de señas venezolana” [7]. Por lo tanto es un deber como ciudadanos velar por la igualdad de oportunidades.

En el presente proyecto se plantea un sistema Traductor de Voz a LSV compuesto por 3 sub-sistemas ó módulos, descrito en la figura 1.2, que busca ser el decodificador del lenguaje y el elemento aglutinante en la comunicación unidireccional entre oyentes y Sordos (Se escribe con mayúscula cuando nos referimos a una persona con deficiencia auditiva cuya primera lengua es la LS).

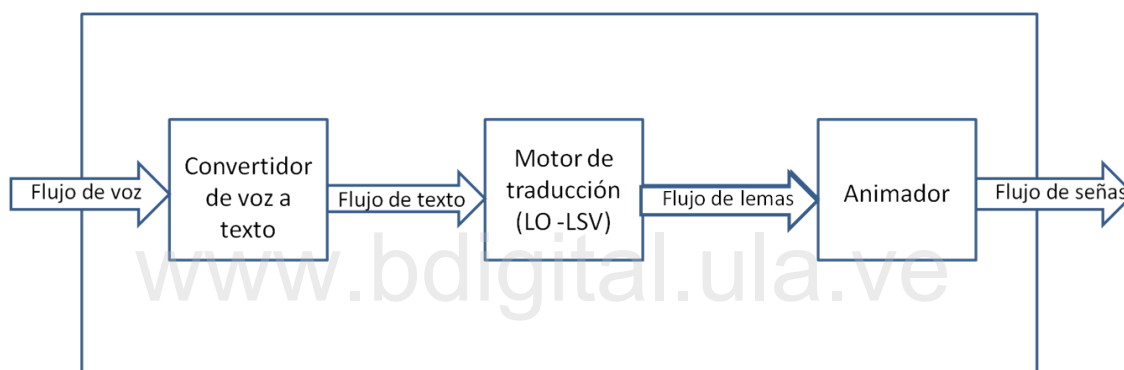


Figura 1.2: Sistema traductor de voz a LSV.

El sistema mostrado en la figura 1.2 tiene como entrada un flujo de voz de lenguaje natural, en español y con sentido coherente. Este flujo pasa al módulo convertidor de voz a texto, el cual se encarga de hacer una limpieza de sonido, convertir la entrada en un texto coherente, agregar puntuación y crear un flujo de texto correctamente escrito. El flujo de texto pasa al motor de traducción el cual traduce cada sentencia a LSV y tiene como salida un flujo de lemas que resultan de la traducción del texto de entrada, la salida de este módulo entra a un sub-sistema llamado animador, el cual se encarga de buscar en su base de datos interna la seña única correspondiente a cada uno de los elementos del flujo de lemas, la salida final del sistema es, por lo tanto las señas en LSV correspondientes a la entrada del flujo de voz.

El presente proyecto muestra el desarrollo del módulo de traducción llamado TANtALUm (acrónimo de TrAductor NATural Lsv Ula). Los fundamentos teóricos

necesarios para la comprensión de los conceptos de procesamiento de lenguaje natural (PLN), el funcionamiento de la biblioteca utilizada, todos los aspectos lingüísticos correspondientes a la sintaxis, gramática y aspectos no manuales de la LSV se mostrarán en el Capítulo 2. El planteamiento de una sintaxis para la LSV y el diseño del traductor se explican en el Capítulo 3, en el Capítulo 4 se presenta la implementación y las pruebas realizadas, los resultados obtenidos y la implementación de un módulo de pruebas, finalmente en el capítulo 5 están las conclusiones y algunas sugerencias útiles para futuros proyectos.

1.1 Antecedentes

En 1959, José Arquero Urbano (un sordo español llegado a Venezuela en los años cincuenta, recordado como el “creador de la LSV”) junto a Gustavo Alvarez Díaz y Eduardo López Ferrero, fundan la Asociación de Sordomudos de Caracas, institución que trató de velar por los derechos asistenciales y culturales de los sordos caraqueños y gracias al cual surgió un código señado inteligible entre ellos [8]. Sin embargo el interés en explorar este código surge a mediados de los ochenta y el término se acuña, como se comentó previamente [3].

En 1985 el Ministerio de Educacion venezolano encomendó el estudio lingüístico de la LSV al Postgrado de Lingüística de la Universidad de Los Andes, así que, para esta época la mayoría de los trabajos publicados sobre el tema se habían realizado en esta institución [8].

Del trabajo [9], se lee que los avances con respecto a la lingüística de la LSV han sido plasmados en diferentes informes técnicos a los que hace referencia el autor. Aquí se introducen conceptos como la nominalización y algunas reglas de sintaxis que como veremos mas adelante debido a la evolución de la LSV quedan un poco obsoletas, sin embargo su investigación aclara muchos aspectos sobre el uso de los nombres en el lenguaje.

De igual forma el trabajo presentado por [8] es una muy completa guía sobre la gramática y todos los aspectos lingüísticos de la LSV, aunque por la constante evolución del lenguaje muchas de las definiciones y reglas dadas en esta publicación deben ser

revisadas en búsqueda de su continuidad en el tiempo.

Por otra parte, a nivel mundial varias empresas se perfilan como desarrolladoras de software de accesibilidad. En la categoría de sistemas web, se pueden encontrar múltiples hitos funcionales de tecnologías al alcance del Sordo, como por ejemplo: aplicaciones del nivel de Prodeaf [10] el cual es un eficaz traductor de frases en portugués al lenguaje de señas brasileño gratuito.

Destacan también aplicaciones como [11] un traductor de lengua de señas colombiano, cuyo funcionamiento es muy similar al que se pretende con el proyecto macro al que pertenece este motor de traducción.

El avance tecnológico con respecto a la lengua de señas resulta abarcado en cada país por al menos un grupo de investigación, se deben mencionar dos proyectos en especial, el sistema de traducción directa de español a lenguaje de señas mexicano (LSM) con reglas marcadas, del cual se extrae la importancia de acotar un lenguaje en estos tipo de proyectos con tiempos limitados y del trabajo léxico que se debe realizar para trabajar con LS por la poca estandarización de las mismas [12] y el segundo proyecto es un traductor de lenguaje texto a lenguaje de señas costarricense (LESCO), el cual utiliza la biblioteca Freeling para las tareas de tokenización y filtro part-of-speech (PoS) [13].

Venezuela se perfila como uno de los países modelo en torno a proyectos de mejoramiento de calidad de vida a personas con diversidad funcional. Estos proyectos se enfocan en la creación de instituciones que sirven de apoyo a grupos minoritarios así como también proyectos de investigativos llenos de datos sobre el estudio de las señas y su evolución entre los hablantes venezolanos.

En este ámbito los casos resaltantes son [5] con su: “Software educativo para el uso del Alfabeto Manual como herramienta de enseñanza del español como segunda lengua en niños sordos del subsistema de educación inicial”, en éste afina la necesidad de crear software para el apoyo de la comunidad sorda, en especial de los niños que se adentran al universo de la lengua escrita, equiparando sus oportunidades con las de niños oyentes, cuya oralidad es nata. También se puede mencionar el trabajo de [2] en el estudio de “La lengua de señas en la vida de los sordos o el derecho de apalabrar su realidad” en el que resalta la extrema necesidad de respetar la LS como una forma de expresión validada como lengua materna y la como única forma de dar un contexto

preciso a la realidad del sordo. Ambos proyectos fueron desarrollados en el territorio nacional por venezolanos.

1.2 Justificación

Tomar el lugar de otro no es una tarea trivial, al ubicarse en la vida del Sordo puede entenderse porque la LS juega un papel tan decisivo en la visión del mundo que construye. Es darle nombre a las cosas desde lo viso-espacial. Es cómo percibe el mundo a través de la mirada o cómo organizan la experiencia de lo cotidiano por medio de lo visual [2]. Percibir el mundo en imágenes no es tarea fácil si se toma en cuenta que la oralidad siempre ha sido la única forma de transmitir la realidad. [2] en su trabajo cita a un individuo sordo anónimo que acota: “Si sólo me hablan lengua oral será muy difícil para el Sordo entender, estructurar los conocimientos, desarrollar el pensamiento así que necesito la lengua de señas para entender las cosas, el mundo”.

Teniendo en cuenta esto, se hace evidente la necesidad de crear un canal de comunicación entre oyentes y sordos. Así, la tecnología pone de su parte y ofrece todo un compendio de opciones para desarrollar herramientas que mejoren la calidad de vida, sólo queda dar rienda suelta a la creatividad y poner en práctica los conocimientos adquiridos.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un motor de traducción del lenguaje hablado al lenguaje de señas venezolano y aplicar un caso de prueba en un dispositivo móvil Android.

1.3.2 Objetivos Específicos

- Estudiar el lenguaje especializado usado en la carrera de Ingeniería en Sistemas y asignarle a cada estructura gramatical una seña en LSV de conversión de voz a texto.

- Convertir texto en unidades mínimas traducibles a LSV.
- Crear una estructura de clasificación de lenguaje implantable en dispositivos móviles y computadores personales.
- Crear una aplicación móvil (caso de prueba), atractiva, de fácil uso y que optimice los recursos del equipo anfitrión cuya entrada sea la voz del interlocutor y cuya salida sean las señas que traducen lo que este quiere comunicar.
- Implementar la aplicación móvil en un dispositivo Android.

www.bdigital.ula.ve

Capítulo 2

El procesamiento del lenguaje natural, Freeling y la LSV

2.1 Procesamiento de lenguaje

2.1.1 Un poco de historia

Tras haber presidido en Panel de Matemáticas Aplicadas de la Oficina de Investigación y Desarrollo Científico de los Estados Unidos durante la Segunda Guerra Mundial y ser testigo de la potencia y velocidad de las computadoras en aquel momento Warren Weaver, escribió un memorando llamado “Translation”. En este documento, relata la historia de un matemático que tras haber desarrollado una técnica de descifrado le pide a un colega un mensaje codificado para ponerla a prueba. El colega le entrega un mensaje que el matemático logra descifrar correctamente y vuelve a su colega con un mensaje en escrito en Turco. En este mismo memorando, el autor propone que un mensaje en determinado idioma podría ser considerado como si estuviera escrito en código, es decir, el idioma alemán es inglés encriptado y que la traducción es un proceso de descifrado. Las conclusiones extraídas de este memorando se consideraron muy influyentes y estimulantes para el comienzo de la investigación de la traducción automática [14].

Los primeros estudios en el campo de la traducción automática se realizaron en la traducción del alemán al inglés dado que aún habían documentos de la Segunda

Guerra Mundial pendientes por traducir, pero con la venida de la Guerra Fría se pasó a la traducción del francés y ruso al inglés [14].

Los sistemas pioneros en el área fueron realizados por matemáticos bilingües que estaban convencidos de poder llevar su conocimiento de la lengua a la computadora, pero con maquinaria de computación un tanto primitiva y sin técnicas de PLN estos primeros sistemas no tuvieron éxito. Entonces, se busca apoyo en los lingüistas. Una gran cantidad de jóvenes investigadores en el campo de la lingüística se unieron a estos matemáticos en el desarrollo de sistemas de traducción, pero para el pesar de todos en ese momento aún no existían técnicas lingüísticas apropiadas para afrontar el problema de la traducción automática.

Todo cambia cuando en 1957 Noam Chomsky publica su libro: Syntactic Structures [15] el cual se vuelve una guía obligatoria en cuanto al procesamiento de lenguaje de refiere.

Los altos costos que implicaba la traducción automática con las computadoras de la época y tomando en cuenta que era más económica la traducción manual, los inversionistas pararon totalmente el flujo de dinero hacia esta causa, lo que trajo consigo una edad oscura para el PNL de al menos 14 años [14].

Luego en los 80s con la aparición de los algoritmos de aprendizaje automático la investigación con respecto a la traducción automática entro un en auge donde predominaron los algoritmos basados en probabilidad [14], muchos de estos son hoy en día muy utilizados.

Actualmente el PLN es una de las áreas de mayor estudio para computistas, lingüistas y demás interesados en el tema a nivel mundial.

2.1.2 ¿Qué es el Procesamiento del Lenguaje Natural?

En palabras de [16] “Por Procesamiento del Lenguaje Natural, estamos hablando de técnicas computacionales que procesan lenguaje humano hablado y escrito. Esto es una definición inclusiva que abarca todo desde aplicaciones mundanas como conteo de palabras, hasta aplicaciones de punta como sistemas de respuesta de preguntas en la Web, o traducción automática en tiempo real. Lo que distingue a estas aplicaciones de procesamiento del lenguaje de otros sistemas de procesamiento de datos es el uso del

conocimiento del lenguaje.”

El PLN a su vez se encuentra dividido en disciplinas que trabajan conjuntamente. Un módulo de PLN debe ser capaz de generar una señal de audio y analizarla con el uso de la fonética y la fonología, procesar variaciones de palabras con el uso de la morfología (el estudio de la estructura interna de las palabras) y precisar si el orden de aparición es correcto gracias a la sintaxis. Aunque todo este conocimiento se ve opacado si el sistema no comprende la naturaleza de lo que se desea procesar, la capacidad de entender el significado de las palabras y su composición es la semántica. Además puede ser deseable que un sistema que haga PLN pueda generar un análisis que tome pueda producir y utilizar la amabilidad gracias a la pragmática (el estudio de algunos componentes de la relación entre el lenguaje y el contexto de uso). El estudio del lenguaje natural puede dividirse entonces en cuatro niveles de análisis: morfológico, sintáctico, semántico y pragmático, pudiéndose excluir el análisis fonético y fonológico [14].

Los cuatro niveles nombrados previamente siguen una secuencia en la cual la salida del nivel anterior es la entrada del siguiente, como se muestra en la figura 2.1.

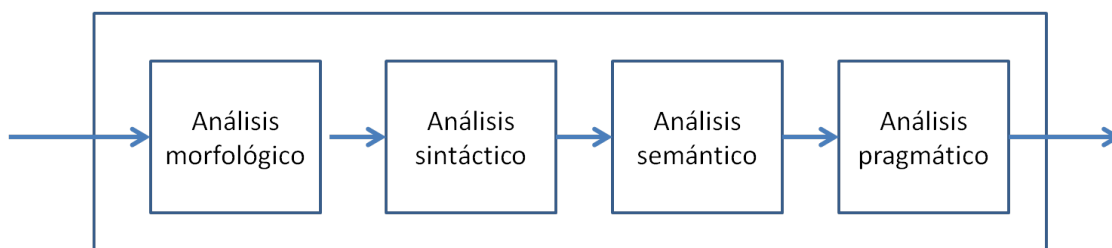


Figura 2.1: Los niveles de PLN

Las siguientes definiciones correspondientes a los niveles de análisis son tomadas de [16] y al resumen provisto en [14]:

- **Análisis morfológico** Consiste en detectar la relación entre las unidades mínimas que forman una palabra (morfemas), reconocerla y construir una representación estructurada. Una palabra es la combinación de una raíz y un conjunto de morfemas, los cuales son aplicados mediante flexión, derivación y composición.

La flexión es un mecanismo de producción de palabras dentro de una misma clase. Por ejemplo, al flexionar la raíz “maestr-” con el morfema flexivo de género masculino, se obtiene la palabra “maestro”, lo mismo ocurre por ejemplo con la palabra “maestras” la cual esta tiene los morfemas flexivos de género femenino y el de número plural.

La derivación es la combinación de una raíz con un afijo para generar una palabra de otra clase, o con otro significado. Ejemplo: estable (adjetivo, sufijo “-able”) → estabilizar (verbo, sufijo “-izar”) → estabilización (sustantivo, sufijo “-ación”) desestabilización (sustantivo, prefijo “des-”, sufijo “-ación”).

Y finalmente la composición es la unión de dos raíces. Por ejemplo: medio + día = mediodía.

- **Análisis sintáctico** El análisis sintáctico (parsing) tiene como cometido el etiquetado de cada componente sintáctico (sintagma) que aparece en una oración, construyendo en el proceso un árbol de análisis sintáctico, el cual tiene como nodo raíz el símbolo de oración, y tiene como hojas a las palabras. Un analizador sintáctico (parser), para la construcción de dicho árbol, toma como entrada un texto y una determinada gramática, y devuelve el árbol, o falla en caso de no poder hacerlo con las reglas de producción gramaticales especificadas.

Los algoritmos de análisis sintáctico pueden ser agrupados de acuerdo a su estrategia de análisis: descendente (top-down) y ascendente (bottomup).

En el tipo descendente, se comienza a partir de la raíz del árbol (el símbolo de oración) y se trabaja hasta llegar a las hojas (las palabras).

El tipo ascendente comienza desde las hojas (las palabras de la entrada) y se sube hasta llegar al símbolo de la oración.

- **Análisis semántico y pragmático** El análisis semántico es la tarea de procesar el lenguaje natural y comprender su significado. Esto permite saber si una oración es una correcta representación de la realidad, responder preguntas o saber que se requeriría para responderlas en caso de faltar información.

La pragmática provee contexto, conocimiento del mundo, convenciones de uso del lenguaje, creencias e intenciones de los participantes en la comunicación.

2.1.3 Aplicaciones del PLN

Actualmente, buena parte del saber humano se encuentra en forma digital en distintos tipos de colecciones de datos. Los volúmenes de información son inmensos y las computadoras archivan toda esta data, pero sin los PLN es difícil aprovecharla, el PLN simplifica muchas tareas computacionales y del uso de datos. Las siguientes definiciones y descripciones son tomadas de [17].

- **Recuperación y extracción de información** La recuperación de información (RI) es el proceso de encontrar en un repositorio grande de datos, material (usualmente documentos) de naturaleza no estructurada (usualmente texto) o semiestructurada (páginas Web, por ejemplo), que satisfaga una necesidad de información.
- **Minería de datos** La minería de datos proporciona herramientas poderosas para descubrir patrones ocultos y relaciones en datos estructurados. Este proceso asume que los datos ya se encuentran almacenados en un formato estructurado. Por esta razón su pre-procesamiento consiste en la limpieza y normalización de los datos y la generación de numerosos enlaces entre las tablas de las bases de datos.
- **Sistemas de búsquedas de respuestas** Son sistemas diseñados para tomar una pregunta en lenguaje natural y proporcionar una respuesta. De esta manera los usuarios no tendrían que navegar y leer una o varias páginas de resultados de búsqueda. Estos sistemas se construyen sobre motores de búsqueda y requieren contenido como fuente para descubrir las respuestas.
- **Generación de resúmenes automáticos** El problema de la generación de resúmenes automáticos a dos diferentes niveles: a nivel de documento y a nivel de grupos de documentos. Los resúmenes pueden ser con enfoque extractivo o abstractivo. Los métodos extractivos se basan en los mismos principios

usados en la identificación de términos, consisten en una colección de términos, frases o párrafos significativos que definen el significado del texto original. Los abstractivos depende de técnicas de parafraseo para producir las síntesis, las técnicas aún están siendo desarrolladas.

- **Análisis de sentimientos** El análisis de sentimientos en textos es la identificación y extracción de información subjetiva. También llamado “minería de opiniones”, ese proceso generalmente involucra el uso de herramientas de PLN y software de análisis de textos para automatizar el proceso.
- **Traducción automática** La traducción automática tiene objetivos claros: tomar el texto escrito en un lenguaje y traducirlo a otro, manteniendo el mismo significado.

En general el proceso de traducción automática sigue tres pasos: primero, el texto en el lenguaje original se transforma a una representación intermedia, luego, de acuerdo a la morfología del lenguaje destino, se realizan modificaciones a esta representación intermedia y por último ésta se transforma al lenguaje destino.

A pesar de los avances en esta tecnología, todavía hay muchos retos en la traducción. La investigación en esta área tiene muchas direcciones no exploradas: métodos de aprendizaje avanzado en modelos estadísticos de traducción, modelos sintácticos y sus representaciones, integración de traducción del habla con otras aplicaciones como reconocimiento de voz y traducción automática.

2.2 Freeling

FreeLing es una biblioteca de código abierto para el procesamiento multilingüe, que proporciona una amplia gama de funcionalidades de análisis para varios idiomas. El proyecto FreeLing se inició desde el centro TALP (siglas para The Language and Speech Technologies and Applications Center) de la Universitat Politècnica de Catalunya (UPC) para avanzar hacia la disponibilidad general de recursos y herramientas básicos de PLN. Esta disponibilidad debería posibilitar avances más rápidos en proyectos de investigación y costes más reducidos en el desarrollo de aplicaciones industriales de

PLN. El proyecto se estructura como una biblioteca que puede ser llamada desde cualquier aplicación de usuario que requiera servicios de análisis del lenguaje. El software se distribuye como código abierto bajo una licencia GNU General Public License y bajo licencia dual a empresas que deseen incluirlo en sus productos comerciales [18].

FreeLing está concebido como una biblioteca sobre la cual se puedan desarrollar potentes aplicaciones de PLN, y orientado a facilitar la integración con las aplicaciones de niveles superiores de los servicios lingüísticos que ofrece [1], actualmente se encuentra en su versión 4.1, siendo lanzada por primera vez en septiembre de 2010, es decir, es un proyecto con 8 años de trabajo conjunto, con fuertes bases y que para su versión 2.1 contaba ya con más de 16.000 descargas (No se encontraron datos sobre el número de descargas para la version 4.1), deben ser mencionados los trabajos [18], [1], [19], [20] y [21] en los que se tratan las funcionalidades de la biblioteca y su evolución.

La capacidad de análisis disponible para cada lenguaje actualmente en FreeLing se detalla en la tabla 2.1, donde las columnas se refieren a los idiomas y las filas a las capacidades de análisis disponibles. Entre las funciones más importantes y más utilizadas por este proyecto están la tokenización, la división de oraciones, la detección de la morfología de las palabras, el etiquetado de discurso (PoS tagging) y el Full/dependency parsing. Para el español (es) se encuentran desarrolladas todas las capacidades, la cual fue la principal razón para elegir la biblioteca en este proyecto de desarrollo.

Las clases básicas de la biblioteca mostradas en el diagrama de la figura 2.2 tienen la finalidad de contener los datos lingüísticos (palabras, etiquetas morfológicas, frases, árboles sintácticos, párrafos...) resultado de los análisis realizados. Cualquier aplicación cliente debe usar estas clases para poder proporcionar a los módulos de análisis los datos en el formato oportuno, y para poder recuperar el resultado de los analizadores [1].

Las clases de datos lingüísticos tomados de [1] son las siguientes:

- **analysis** Una tupla <lema, etiqueta, probabilidad, lista de sentidos >
- **word** Forma una palabra con una lista de posibles objetos analysis.

	as	ca	cy	de	en	es	fr	gl	hr	it	nb	pt	ru	sl
Tokenization	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sentences splitting	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Number detection		X		X	X	X	X	X		X		X	X	
Date detection		X		X	X	X	X	X				X	X	
Morfological detection	X	X	X	X	X	X	X	X		X	X	X	X	X
Affix rules	X	X	X	X	X	X	X	X		X	X	X		
Multiword detection	X	X	X		X	X	X	X		X		X		
Basic named entity detection	X	X	X		X	X	X	X		X		X	X	X
B-I-O named entity detection		X			X	X		X				X		
Named Entity Classification		X			X	X						X		
Quantity detection		X			X	X		X				X	X	
PoS tagging	X	X	X	X	X	X	X	X		X	X	X	X	X
Phonetic encoding					X	X								
WN sense annotation		X			X	X	X	X	X					X
UKB sense disambiguation		X			X	X	X	X	X					X
Shallow parsing	X	X			X	X		X				X		
Full/dependency parsing	X	X			X	X		X	X					X
Semantic Role Labelling		X		X	X	X								
Coreference resolution					X	X								

Tabla 2.1: Análisis disponibles en Freeling y lenguajes

- **sentence** Una lista de objetos **word** marcada como una frase completa. Puede contener también un árbol de constituyentes o de dependencias.
- **paragraph** Una lista de objetos **sentence** marcada como un párrafo independiente.
- **document** Una lista de objetos **paragraph** que forman un documento completo. Puede contener también información sobre la conferencia entre las menciones a entidades del documento.

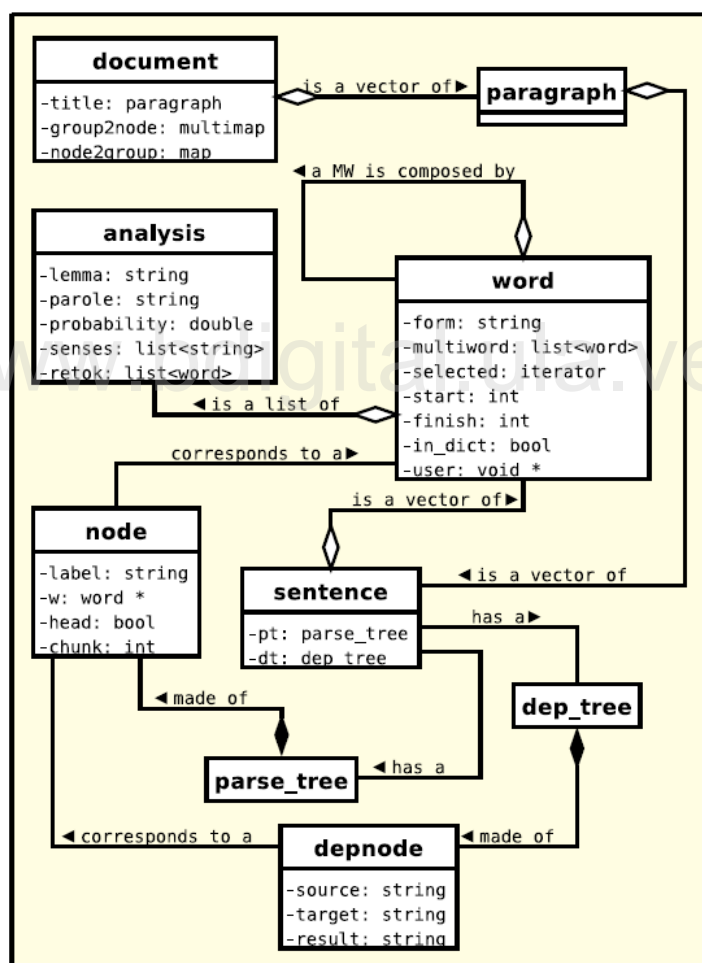


Figura 2.2: Clases de datos lingüísticos Freeling [1]

Además de las clases de datos lingüísticos la biblioteca provee también clases para transformarlos, generalmente enriqueciéndolos con información adicional. Las clases

de procesamiento se muestran en el diagrama de la figura 2.3 y se explican las clases más resaltantes para este proyecto a continuación:

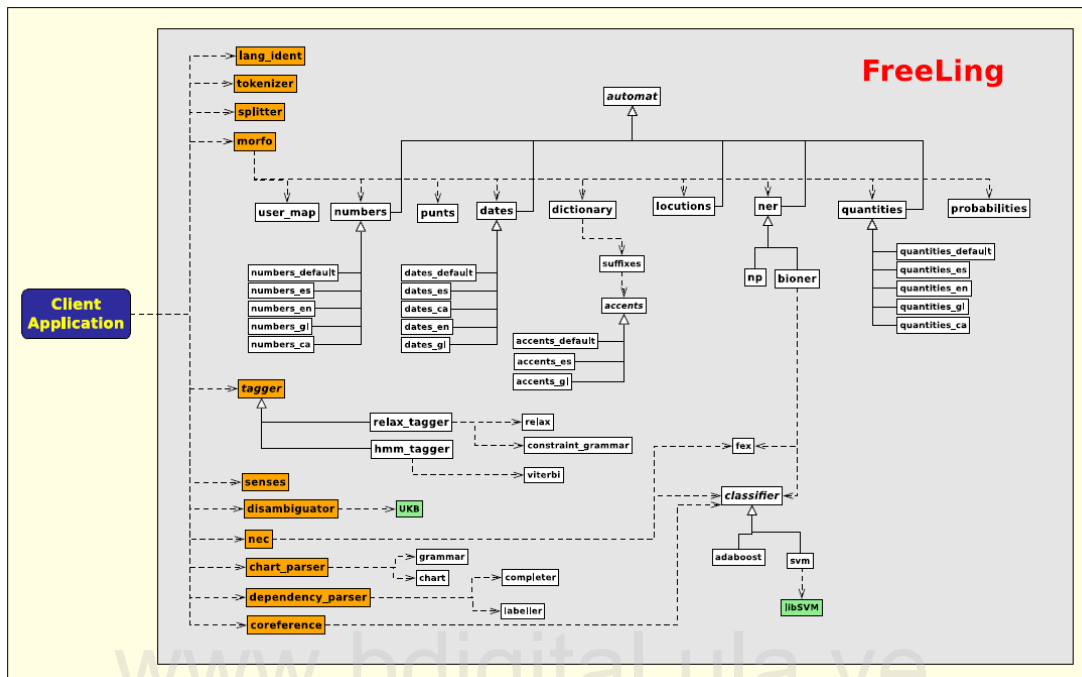


Figura 2.3: Clases de procesamiento FreeLing [1]

- **tokenizer** Recibe texto plano y devuelve una lista de objetos word.
- **splitter** Recibe una lista de objetos word y devuelve una lista de objetos sentence.
- **morfo** Recibe una lista de objetos sentence y analiza morfológicamente cada word de cada sentence de la lista. Esta clase es un metaanalizador que simplemente aplica una cascada de analizadores especializados (detección de números, fechas, locuciones y multipalabras, búsqueda en formario, etc.)
- **tagger** Recibe una lista de objetos sentence y desambigua la categoría morfosintáctica de cada palabra en las frases de la lista. Si el análisis seleccionado incorpora información de retokenización (p.e. del → de+el, dárselo → dar+se+lo) la palabra puede separarse en varias. FreeLing ofrece dos taggers con una precisión del estado del arte 97%-98%): Uno basado en modelos ocultos

de markov (HMM por sus siglas en inglés) y otro basado en relaxation labelling. En este proyecto se utilizó el modelo basado en HMM.

El tagger utiliza las etiquetas EAGLES, que trata de cubrir todas las características morfológicas de las palabras, para la mayoría de los lenguajes europeos. EAGLES trata de cubrir todas las características morfológicas de las palabras, para la mayoría de los lenguajes Europeos.

El etiquetado PoS de EAGLES consiste en etiquetas de longitud variable donde cada carácter corresponde a una característica morfológica. El primer carácter de la etiqueta es siempre una categoría de PoS. La categoría determina la longitud de la etiqueta y la interpretación que debe tener cada palabra, por lo tanto si se desea saber cuando una palabra es un verbo, basta con examinar el primer carácter de su etiqueta y verificar que pertenece a la categoría de verbos, las letras restantes del tag corresponden a las diferentes características de los verbos, como el tipo, el humor, el tiempo, la persona, el número y el género. Freeling tiene una excelente documentación sobre cada una de las etiquetas en [22].

2.3 La LSV

Como se mencionó antes, la LS es una lengua natural, que cumple con ciertas características, pero además un sistema lingüístico de señas dado sea considerado LS. Dichas condiciones son las siguientes:

- Las LS constituyen el principal medio de comunicación entre los hablantes (Se usa el término apegado al significado lingüístico como el “usuario de una lengua X”, sea una LO una LS) sordos de una comunidad.
- Las LS tienen una vía natural de transmisión de hablantes sordos a niños sordos, sean estos hablantes adultos: padres o parientes; sean sus pares: compañeros de escuela.
- recíprocamente, un hablante de una LS X ha adquirido esa lengua a través de un usuario sordo y la usa en su interrelación diaria con los individuos de su comunidad como medio básico de comunicación [23].

Estas condiciones se cumplen para la LSV, de manera que se puede inferir que como toda lengua posee una estructura y una organización idiomática propia [8] y no que son traducciones o calgos gestuales de las lenguas orales [23], como se solía pensar en los 80s al comienzo de su estudio lingüístico y como aún ahora todavía se cree que las LS “parasitan” en las lenguas orales, es decir que los gestos son heredados de las LO. Las LS son lenguas con significación propia, no son mímicas puesto que estas últimas no se usan con propósitos comunicativos y además no poseen una estructura propia, cada mimo tiene la libertad de realizar las mímicas de la forma que mejor le parezca, es una actividad sujeta a la subjetividad.

Es importante aclarar también la relación que existe entre el deletro manual y las LS. El abecedario dactilológico es un conjunto de configuraciones manuales (señas) que representan las letras del abecedario escrito. Este abecedario, mostrado en la figura 2.4, se usa cuando los Sordos quieren por alguna razón “traducir” la forma de las palabras de las lenguas orales [23] o bien cuando no existe una señal determinada para una palabra.

www.bdigital.ula.ve

2.3.1 Sintaxis de la LSV

La unidad esencial de la sintaxis es la oración o la proposición. Las oraciones están constituidas a partir de las palabras o morfemas de la lengua [23], en el caso de la LSV, las señas. Es decir una oración es LSV es una secuencia de señas acompañadas de una serie de gestos no manuales que permiten la desambiguación de términos y en la mayoría de los casos la comprensión absoluta de una frase.

Continuando con la sintaxis, entre los morfemas que contruyen una oración, son básicos los nombres y los verbos. En [23] se cita a Givón, él propone que los conceptos de nombre y verbo se encuentran en los extremos de una escala lineal de “estabilidad temporal”, sugiere que la estabilidad en el tiempo de una oración sigue una jerarquía nombre → adjetivo → verbo, donde los nombres son más estables en el tiempo y los verbos sufren cambios más o menos rápidos en esta escala. La aclaratoria previa es de fundamental importancia con para el presente trabajo, porque determina que el análisis de sentido en la oración al igual que en el español hablado en la LSV depende en gran parte de saber que es lo que se está diciendo sobre quien, es decir cual es el

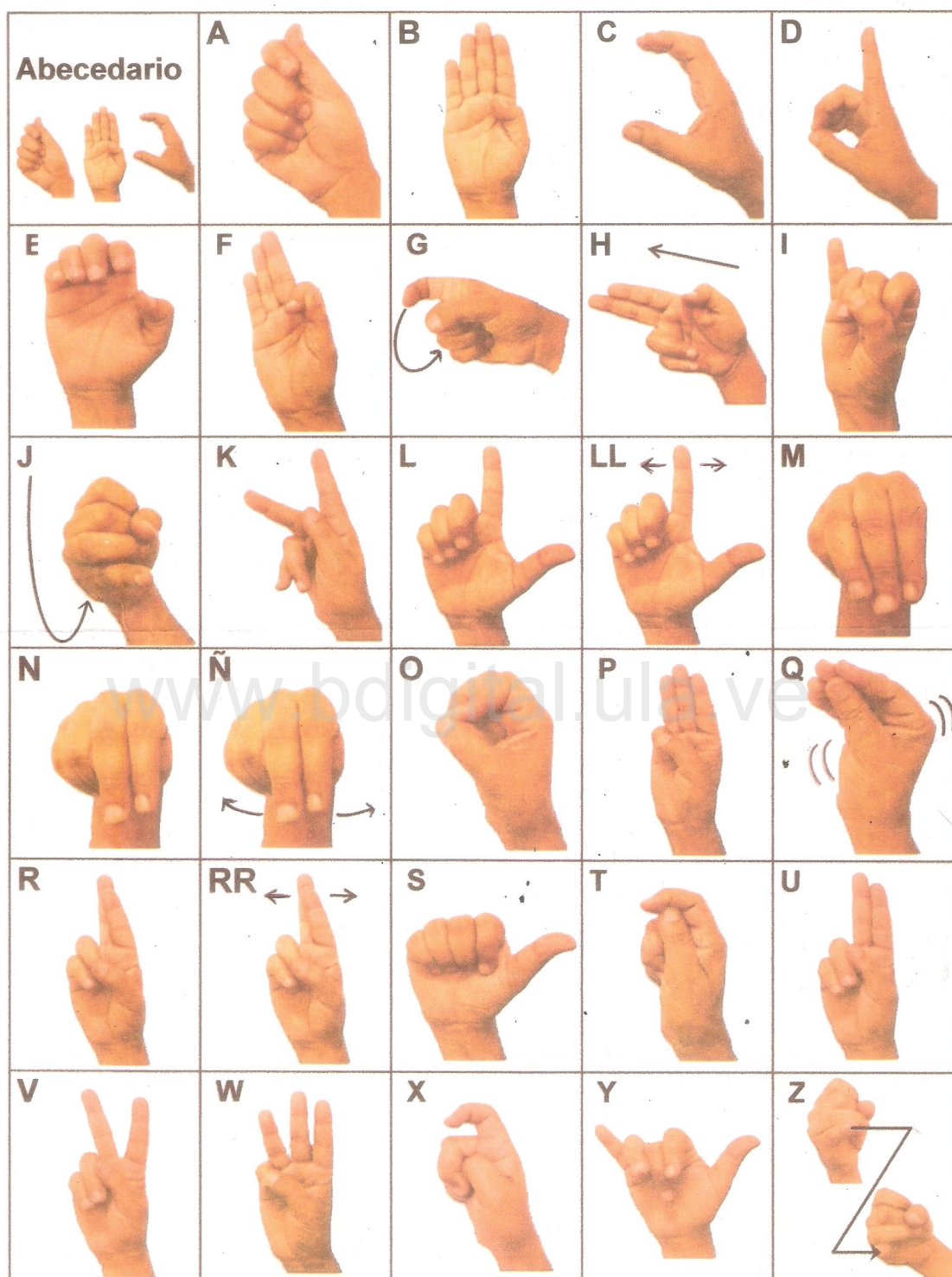


Figura 2.4: Abecedario dactilológico LSV

verbo y cual el sujeto.

Por otra parte en el trabajo investigativo [8] sobre los verbos en la LSV, cuando se desea hablar de la sintaxis, las oraciones se clasifican en: declarativas, interrogativas y de negación.

Las oraciones declarativas a su vez se dividen en oraciones con locativos, de posesión y de orden sustantivo – atributo. Resumiendo, en las oraciones con locativos se sugiere que primero se exprese un punto de referencia, es decir el lugar donde se localiza de lo que se está hablando y luego el objeto o sujeto del que se habla, por ejemplo: “La mariposa está dentro de la flor” la traducción sugerida por el autor es “FLOR – MARIPOSA – DENTRO”. Con respecto a las oraciones de posesión el autor citando a Pietrosevoli sugiere que el orden sea primero lo poseído y luego el poseedor, por lo tanto la frase: “La casa de mi amigo” se traduciría como “MI – AMIGO – CASA”. Finalmente con respecto a las oraciones de orden sustantivo – atributo se sugiere el orden participante + atributo, quedando entonces la frase: “Todos son venezolanos” como “TODOS – VENEZUELA”.

Las oraciones interrogativas, siguen la misma sintaxis propuesta para las oraciones declarativas pero en ellas se destaca que es muy importante el uso de los rasgos no manuales, es decir, la interrogación se realiza con gestos faciales.

Y para culminar las oraciones de negación tienen 3 posibilidades, negación externa, negación implícita y negación simultánea. La negación explícita se realiza utilizando los llamados adverbios de negación de forma explícita en la oración, por ejemplo: “NUNCA”, “NO”, “JAMAS”. Las oraciones negativas implícitas se refieren a oraciones en las que la negación se encuentra dentro de la seña y para finalizar las oraciones en las que se niega simultáneamente, es decir las oraciones que además de seña se acompañan de gestos no manuales de negación, como por ejemplo un movimiento giratorio de cabeza.

De las definiciones pasadas y gracias a la participación del PO de este proyecto se lograron establecer nuevas reglas con respecto a estos apartados, la LSV evolucionó y se ha ido adaptando para volverse menos estricta sintácticamente y un poco más comprensiva con el usuario y se llegaron a ciertos acuerdos de normalización utilizados en el LSV actual, la evolución del lenguaje es monitoreada gracias a reuniones periódicas

realizadas entre hablantes de la LSV en las que se llegan a reglas e incluso se cambian las señas existentes.

Dentro de los acuerdos de normalización mencionados se encuentran por ejemplo que para las oraciones de locación una mejor traducción para “La mariposa está dentro de la flor”

LSV : “MARIPOSA - ESTAR - EN - FLOR”

Es decir es se sigue un esquema de orden Sujeto (S) - Verbo (V) - O (Objeto). El mismo esquema se sigue para las oraciones de posesión, la oración: “La casa de mi amigo” se traduce:

LSV : “CASA - DE - MI - AMIGO”

Es importante resaltar que los artículos no se utilizan en la LSV y que para las demás oraciones la sintaxis se cumple según como dice el autor de [8].

Aún así es evidente que la definición de la sintaxis se queda corta para todo el lenguaje que se debe traducir, en las LO hay más que solo oraciones de locación, posesión, interrogación y negación. El método utilizado para resolver problemas como por ejemplo la conjugación y las reglas de gramáticas se tratan en el capítulo siguiente porque forman parte del trabajo realizado.

2.4 Las señas utilizadas en la Ingeniería de Sistemas

Debida a la libertad que tiene un Sordo de crear señas de acuerdo a la necesidad que tenga, el PO ha podido crear señas nuevas para cada materia aprendida. Para el momento de desarrollo del presente documento, el PO se encuentra cursando el 5to semestre de Ingeniería de Sistemas en la Universidad de Los Andes, por lo tanto el vocabulario especializado documentado en este proyecto se encuentra incompleto para todas las materias que se cursan en toda la carrera.

Por ejemplo, para las materias de cálculo, el PO creó señas como la de la figura 2.5 que representa a la palabra cálculo, los dedos van cruzados frente al rostro del señante y

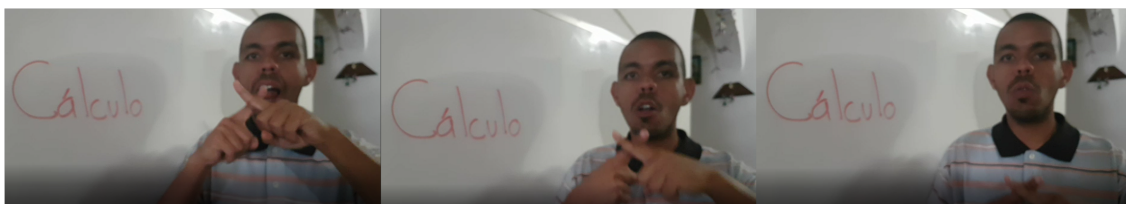


Figura 2.5: Señal cálculo

luego progresivamente las manos bajan con un corto movimiento rotatorio de muñecas hacia el cuerpo.

En la señal de la figura 2.6 el hablante posiciona su mano principal en configuración de la letra “d” del alfabeto dactilológico 2.4 y realiza un movimiento horizontal que va de un lado de su torso al otro.



Figura 2.6: Señal derivar

La señal de la palabra “despejar” puede verse en la figura 2.7, en esta configuración se utilizan las dos manos, una se mantiene estática con la palma hacia arriba y la otra realiza un movimiento de empuje con el dedo medio ligeramente acompañado de los dedos anular y meñique.

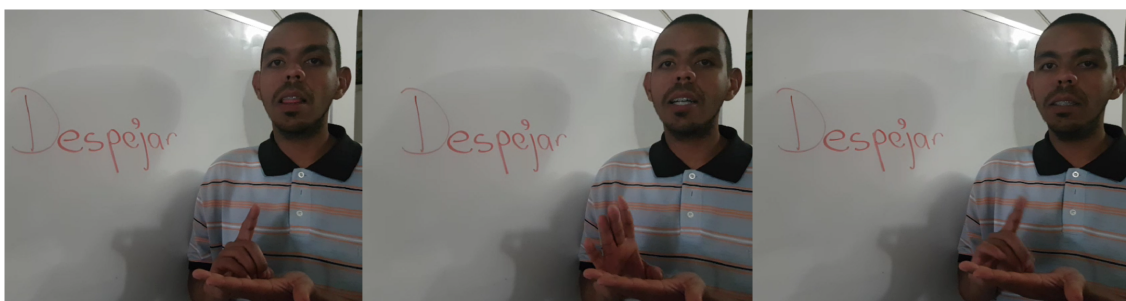


Figura 2.7: Señal despejar

Siguiendo terminos matemáticos la figura 2.8, muestra la señal para la palabra ecuación la cual inicia con la mano dominante en posición de letra “e” como se ve en 2.4, y describe un círculo en frente del torso del señante.

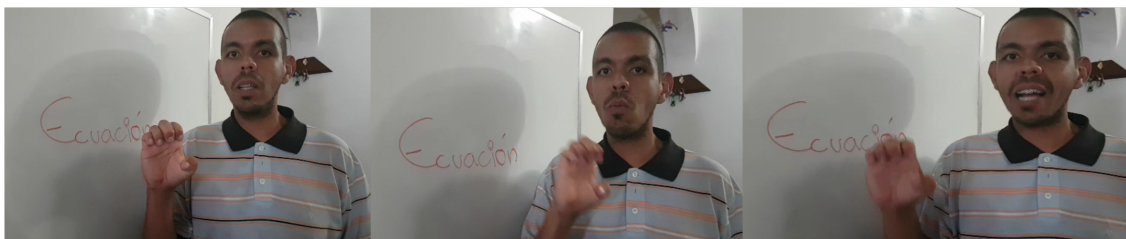


Figura 2.8: Señal ecuación

Otra de las señas creadas por el PO es la de la palabra inecuación de la figura 2.9, la mano principal empieza el movimiento en la posición de la letra “i” del abecedario dactilológico de la figura 2.4 pasando después a la posición de la letra “e” y describiendo un círculo en esta configuración.

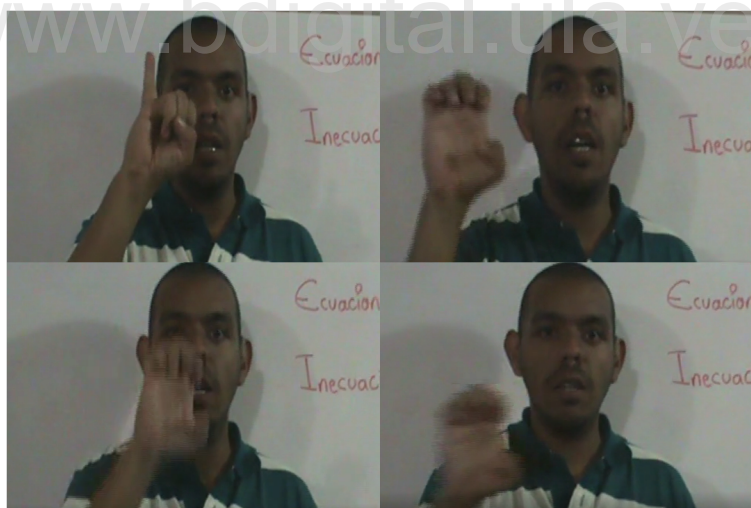


Figura 2.9: Señal inecuación

En la figura 2.10, se observa la señal de la palabra integrar. La señal comienza con las manos cruzadas frente al torso, con los dedos intercalados y luego toda la célula se mueve verticalmente hacia el abdomen del señante.

Las operaciones elementales también tienen su señal establecida, las figuras 2.11,

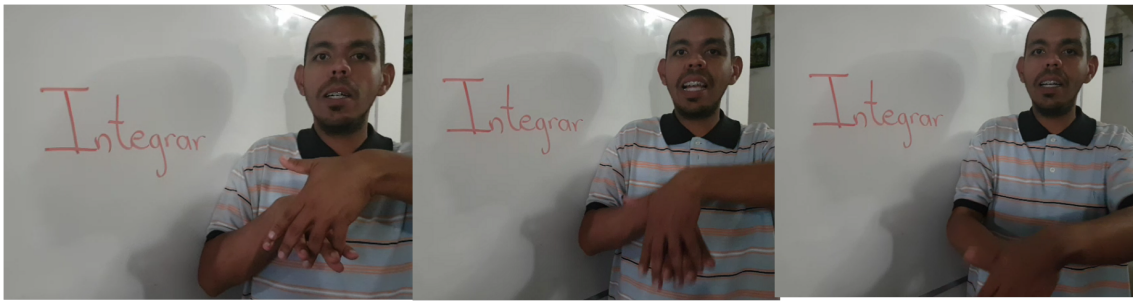


Figura 2.10: Señal integrar

2.12, 2.13, 2.14, corresponden a las señas de las palabras sumar, restar, multiplicar y dividir.



Figura 2.11: Señal sumar

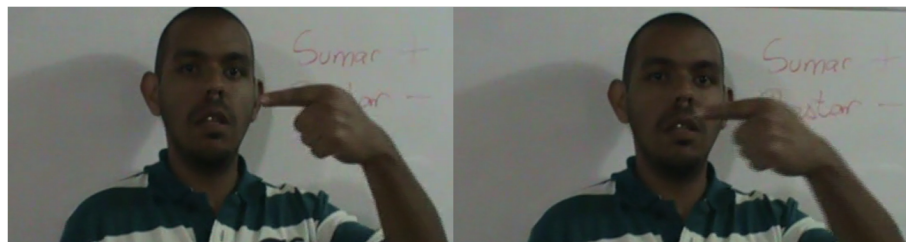


Figura 2.12: Señal restar

La tabla 2.2 muestra una lista de las palabras que se documentaron durante la realización de este proyecto, son un total de 134 señas contando las señas los números del 0 al 20, todas disponibles en el repositorio del proyecto [24] en formato mp4 y gif.

Se espera que pronto se realice una publicación sobre las señas creadas para abstraer los términos utilizados en la carrera de ingeniería de sistemas. Las señas mostradas conforman una parte de todas las señas creadas y se asume que con el paso del tiempo,

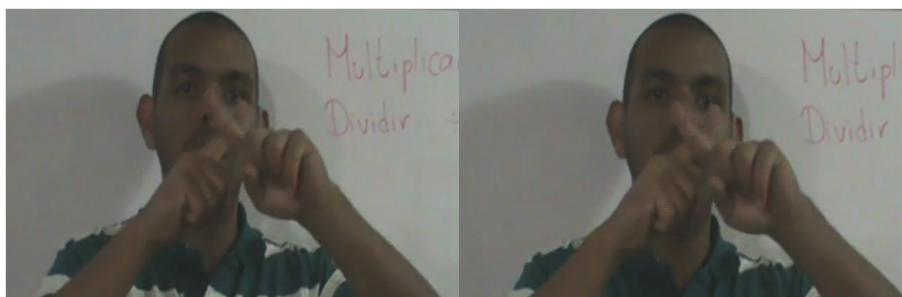


Figura 2.13: Señal multiplicar

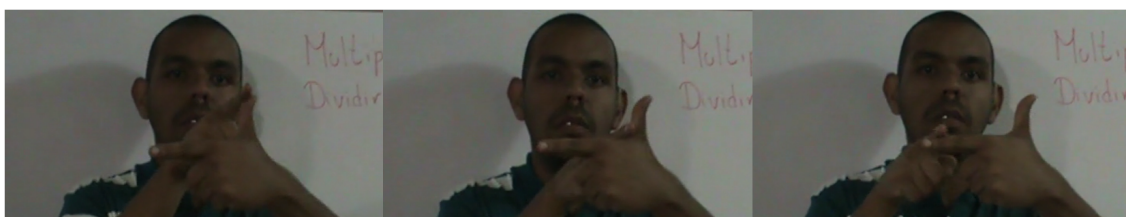


Figura 2.14: Señal dividir

se irán creando señas nuevas conforme se vayan necesitando.

ACELERACION	ALFA	ALGORITMO	ALUMNO
ANGULO	ANO	ARBOL	ARREGLO
BACHILLER	BIBLIOTECA	BINARIO	BIT
BLOQUE	BOOL	BOOLEANO	C++
CALCULO	CARGA	CARTESIANO	CENTIGRADO
CLASE	COLA	COLUMNA	COMPILAR
COMPLEJO	COMPUTACION	DATA	DATO
DECISION	DECLARACION	DEMOSTRAR	DERIVAR
DESPEJAR	DIA	DISCRETO	ECUACION
ENTERO	ENTRADA	ESPECIAL	ESTOCASTICA
ESTRUCTURA	ESTRUCTURAR	FACTOR	FALSE
FALSO	FILA	FISICA	FLOAT
FLOTANTE	FOR	FORMULA	FRACCION
FUNCION	GRADO	HORA	IMAGINARIO
INGENIERIA	INTEGRAR	KILOMETRO	LENGUAJE
LIBRERIA	LIMITE	LISTA	MAIN
MATEMATICA	MATRIZ	MES	METODO
METRO	MIENTRAS	MILIMETRO	MINUTO
NATURAL	NEGATIVO	NUMERO	OBJETO
ORIENTADO	PARAMETRO	PI	PILA
PLANO	PORCENTAJE	PRINCIPAL	PROCEDIMIENTO
PROFESOR	PROGRAMACION	PROGRAMAR	PROYECTO
PUNTO	PUPITRE	RAPIDO	WHILE
REAL	REPETICION	VELOCIDAD	VERDADERO
RETORNO	RETURN	SALIDA	SEGUNDO
SEMANA	SILLA	VECTOR	SISTEMA
SUSTITUIR	TEOREMA	TIEMPO	TIPO
TITA	TRUE	VARIABLE	

Tabla 2.2: Vocabulario documentado de la ingeniería de sistemas

Capítulo 3

Un traductor de lenguaje natural a LSV (TANtALUm)

La investigación y documentación deben ser siempre los primeros pasos para emprender la resolución de cualquier problema. Cuando se planteó la creación TANtALUm se buscó la forma de dividir el trabajo en pequeñas tareas que fueran realizables en periodos cortos de tiempo y que la secuencia de las mismas llevara a la creación del producto de software que solucionara el problema planteado y que cumpliera con los objetivos propuestos.

Como primeras tareas se plantearon la búsqueda de información sobre la cultura de LS en Venezuela y sobre sus hablantes. Esto conllevó al encuentro con los escritos de grandes lingüistas de la Universidad de Los Andes, ejemplos como el de Lourdes Pietrosevoli y Rafael Oviedo (citados previamente) en sus trabajos para describir el lenguaje, formalizarlo y darle una base sobre la cual construir cualquier producto que necesite de cierta formalidad en las definiciones, fueron clave en la investigaciones realizadas. Aún así como dice [25] en su artículo “No hay en el país una Academia de la LSV (ni su equivalente en otro lugar del mundo) y tampoco existe una gramática de esta lengua, ni diccionarios. Los textos que comúnmente son llamados diccionarios en la comunidad de Sordos son más bien manuales de señas muy utilizados para la enseñanza de la LSV a los oyentes, pero que están muy lejos de reunir los requisitos propios de la metodología científica del área lexicográfica.” aunque en la última década

el gobierno nacional impulsó la creación y manutención de institutos de asistencia a sordos no se apoyaron causas como el estudio de la lingüística y la normalización de las señas, en teoría, un sordo puede crear una seña para las cosas que no la poseen, siempre y cuando lo consulte con su comunidad de Sordos y se establezca que ésta seña univocamente define este nuevo termino, es decir no existe una seña igual a la creada que se utilice para otro término, lo cual es posible, dado que la definición del lenguaje lo contempla y a las configuraciones y herramientas con las que se cuentan en la LSV el lenguaje es infinito. En el caso de estudio que se presenta en este proyecto el PO ha creado muchas señas para describir las cosas que aprende día a día en la universidad, él y sus intérpretes son las comunidad de personas que empiezan a utilizar éstas señas y a difundirlas, no se cuenta con un manual de señas aplicables al campo de la ingeniería así que también se realizó una investigación sobre los nuevo terminos acuñados.

Una vez que el Sordo crea una seña para el término nuevo y ésta es aprobada por el grupo de Sordos se pasa a la etapa de compartir esta seña con nuevos grupos de interesados y así este término se divulga entre todos los hablantes de una comunidad en algún momento. Pero el proceso de esparcimiento de una nueva seña o el cambio en una seña existente no es documentado de ninguna manera, no hay escritos o grabaciones sobre las reuniones en las que se realizan estos acuerdos, la información se obtiene mediante la conversación con uno de los hablantes de la LSV que pertenezca a un grupo de estudio del lenguaje. El desarrollo de las reglas que rigen la sintaxis de la LSV hizo mediante la elicitación.

3.1 El proceso de elicitación

La elicitación es una metodología lingüística a la cual recurre el lingüista para indagar propiedades del lenguaje. En otras palabras “consiste en pedir al informante o hablante nativo su opinión sobre la gramaticalidad de una construcción, sus posibles interpretaciones, etc., así como ejemplos o contraejemplos de dicha construcción.”, siguiendo esta metodología se procede a realizar una lista de posibles oraciones simples divisibles por categoría con ejemplos, las cuales se muestran al PO y éste da su traducción a LSV. La lista de elicitación para oraciones simples de prueba es la que se

muestra a continuación.

- Según la voz verbal

- Activa

El perro persigue al gato

PERRO–PERSEGUIR–GATO

Yo le compré una pelota al niño.

YO–COMPRE–UNO–PELOTA–NINO

- Pasiva

El gato es perseguido por el perro.

GATO–SER–PERSEGUIR–POR–GATO

- Según el objeto

- Personales

- * De sujeto explícito

El perro ladró toda la noche.

PERRO–LADRAR–SENAPASADO–TODO–NOCHE

- * De sujeto implícito

He visto cosas increíbles.

YO–HABER–SENAPASADO–VER–COSA–MUCHO–INCREIBLE

- Impersonales

- * Con verbo meteorológico.

No ha parado de llover.

NO–HABER–PARAR–DE–LLOVER

- * Gramaticales

No había ningún refresco en la nevera.

NO–HABER–NINGUNO–SENAPASADO–REFRESCO–EN–NEVERA

- Según la naturaleza del predicado

- Copulativas o atributivas

La comida parece deliciosa.

COMIDA–PARECER–DELICIOSA

Yo soy el profesor.

YO–SER–PROFESOR

* Predicativas

- Transitivas

Estuve viendo la casa.

YO–ESTAR–SENAPASADO–VER–CASA

José caminaba por la calle.

J-O-S-E–CAMINAR–SENAPASADO–POR–CALLE

- Reflexivas

Yo me ducharé.

YO–ME–DUCHAR–SENAFUTURO

- Recíprocas

El gato y perro se pelean.

GATO–Y–PERRO–SE–PELEAR

- Intransitivas

José ama.

J-O-S-E–AMAR

- Según la actitud del hablante

- Exclamativas

María me prestará su carro.

M-A-R-I-A–ME–PRESTAR–SU–CARRO

- Negativas

María no me prestará su carro.

M-A-R-I-A–NO–ME–PRESTAR–SU–CARRO

- Imperativas o exhortativas

María préstame tu carro

M-A-R-I-A-PRESTAR-ME-TU-CARRO

- Desiderativas

Ojalá María me preste su carro

OJALA-M-A-R-I-A-ME-PRESTAR-SU-CARRO

- Dubitativas

Tal vez María me prestará su coche

TAL-VEZ-M-A-R-I-A-ME-PRESTAR-SENAFUTURO-SU-CARRO

Dado que una oración compuesta está formada por dos o más oraciones simples, se asume que se pueden traducir oraciones compuestas traduciendo las oraciones simples que la forman y concatenando el resultado.

3.2 Reglas de sintaxis a partir de la elicitación

En la lista de elicitación seguramente se escapan algunos tipos especiales de oración, pero se lograron asentar las normas reflejadas en las tablas 3.1 y 3.2. Estas reglas van a regir el proceso de análisis léxico de TANtALUm y serán utilizadas nuevamente en el proceso de pruebas así como también los ejemplos provistos en la lista de elicitación.

3.3 Diseño TAMtALUm

Por otra parte, la investigación previa también comprende la interrogante de cómo procesar el lenguaje, qué herramientas existen y cuál es la herramienta ideal para el desarrollo de un proyecto de este tipo. La elección de la biblioteca Freeling para PLN se llevó a cabo gracias a consultas vía email con el profesor Arturo Curiel de la Universidad Nacional Autónoma de México (UNAM) que cuenta con gran experiencia en el tema de los sistemas traductores de LS gracias a su trabajo de doctorado con la lengua de señas francesa titulado “Reconocimiento Automático de la Lengua de Señas” [26],

Nro	Regla	Ejemplo
1	Se omiten todos los determinantes de tipo artículo.	El carro - CARRO
2	Cuando el sujeto es una persona, sea un nombre común, pronombre personal, demostrativo o indefinido, determinante demostrativo de género masculino, se debe agregar a la secuencia de lemas la seña de sexo masculino luego de el sujeto, siempre y cuando la palabra el genero no sea implicito al sujeto.	Niño - NINO SENAMASCULINO
3	Cuando el sujeto es una persona, sea un nombre común, pronombre personal, demostrativo o indefinido, determinante demostrativo de género femenino, se debe agregar a la secuencia de lemas la seña de sexo femenino luego de el sujeto	Niña- NINA SENAFEMENINO
4	Cuando el número del sujeto es plural se debe agregar la seña mucho después del genero si aplica o en caso contrario después del sujeto.	Niñas- NINA SENAFEMENINO MUCHO
5	Los nombres propios siempre se deletrean.	María - M A R I A
6	En oraciones reflexivas, el pronombre personal que indica la flexión se coloca antes del verbo y después del sujeto.	Yo me ducho - YO ME DUCHAR
7	Los tiempos verbales se indican con una seña posterior al verbo en infinitivo, con excepción del verbo ser (regla 8)	Yo me fuí - YO ME IR SENAPASADO
8	El tiempo verbal del verbo SER posee una seña diferente para el pasado, presente y futuro.	Yo soy feliz - YO SERPRESENTE FELIZ Yo fuí feliz - YO SERPASADO FELIZ

Tabla 3.1: Reglas LSV por elicitación

Nro	Regla	Ejemplo
9	Cuando no existe un sujeto explícito, colocar el sujeto que indique la conjugación antes del verbo.	Comí torta. - YO COMER SENAPASADO TORTA
10	Si ya se tiene un usuario en la oración los nombres comunes siguientes no tienen número, es decir, no se les agrega la seña mucho. Pero si son nombres comunes pertenecientes a humanos si se les agrega la seña de género distintiva.	Las niñas comen pasteles - NINA SENAFEMENINO MUCHO COMER PASTEL
11	Se se encuentra un adverbio negativo, éste se debe colocar entre el sujeto y el verbo	Yo no soy venezolano- YO NO SER VENEZOLANO Ninguno quiere comer- NINGUNO QUERER COMER
12	Si se tiene un verbo de la tercera persona de género no especificado y la oración no tiene un sujeto explícito entonces no se agrega el sujeto de la conjugación ni el género del mismo.	Quiere comer - QUERER COMER
13	Si se tiene un verbo de la primera o segunda persona del singular o plural entonces se agrega el sujeto de la conjugación	Quiero comer - YO QUERER COMER

Tabla 3.2: (continuación) Reglas LSV por elicitación

en éstas consultas también se trataron los temas de elección de metodología análisis de lenguaje, el cual se decantó por un análisis léxico debido a que no hay suficiente conocimiento lingüístico en LSV para hacer traducción basada en reglas (mucho menos machine learning) y al trabajo de Traducción de LESCO [13] el cual utiliza Freeling para la tokenización, splitter y PoS tagging.

La tarea principal de TANtALUM es llevar un texto escrito en español a alguna

representación que pueda ser entendida como señas, en esta implementación esta salida se describe como un flujo de lemas. Un lema es una palabra que describe a una seña, los lemas son la forma canónica que toma una palabra, la raíz, una entrada de diccionario. Por ejemplo, corres, corremos, corrían son palabras pertenecientes al mismo lexema, que poseen el mismo lema correr, gracias a su definición un lema puede ser utilizado como elemento de búsqueda dentro de un diccionario de datos, por lo tanto un lema se convierte en la representación ideal de la salida, porque un lema es la forma perfecta de buscar dentro de una gran base de datos de señas la seña correspondiente y además, como se puede observar en la lista de elicitación las salidas provistas por el PO son una secuencia de lemas.

En general el proceso de traducción lleva 3 pasos: el texto en lenguaje de entrada se lleva a una representación intermedia, se realizan modificaciones a esta representación intermedia y por último esta se transforma al lenguaje destino [17].

El traductor de lenguaje natural a LSV (TANtALUm), es el módulo central del sistema traductor de voz a LSV que se mencionó en el capítulo 1 y como se muestra en la figura 3.1, sigue la secuencia recomendada de acciones.

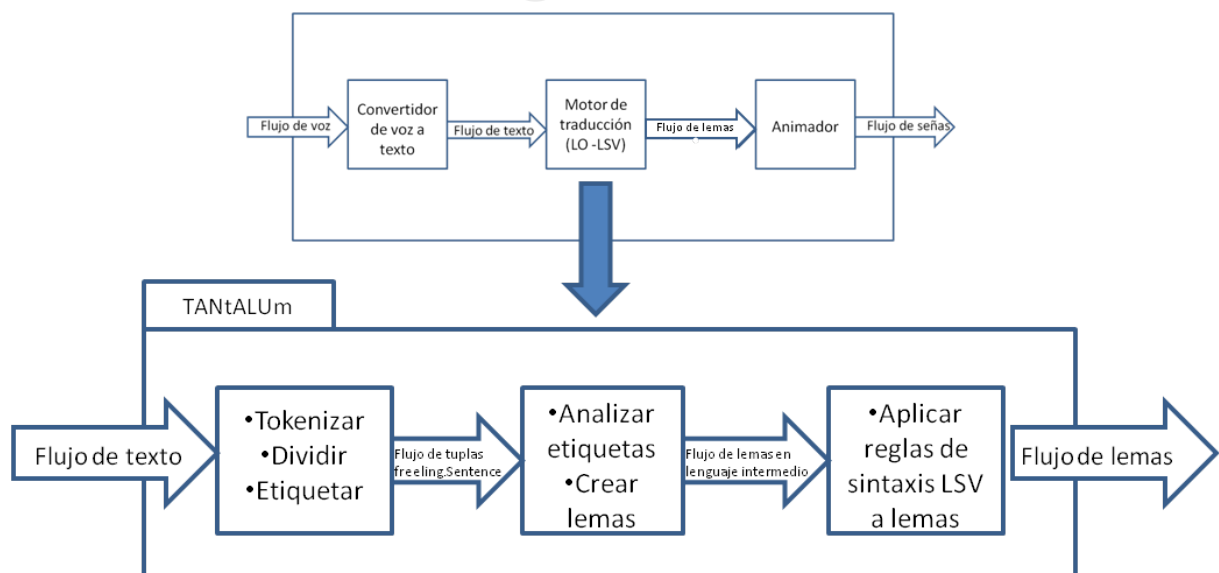


Figura 3.1: El proceso de traducción de 3 pasos.

La entrada al sistema es un flujo de texto que debe cumplir con 2 condiciones:

- Debe estar escrito en español.
- Debe tener una ortografía correcta.

Como se mencionó en el capítulo anterior el uso del alfabeto para deletrear las palabras es muy común. Cuando no se recuerda una seña entre practicantes de la LSV cuya lengua materna no es la LSV, se suele recurrir a deletrear la palabra, aunque es muy difícil que esto le ocurra a un Sordo porque en ese caso se recurre al uso de otras señas para explicar lo que se desea decir. Sin embargo, en este caso y por cuestiones de agilizar el proceso comprensivo del usuario del sistema, se acordó que cuando el flujo de entrada de texto no está bien escrito o no se encuentra una seña registrada, el traductor toma la palabra como un nombre común y se muestra en pantalla para que no se pierda la información que contiene el mensaje. Con respecto a la entrada del sistema, cuando se habla de un flujo de texto, se habla de una cadena de texto que viene del sistema de reconocimiento de voz y entra a TANtALUm. En el primer paso o analizador morfológico se llevan a cabo las tareas de tokenización, división (splitter), desambiguación y etiquetado (tagger). TANtALUm, se utiliza Freeling en su versión 4.1 para estas tareas con algunas modificaciones en el archivo de configuración de la biblioteca para que realizar el split de las oraciones en búsqueda de tener oraciones simples que cumplan con alguno de los criterios planteados en la lista de elicitación, para la elección de estas palabras que suelen usarse como marcadores conversacionales se toman en cuenta los resultados de [27] donde se concluye que la conjunción copulativa “y” es utilizada como marcador conversacional en el 40% de los casos, además también se utilizan como divisores la “,”, el “.” y un máximo de 100 palabras por oración.

En el analizador morfológico el tagger, le asigna a cada objeto word una etiqueta que gracias a la desambiguación de términos es correcta en el 96% de los casos. El tag de cada palabra es lo que permite hacer el análisis léxico. Las tags están basadas en el etiquetado EAGLES, que fue mostrado en refcap:2.

El segundo paso para la traducción, es llevado a cabo por el analizador léxico como se ve en el diagrama procesos de TANtALUm 3.2. La salida del analizador morfológico es un objeto de tipo paragraph, el cual está compuesto como se muestra en la figura 2.2 por una lista de objetos del tipo sentence y éstas a su vez son una lista de objetos

tipo word. Se itera sobre los componentes del paragraph y cada sentence es analizada palabra por palabra, para comprobar las palabras útiles que contiene.

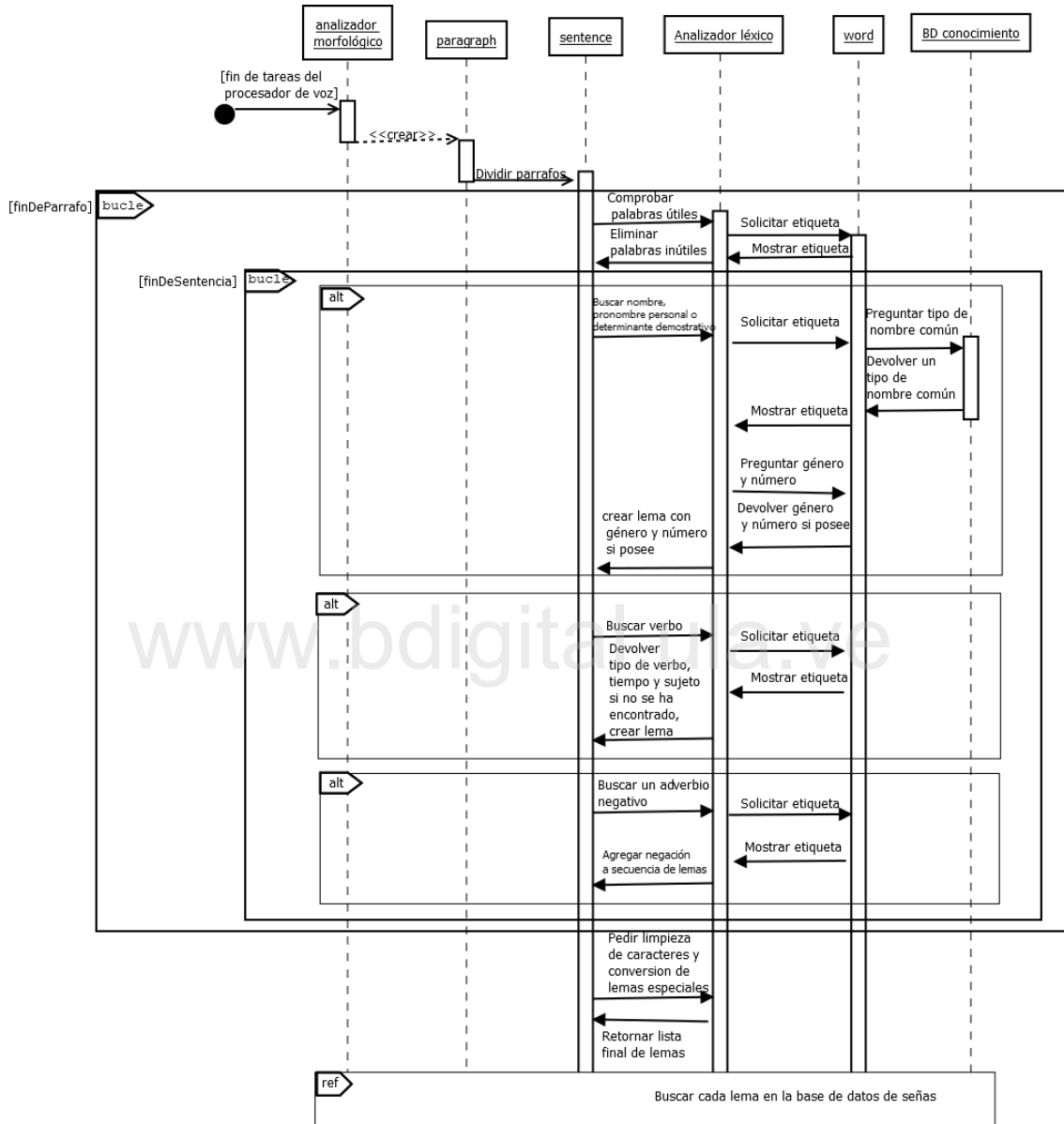


Figura 3.2: Diagrama de procesos, TANtALUM

La eliminación de palabras o limpieza previa se realiza gracias a una lista que contiene los tags de las palabras que se desean evitar. La etiqueta de la palabra que se desea evaluar se compara con cada una de las etiquetas de la lista de tags a evitar y luego con expresiones regulares se buscan coincidencias, las palabras que no son

aceptadas se van añadiendo a un nuevo objeto sentence, las que si, simplemente se ignoran. Al finalizar el nuevo objeto sentence contiene las palabras que nos interesa traducir.

El siguiente paso es analizar palabra por palabra la nueva oración limpia. Se analizan entonces las etiquetas de la palabra y se busca un verbo principal seguido de un verbo en infinitivo, para analizar las oraciones reflexivas que requieren tratamiento especial, por ejemplo: Si la entrada de traductor es, “Quiero comerme una pizza”, la salida del analizador morfológico es [querer comer me pizza]. Según las reglas de la tablas 3.1 y 3.2 el pronombre “me” se debe eliminar y por la ausencia de sujeto éste debe ser extraído del verbo principal luego se debe crear una cadena que contiene la secuencia de lemas que corresponden, en este caso sería “YO QUERER COMER PIZZA”.

El proceso empieza con la búsqueda de un nombre, se identifican los nombres como las estructuras más perdurables en el tiempo según [8], es debido a esto la importancia de buscar como primer elemento un nombre. Cuando se reconoce una palabra como nombre se pasa entonces a buscar entre las características de su etiqueta si es un nombre común o un nombre propio, cuando se encuentra un nombre común y es el primer nombre en la oración en ser encontrado, entonces se busca su valor dentro de la base de datos de conocimiento proveída por Freeling (SUMO) y se evalúa si es una palabra que en su significado define a un humano, por ejemplo: niño, mujer, dama, etc. En este caso se busca de la palabra el género y el número, en caso contrario se busca el número solamente y cuando la etiqueta de la palabra es un nombre propio se deletrea el nombre, no se busca género ni número. Si no se encuentra un nombre se pasa a buscar entonces la segunda célula más importante dentro de la oración, un verbo.

Al encontrar un verbo, se encuentran dos casos posibles, si ya se ha encontrado un nombre se pasa entonces a asumir que ese nombre es el sujeto, entonces se debe analizar el verbo en búsqueda de su tiempo y éste se concatena al verbo que se está analizando luego del sujeto, si no existe un sujeto antes de encontrar un verbo, entonces el verbo es analizado en búsqueda de un sujeto en su etiqueta y el tiempo correspondiente, siempre y cuando el género del verbo esté definido, para verbos con género indefinido no se concatena el pronombre personal que identifica al sujeto. Luego de esto se pasa

entonces a buscar una negación, en la oración sobre la cual se está iterando.

Cuando la oración posee un adverbio de negación se hace una marca en búsqueda de un sujeto, si ya existe un sujeto entonces se concatena el lema de la negación después del sujeto, si no se ha encontrado un sujeto, se debe esperar la aparición de un verbo y extraer de éste el sujeto, si es así entonces se concatenan el lema sujeto, seguido de la negación y el verbo que se analizó en caso contrario se concatena el lema del adverbio de negación y el verbo.

Cuando se termina la iteración sobre la oración entonces se asume que el análisis léxico ha llegado a su fin, después de hacer una verificación de orden final, se debe limpiar el contenido de la cadena de lemas, quitándose los especiales como ' y además de los caracteres espacios extras que puedan aparecer debido a las concatenaciones según los criterios que se muestran en la tabla 3.3.

Candidato	Sustituto	Candidato	Sustituto
Á	_A	E	_E
Í	_I	Ó	_O
Ú	_U	Ñ	_N

Tabla 3.3: Caracteres sustituibles en el proceso de limpieza

La salida de todo este proceso es una cadena de lemas limpia y lista para ser utilizada.

Capítulo 4

Implementación y pruebas

4.1 Creando un traductor de texto a LSV

Las tareas de tokenización, splitter, desambiguación y tagging corren por parte de la biblioteca, el código base que realiza éstas tareas se extrae de [28] y puede ser encontrado en el proyecto en el archivo TAMtALUm.py, si embargo se recomienda hacer todas las actividades del tutorial de Freeling antes de empezar con el estudio de la implementación de TAMtALUm.

La salida del analizador morfológico es un objeto de tipo paragraph, que consta de una lista de oraciones, Python permite iterar dentro de las palabras de cada oración por medio de 2 estructuras con **for** anidados.

```
for sentence in ls :  
    for word in s:
```

Donde *ls* es la salida del analizador morfológico, *sentence* es cada una de las oraciones y *word* cada una de las palabras que conforman las oraciones, cada objeto del tipo *word*, tiene 3 funciones de consulta muy importantes: la consulta del lemma, del tag y de su significado.

```
word.get_lemma()  
word.get_tag()  
word.get_senses()
```

Cada una de los valores devueltos por cada palabra son clave en el proceso del análisis léxico, así que se guardan en una variable.

El proceso llamado *Comprobar palabras útiles* de la figura 3.2 consiste en el análisis de cada etiqueta mediante expresiones regulares en búsqueda de las palabras que no son utilizadas dentro del lenguaje, las expresiones regulares de las palabras cuyas etiquetas se “saltan” en el análisis son:

- $\hat{D}A$: Las palabras cuyas etiquetas comienzan con DA son artículos y no se toman en cuenta en el LSV.
- Fd, Flt, Fla, Fe, Frc, Fra, Fx, Faa, Fat, Fia, Fit, Fp: Son etiquetas de diferentes signos de puntuación como !, }, {, etc.
- $\hat{P}[0-3A-Z]\{4\}D$: Expresión que identifica a los pronombres dativos, como *le*, *les* en LSV no son utilizados.

El significado completo de las etiquetas se encuentra disponible en [22].

Después de eliminar las palabras que no son útiles, cada oración (sentence) se analiza palabra por palabra, en búsqueda de nombres luego verbos y finalmente adverbios de negación. Esto basado en el estudio realizado en 3 se encontró que estas palabras son las que cambian drásticamente y marcan una diferencia en la sintaxis del LSV.

Para el análisis de nombres se sigue el siguiente procedimiento mostrado en el diagrama de actividades de la figura 4.1.

Si para este momento no se tiene un nombre que pueda ser utilizable como sujeto, se empieza a buscar entre las posibles palabras utilizables en el proceso de pronominalización, los pronombres y los determinantes determinados. Su análisis consiste en determinar si un pronombre define a un humano o no y asignar un valor de contingencia de existencia de sujeto para que en la iteración siguiente de la oración se pueda analizar el nombre correspondiente o no, en caso de que el pronombre o determinante no venga seguido de ningún nombre se debe extraer el género y número del pronombre o del determinante en caso contrario el género y el número son valores que se dejan en vacío, el análisis de pronombres y determinantes es el mostrado en la figura 4.2.

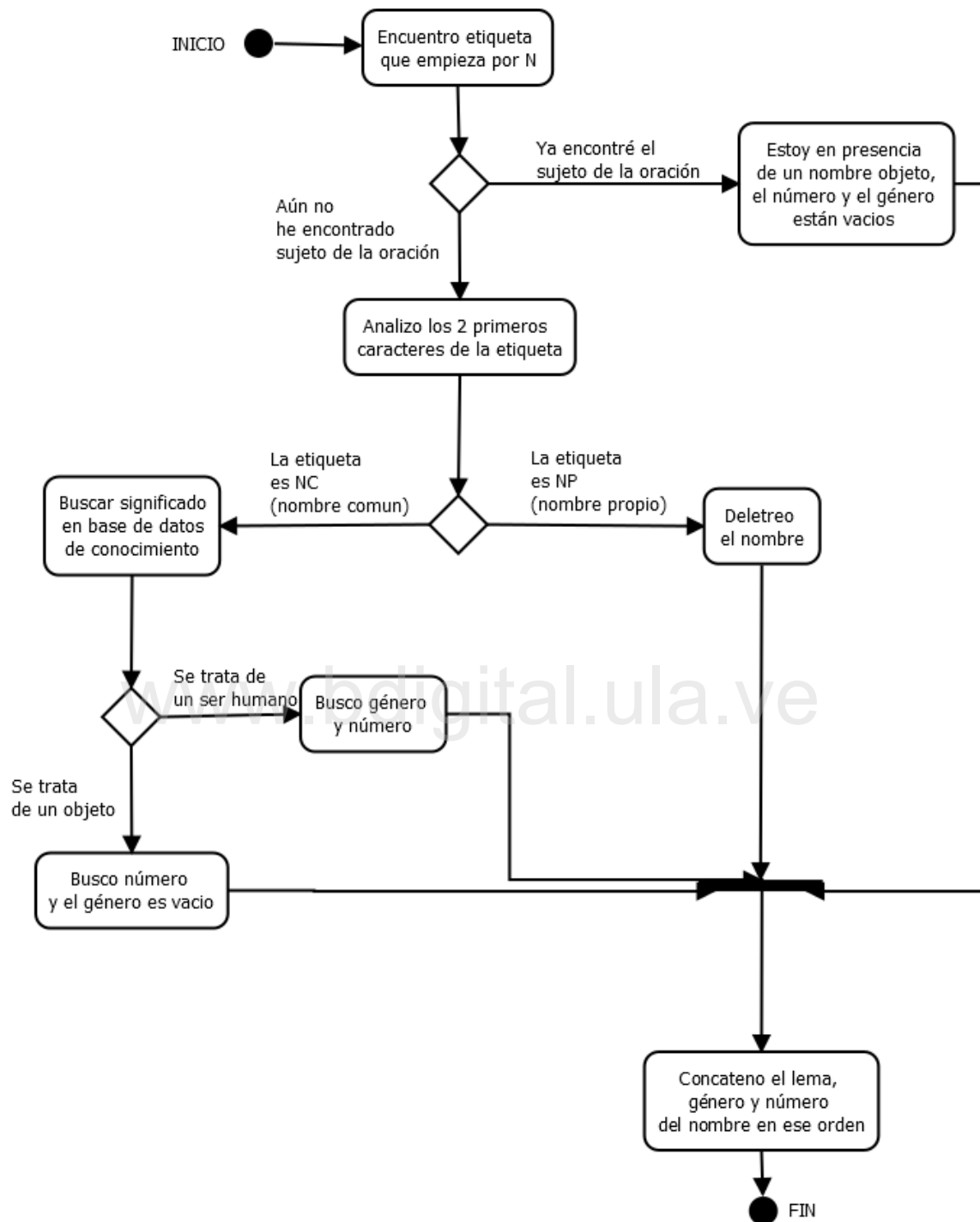


Figura 4.1: Diagrama de actividades para el análisis de nombres

Por la estabilidad temporal de datos en el discurso expuesta en [3], el siguiente valor a analizar son los adjetivos, pero gracias al proceso de elicitación se determinó que el

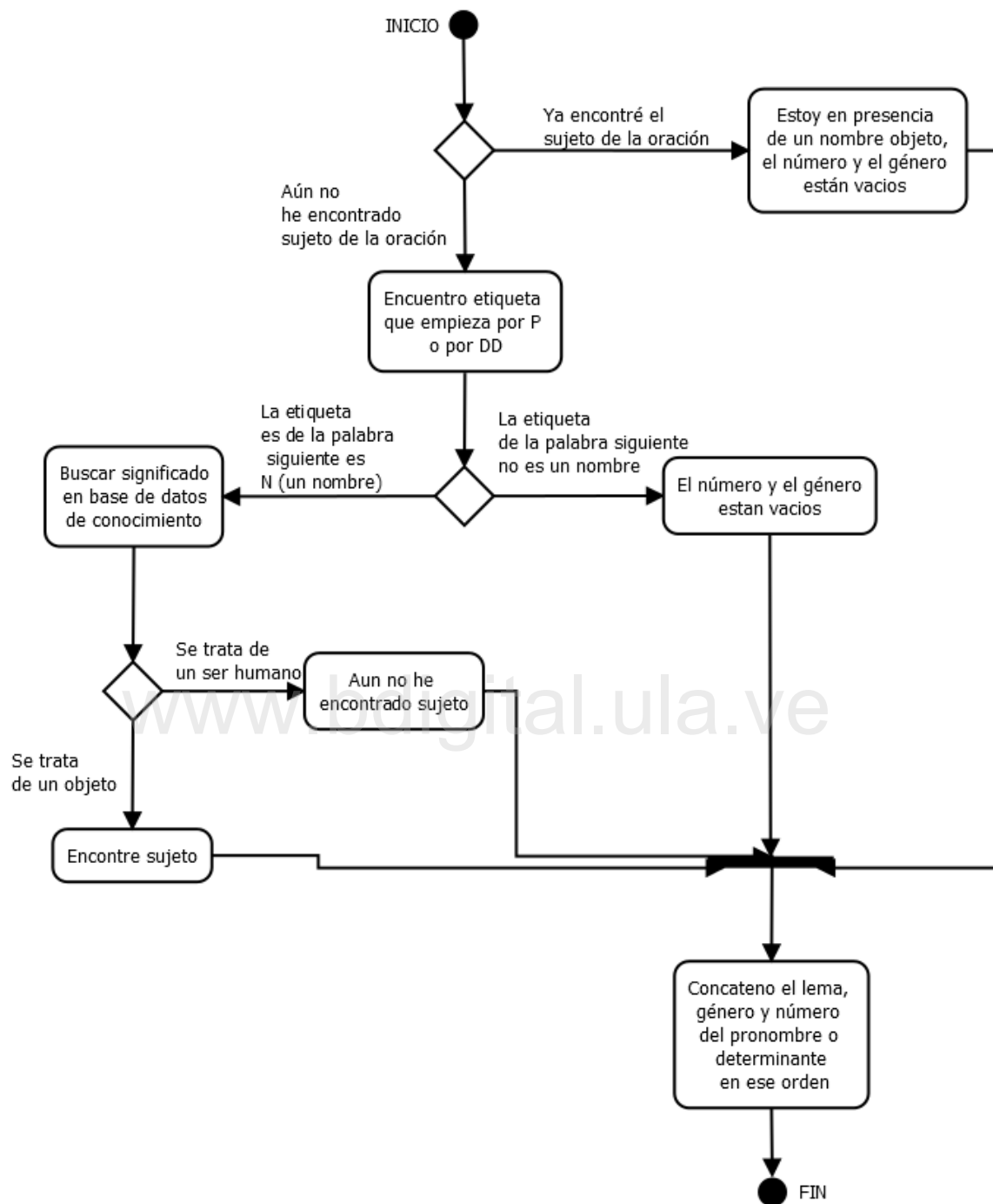


Figura 4.2: Diagrama de actividades para el análisis de pronombres y determinantes determinados

análisis de los mismos no es necesario para la sintaxis de la LSV, por lo tanto se pasan a analizar los verbos.

Los verbos constituyen junto con los nombres el corazón de la LSV, un nombre seguido de un verbo así como en las LO pueden expresar una idea completa. En esta parte es necesario saber si se ha encontrado una estructura de negación previa y un sujeto pues las negaciones se colocan antes del verbo y después del sujeto, tener una cota superior e inferior hace el análisis de verbos y adverbios de negación se haga en paralelo. Los pasos para el estudio de los verbos se muestran en la figura 4.3

Para finalizar el proceso de análisis, el flujo de lemas vuelve a ser limpiado, las etiquetas creadas en el proceso que identifican a un número o a un género son sustituidas por etiquetas que inequívocamente describen una seña, los caracteres especiales son sustituidos y los dobles espacios en blanco eliminados quedando entonces el flujo de lemas listo para ser procesado por cualquier buscador configurado para buscar el nombre de una seña dentro de una colección y luego mostrarla.

Todos los códigos implementados para cumplir con las tareas de diseño descritas en el apartado anterior se encuentran en un repositorio público de licencia GNU de TAMtALUm [24].

Se documentaron 416 señas correspondientes a adverbios, pronombres, letras del abecedario, días de la semana, conjunciones, preposiciones, verbos y adjetivos para el uso de la versión demo, todas estas señas se listan en la tabla 4.1 y 4.2, además pueden ser descargadas para su posterior utilización del repositorio [24].

4.2 Pruebas

El proceso de pruebas inició con pruebas unitarias que buscaban cumplir con los los items planteados en la lista de elicitación.

Para los elementos de la tabla 4.3 se obtuvieron salidas correctas, pero con pantallas de texto, es decir muchas de las palabras que se traducen no tienen una seña guardada dentro del sistema de pruebas y se obtienen salidas con patallas negras, por ejemplo para el flujo de entrada: Juan caminó hasta muy tarde obtenida es: J U A N COMER SENA_PASADO HASTA MUY TARDE, la secuencia de gif obtenida es la de la figura 4.4.

Por el contrario si se usan las palabras que se tienen en el catalogo de señas de

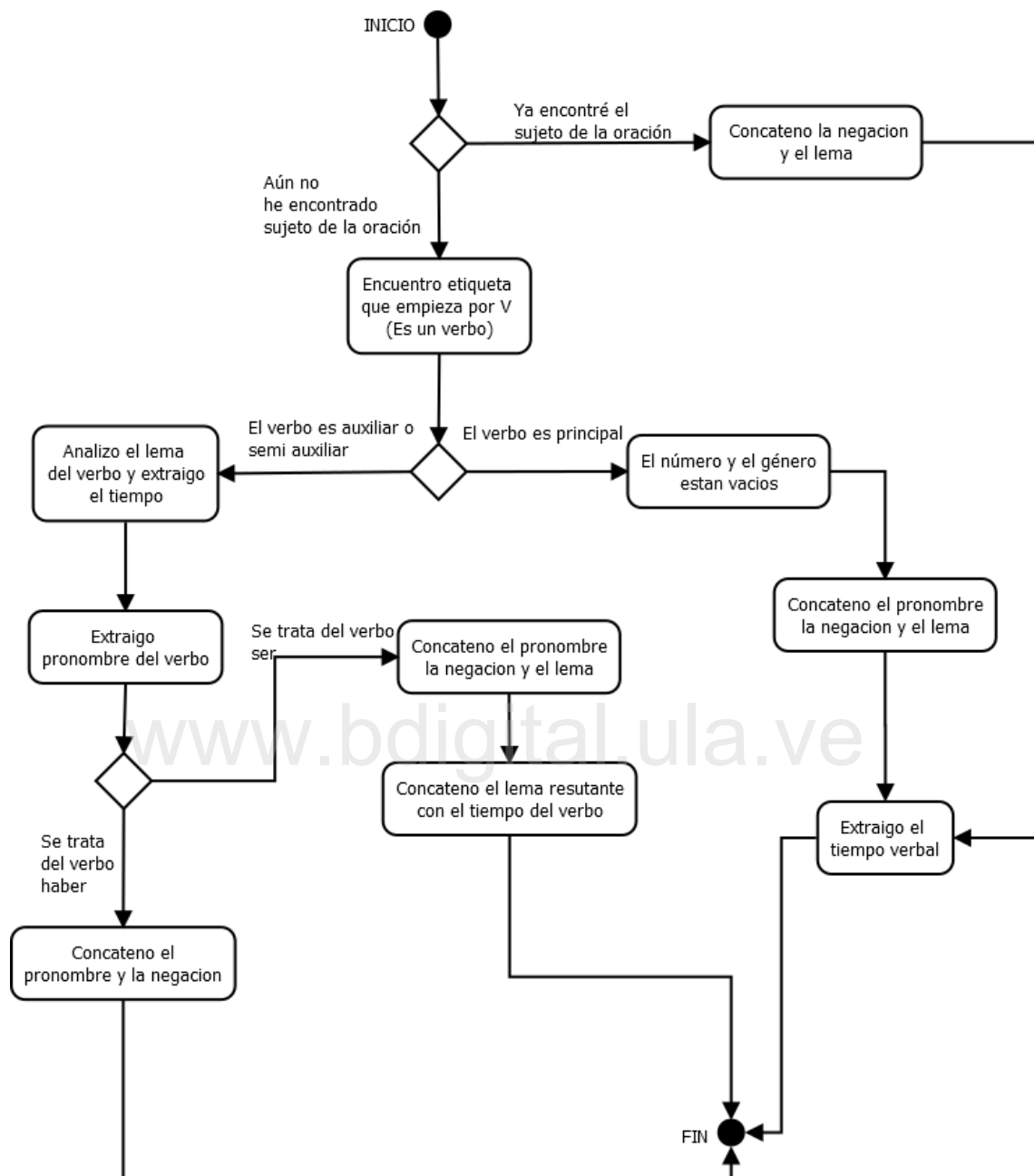


Figura 4.3: Diagrama de actividades para el análisis verbos

la tabla 4.1, 4.2 o de la tabla 2.2 se obtiene una secuencia de gif producto de la traducción, como el de la figura 4.5 cuyo flujo de entrada fue: abres el computador y las señas indicadas para la traducción son TU ABRIR COMPUTADOR.

Buscando una salida cada vez más parecida a los casos planteados en la lista y

YO	TU	EL	NOSOTROS	USTEDES
ELLOS	CUAL	QUIEN	BIEN	MAL
SENAPASADO	APROVECHAR	TARDE	ESTE	HOLA
MUY	ESTAR	SERPRESENTE	COMO	AHORA
SENA	SOLAMENTE	NADA	SI	NO
TAMBIEN	CUANDO	SENAPRESENTE	ANTES	DESPUES
NUNCA	SIEMPRE	SENAFUTURO	ABAJO	ARRIBA
HABER	SERPASADO	AQUEL	AQUELLOS	DETRAS
DONDE	FRENTE	FUERA	LEJOS	SOBRE
MAS	MENOS	POCO	PERO	DIFERENTE
PORQUE	SINO	Y	SE	ES
DE	EN	ENTRE	HACIA	PARA
QUE	SIN	MI	GRACIAS	FAVOR
CHAO	SALUDO	ABRIR	ACABAR	ACEPTAR
AHORRAR	AGARRAR	ALINEAR	AMAR	APLAUDIR
APRENDER	APURAR	ARREGLAR	CORTAR	AUMENTAR
SERPASADO	BUSCAR	CAER	CALENTAR	CALMAR
SERFUTURO	BOTAR	MUY	TRADUCIR	TAMPOCO
ENTONCES	DELANTE	TEMPRANO	CUANTOS	IGUAL
CON	POR	ACOMPANAR		

Tabla 4.1: Catálogo de señas extras

no restringiéndose al contenido de la misma se hicieron también pruebas unitarias con respecto al vocabulario usado en EISULA en la tabla 4.4 se observan las salidas obtenidas para entradas de distinta longitud en las que se utiliza un vocabulario especializado, cabe resaltar el lenguaje técnico suele ser simple gramaticalmente, porque suele buscar que el receptor entienda claramente lo que se pide, si por ejemplo se usan los ejemplos de la tabla 4.5 se encuentran casos en los que la gramática desarrollada es más visible.

Para la entrada de texto “el teorema demuestra que 0 es el límite de X” cuyo flujo de lemas de salida es: TEOREMA DEMOSTRAR QUE 0 SER.PRESENTE LIMITE



Figura 4.4: Secuencia de señas para: Juan comió tarde



Figura 4.5: Secuencia de señas para: abres el computador

DE X y su secuencia de señas la que se muestra en la figura 4.6.



Figura 4.6: Secuencia de señas para: el teorema demuestra que 0 es el límite de X

Cabe destacar que mientras se hacían pruebas surgieron nuevos problemas que resolver que por alguna u otra razón se escaparon de iteraciones pasadas, también se realizaron pruebas con frases en las que se utilizan en el lenguaje especializado de la

carrera de ingeniería de sistemas, obteniendo resultados muy buenos, pues en el diseño del proyecto no se planteó la traducción directa de una estructura de palabras dadas sino la traducción de células gramaticales que abarcan muchísimas combinaciones de palabras.

Junto al PO, se realizaron además muchas pruebas de caja negra para validar que las traducciones obtenidas fueran correctas, en estas pruebas se encontraron errores principalmente de los casos de nombres propios, los cuales deben ser empezar por una letra mayúscula que los distinga, caso similar ocurre cuando no se cuidan los signos ortográficos, así como también cuando se encuentran palabras en inglés, por ejemplo: Para la frase [María está en clases] la salida correcta es

M-A-R-I-A ESTAR EN CLASES

Pero si se incumplen las precondiciones establecidas en 3.3 se obtienen salidas sin sentido.

- Error en acentos

María esta en clases

M-A-R-I-A ESTE SENAFEMENINO EN CLASE

- Error en escritura de nombres propios

maría está en clases

MARIA ESTAR EN CLASE

- Error en lectura de palabras en inglés

María it's en clases

M-A-R-I-A IT'S EN CLASE

4.3 Configuración del entorno de trabajo

Implementar a TAMtALUm, fue una tarea que se llevó a cabo en el ambiente de desarrollo Jupyter Notebook.

Jupyter Notebook es un proyecto de código abierto sin fines de lucro, surgido del Proyecto IPython en 2014, que evolucionó para admitir la ciencia de datos interactiva y la informática científica en todos los lenguajes de programación. Jupyter es un software 100% de código abierto, gratuito para utilización bajo los términos liberales de la licencia BSD modificada [29].

Según el portal del proyecto Jupyter es recomendable su instalación en conjunto con el proyecto Anaconda. Sin embargo, TAMtALUm se descargó y se instaló con pip3.

Pip3 es un sistema de gestión de paquetes utilizado para instalar y administrar paquetes de software escritos en Python, disponible en los repositorios de cualquier distribución de Linux.

Para instalar Jupyter sin Ananconda es necesaria la previa instalación de Python, TAMtALUm utiliza Python 3 en toda su implementación.

La configuración de Jupyter para trabajar con Freeling requiere la previa instalación de la biblioteca en el sistema operativo, La forma más fácil es instalar es desde un paquete binario pre-compilado. Los paquetes para la última versión estable se proporcionan para Ubuntu/Debian Linux y Windows [28].

TAMtALUm se desarrolló en Ubuntu 16.04 LTS, como requisitos previos se requiere la instalación de Python 3 con:

```
sudo apt-get install python3.5
```

Para la instalación de la biblioteca Freeling es necesaria la configuración de algunas bibliotecas que son parte de los requerimientos previos, con los siguientes comandos en terminal de Ubuntu:

```
sudo apt-get install libboost-regex-dev libicu-dev zlib1g-dev  
sudo apt-get install libboost-system-dev  
libboost-program-options-dev libboost-thread-dev
```

Luego es necesario ir al directorio donde se descargó el paquete de Freeling y ejecutar:

```
sudo dpkg -i freeling-4.1.deb
```


Este comando configura la biblioteca en el sistema. Para configurar la API de compatibilidad con Python, se deben seguir los pasos del archivo del archivo README ubicado en el directorio APIs/Python3/ del directorio donde se instaló Freeling.

Al completar estos pasos se puede pasar a la instalación del Jupyter con pip. En primer lugar debe comprobar que se tiene instalado pip, si no lo tiene se puede ser instalar con:

```
sudo apt-get install pip3
```

Posteriormente puede descargar e instalar Jupyter con:

```
pip3 download Jupyter
```

```
pip3 install Jupyter
```

Para ejecutar Jupyter, se debe ejecutar el siguiente comando en la consola:

```
jupyter notebook
```

Este comando abre una ventana del explorador con una estructura de directorios de fácil comprensión que permite editar distintos tipos de archivo como .txt y .py. Sin embargo, la mejor de las herramientas es el notebook de Jupyter.

El notebook amplía el enfoque basado en la consola, proporcionando una aplicación web que captura todo el proceso de cálculo: se puede desarrollar, documentar y ejecutar código, así como comunicar los resultados en un solo cuaderno [29].

Al culminar este paso se tiene un entorno listo para la utilización de Freeling y para probar TAMtALUm.

4.4 Creación de un módulo de pruebas

Para probar las posibilidades de integración que tiene TAMtALUm se desarrollaron 3 versiones de prueba, cada una con una interfaz de entrada distinta pero con la misma salida, un gif resultante de la concatenación de las señas que traducen la entrada. La versión de pruebas mostrada en la figura 4.7 es un modelo demo del sistema que se desea implementar en primera instancia con TAMtALUm, su entrada es un flujo de voz, que se convierte en texto gracias a la API de Google para Python, la cual cuenta con un reconocimiento de flujo de voz acertado de casi un 98% pero con limitaciones

de uso (solo se pueden hacer 50 consultas al día), esta gran desventaja impide su implementación en futuros proyectos pero gracias a su portabilidad y fácil instalación es la herramienta elegida para esta versión de prueba.

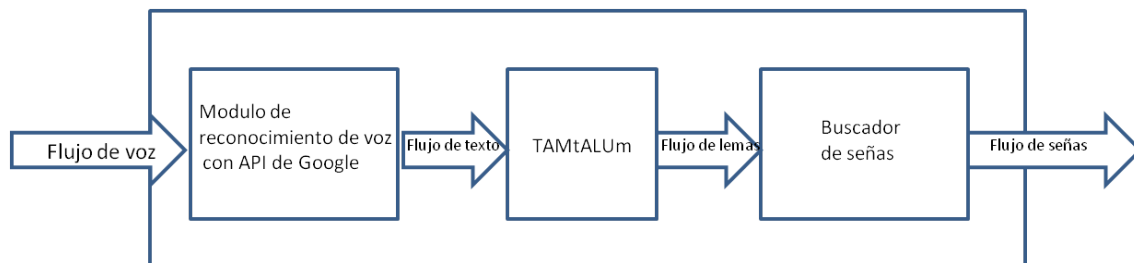


Figura 4.7: Módulo de pruebas, TAMtALUM

Las bibliotecas necesarias para la ejecución de esta version de prueba son:

- De IPython.display: Image
- De PIL: Image, ImageFont e ImageDraw
- Toda la biblioteca speech_recognition
- Toda la biblioteca moviepy.editor
- Toda la biblioteca TAMtALUM

Estas bibliotecas pueden ser instaladas con pip3 y TAMtALUM se encuentra disponible en [24].

El flujo de texto que sale del reconocedor de voz es guardado en una variable y debe ser pasado al tokenizador, splitter, desambiguador y tagger como se muestra a continuación:

```

lw=tk.tokenize(text.replace('\n', ' '))
ls=sp.split(lw)
ls=morfo.analyze(ls)
ls=tagger.analyze(ls)
ls=sen.analyze(ls);
ls=wsd.analyze(ls);
out=ProcessSentences(ls, sdb)

```

La función `ProcessSentences` es el analizador léxico, el corazón de TAMtALUm.

La salida del traductor se guarda en una variable, que pasa a ser dividida como una lista por los espacios en blanco que posee. Cada elemento de la nueva lista es un lema. En la dirección donde se está ejecutando el traductor existe un directorio llamado `Images`, aquí se guardan todos los gif creados que representan una seña. Se crea una lista con todos los elementos existentes en el directorio `Images` y luego se buscan uno a uno los elementos de la lista de lemas dentro de la lista de `Images`, si el lema existe, es decir la seña para esa palabra está en colección se crea un objeto clip de película de moviepy y se anexa a una lista de clips vacía.

```
clip.append((VideoFileClip(input_path+'Images/'+lema)))
```

Si no existe el lema dentro de la colección se procede a crear dinámicamente una imagen de fondo negro y fuentes blancas con el lema en el centro con la biblioteca Pillow (PIL), esta imagen también se convierte a un objeto de tipo clip de película y se anexa a la lista de clips. El proceso se repite hasta terminar de iterar sobre la lista de lemas.

Entonces se pasa a concatenar los clips de película con la biblioteca moviepy y se crea un clip final que se convierte en gif.

```
composition = concatenate_videoclips(clip)
composition.write_gif('out.gif', fps=10)
```

Para mostrar el gif resultante se utiliza la biblioteca `Python.display` con la función `display`.

```
x = Imagepy(filename="out.gif", width=700)
display(x)
```

Se implementaron además dos módulos de pruebas adicionales, uno con entrada con flujo de texto por teclado y otro lectura de un archivo de texto. Se puede comentar que en la versión de entrada por teclado hay una versión que cuenta con comparaciones entre los lemas de entrada y la lista de señas disponibles con el uso de inferencia mediante la lógica difusa gracias a la biblioteca `diffib` incluida en Python y su función `SequenceMatcher`, la cual busca un parecido entre dos palabras y retorna un porcentaje de coincidencia, en la versión implementada se aceptan coincidencias del 90% ya que con

números menores se encontraron coincidencias erróneas, por ejemplo match de 85.75% entre la palabra amor y la palabra mora. Así mismo se encontraron match entre las palabras amor y amar de 85.75% también, lo que hizo que se descartara su uso en la implementación principal, pero se comparte el código para futuras mejoras. Las 3 versiones de prueba se encuentran disponibles en [24] pueden ser utilizadas siguiendo los pasos descritos en la sección 4.3.

TAMtALUm, puede volverse parte de una aplicación móvil gracias a la biblioteca Kivy [30].

Kivy es una biblioteca de código abierto de Python, para el desarrollo rápido de aplicaciones que funciona bajo la licencia MIT, profesionalmente documentada con guías para empezar el desarrollo de cualquier app desde cero.

Con la sugerencia del uso de Kivy para el desarrollo de apps que incluyan a TAMtALUm se busca evitar problemas de integración, unificando el lenguaje en el que se va a escribir todo el código.

La implementación de la app debe constar de un esqueleto base de aplicación móvil que utilice una biblioteca de reconocimiento de voz cuya salida sea una cadena de texto, e decir que cumpla la misma tarea que SpeechRecognition de Google, aunque la misma podría utilizarse las consultas estarían restringidas a 50 por día así que se debe buscar una que sea compatible y libre.

TAMtALUm.py puede ser llamado desde cualquier programa escrito en Python 3, pero su uso está condicionado a la previa configuración de Freeling en el sistema anfitrión como parte de la descarga de la aplicación móvil final, volviéndola algo pesada si se adiciona además el costo en memoria que puede traer la implementación basada en gif que se plantea en este proyecto como demo.

La salida del traductor debe realizar las mismas tareas que el demo implementado en la versión de computadora, buscar dentro de la colección y concatenar gif coincidentes o crear imágenes.

CAMINAR	CERRAR	CLASIFICAR	COMPRAR	COMPRENDER
CONSEGUIR	COPIAR	CREAR	CREER	CUIDAR
DAR	DEBER	DECIR	DESCANSAR	DOBLAR
ENSEÑAR	ENTENDER	ENTRAR	ENVIAR	EQUIVOCAR
ESCUCHAR	EMPEZAR	GANAR	INFORMAR	LLEVAR
ESCRIBIR	ESPERAR	ESTUDIAR	EXPULSAR	DOBLAR
GUARDAR	GUSTAR	HABER	HACER	HABLAR
INTERCAMBIAR	INVITAR	LEER	LEVANTAR	LLEGAR
MEDIR	MOLESTAR	MOSTRAR	ODIAR	OIR
OLVIDAR	PAGAR	PARTIR	PEDIR	PEGAR
PENSAR	PEDIR	SABER	TRAER	ALEGRE
PREGUNTAR	PREPARAR	PREOCUPAR	RECORDAR	RESPETAR
SENTAR	SEPARAR	SUSPENDER	TOMAR	TRABAJAR
TRABAJAR	VENDER	VER	VIVIR	ACEPTABLE
ALTO	BONITO	BRILLANTE	BUENO	CALCULADOR
CANSADO	CARO	CERRADO	CLARO	ABIERTO
CUADRADO	DIFÍCIL	ESTUDIOSO	FÁCIL	FALSO
CORTO	FUERTE	LENTO	PESADO	TRISTE
GRANDE	HONDO	HUMEDO	IGUAL	INTELIGENTE
MEJOR	NUEVO	OSCURO	OTRO	PEQUEÑO
POBRE	RÁPIDO	SECO	SABADO	SOLO
ÚLTIMO	ME	TE	NOS	LUNES
MARTES	MIÉRCOLES	JUEVES	VIERNES	DOMINGO
TRISTE	ÚLTIMO	ME	TE	NOS
BORRAR	CAMBIAR	CONFUNDIR	CHOCAR	EMPEZAR
LUNES				

Tabla 4.2: Catálogo de señas extras (continuación)

Entrada	Salida
El perro persigue al gato	PERRO PERSEGUIR GATO
El gato es perseguido por el gato	GATO SERPRESENTE PERSEGUIR POR GATO
El perro ladró toda la noche	PERRO LADRAR SENA_PASADO TODO NOCHE
No ha parado de llover	NO HABER PARAR DE LLOVER

Tabla 4.3: Algunas pruebas unitarias sobre TAMtALUm con la lista de elicitación

Entrada	Salida
Un programa que corre en consola puede ser detenido por el usuario administrador con control c	UNO PROGRAMA QUE CORRER EN CONSOLA PODER SENA_SER_PRESENTE DETENER POR USUARIO ADMINISTRADOR CON CONTROL C
El resultado obtenido para la derivada parcial de x con respecto a y es 850.	RESULTADO OBTENER PARA DERIVADA PARCIAL DE X CON RESPECTO A Y SENA_SER_PRESENTE 850
En primer lugar, como restricción, tenemos que x es diferente de 0. La idea principal es poder quitar los denominadores de esta ecuación. Fijándonos en los coeficientes de cada denominador, tenemos que el mínimo común múltiplo de tres, dos y diez es treinta.	EN 1 LUGAR COMO RESTRICCION NOSOTROS TENER QUE X SENA_SER_PRESENTE DIFERENTE DE 0 IDEA PRINCIPAL SENA_SER_PRESENTE PODER QUITAR DENOMINADOR MUCHO DE ESTE ECUACION FIJAR NOS EN COEFICIENTE DE CADA DENOMINADOR TENER QUE MINIMO COMUN MULTIPLO DE 3 2 Y 10 SENA_SER_PRESENTE 30.

Tabla 4.4: Muestra de algunas de las pruebas unitarias realizadas sobre TAMtALUm sobre vocabulario EISULA

Entrada	Salida
no soy colombiana, soy venezolana	YO NO SENA_SER_PRESENTE COLOMBIANO YO SENA_SER_PRESENTE VENEZOLANO
las hormigas comen en Francia con elegancia	HORMIGA MUCHO COMER EN FRANCIA CON ELEGANCIA
quiero tomarme una copa de vino contigo	QUERER TOMAR ME UNO COPA DE VINO CONTIGO
no me quiero tomar unas copas de vino contigo	NO ME QUERER TOMAR UNO COPA DE VINO CONTIGO

Tabla 4.5: Muestra de algunas de las pruebas unitarias realizadas sobre TAMtALUm sobre las palabras con flexiones

Capítulo 5

Conclusiones y recomendaciones

Muchas veces los problemas a simple vista parecen más fáciles de solucionar de lo que en verdad son, las complejidades van surgiendo a medida que se va trabajando, pero cuando se realizan las investigaciones adecuadas muchas de las complejidades pueden ser previstas y tratadas a tiempo. En este caso no se lograron cumplir en totalidad los objetivos propuestos, la implementación de el traductor en un dispositivo android no fue realizada principalmente por falta de tiempo y por mala especificación de los objetivos a realizar en el tiempo propuesto, se deseó abarcar más de lo que se podía en un proyecto muy ambicioso, se requirió invertir mucho esfuerzo en definición de reglas que se esperaba estuvieran hechas, además en investigaciones que se suponía estaban correctas como las especificadas en [8] sobre el orden de los predicados en oraciones que indican posesión, estos errores se dan porque no se tomó en cuenta que el LSV evoluciona como cualquier otro lenguaje y esta evolución no fue documentada por falta de inversión en proyectos orientados al desarrollo de la LSV. Sin embargo, la creación de una app que incluya a TAMtALUm no es una idea que debe descartarse, es una excelente propuesta de proyecto de grado para cualquier estudiante de ingeniería de sistemas o de computación de cualquiera de nuestras universidades a nivel nacional, porque el código utilizado está disponible en [24] y es libre, se deben realizar los ajustes necesarios para la utilización de Kivy como librería para desarrollo android, también se puede estudiar la posibilidad de implementar TAMtALUm dentro de Android Studio, así que la construcción de una app que utilice TAMtALUm queda pendiente pero no

olvidada.

5.1 Conclusiones

Es importante resaltar que somos tan buenos desarrollando cualquier proyecto y resolviendo problemas como el manejo que tenemos sobre las herramientas que utilizamos, por eso nunca es una pérdida invertir tiempo en la formación previa y en el correcto aprendizaje esto nos llevara por un camino conocido y no uno rocoso en el que se puede estancar cualquier proyecto de desarrollo.

La documentación de las herramientas en la mayoría de los casos es la mejor aliada en un proyecto de desarrollo. Se requiere imaginación, pero es igual de importante empezar a hacer, sabiendo hacia donde va encaminado el proyecto, en este aspecto destaca que la internet es un medio que muchas veces es subestimado, tenemos la capacidad de conectarnos con cualquier persona en cualquier parte del mundo, claro, si la persona posee acceso a internet y gracias a esto se tuvo la posibilidad de contactar al profesor Arturo Curiel de la UNAM, el cual debido a su experiencia supo orientar la dirección que debía tomar el proyecto mucho antes de empezar a investigar, haciendo que un proyecto que se veía imposible terminar en un lapso de 6 meses un hecho. También se debe resaltar la importancia de contar con ayuda de un representante de la comunidad sorda, en este caso se tuvo el apoyo del joven José Gregorio Suarez y de su mamá que a la vez también es una de sus intérpretes en el proceso de elicitación y documentación de todas las señas de los catálogos 4.1 y 4.2, el proceso investigativo no hubiese sido posible.

Por por otra parte cabe resaltar que Freeling, es una herramienta que brinda muchas alternativas para la resolución de problemas referentes al PLN y que se recomienda utilizar cuando se trata de procesamiento de la lengua española, es una biblioteca que cuenta con excelente soporte técnico de la mano de sus creadores gracias a la plataforma Github, así como también cuenta con algoritmos de reconocimiento de fechas, números, lenguaje, ortografía y varias tareas más que serían de mucha utilidad a futuros proyectos de desarrollo de la ULA.

El enfoque sistémico del proyecto fue parte fundamental de la metodología, dividir

todo el gran problema a solucionar en tareas más pequeñas y ver al sistema macro, traductor de voz a LSV funcionando como un todo compuesto de partes que pueden ser analizadas y resueltas individualmente y luego juntar estas partes para ver como funciona el sistema general que se propone en principio, hace concluir que como Ingeniero en Sistemas no importa el problema que se enfrente, si se tiene en enfoque adecuado y el manejo de las herramientas se puede resolver.

Durante 10 semestres, los estudiantes reciben educación en distintos ámbitos de la ingeniería de sistemas. El aprendizaje ganado durante este periodo ha sido clave para afrontar cada uno de los contratiempos que se han presentado en el desarrollo de este proyecto. De las mayores enseñanzas obtenidas en la estancia en la ULA es la capacidad de resolver problemas, buscando soluciones excelentes e inmediatas, a bajo costo monetario y con la utilización de pocos recursos, es muy importante además resaltar que el enfoque que se da a la enseñanza de la programación es el correcto. No tiene sentido aprender todos los aspectos formales de un lenguaje de programación, el verdadero programador entiende la lógica y puede aplicarla con la sintaxis de cualquier lenguaje.

Con respecto a la LSV es importante fomentar la inversión en el desarrollo de una gramática actual, es este proyecto se da un primer paso importante pero que está lejos de ser una gramática formal, se requiere de la intervención de expertos en el área de lingüística que normalicen las ideas planteadas con investigaciones actuales, basadas en la nueva LSV. Además es necesario que se documenten formalmente las reuniones que realizan los diferentes clusters de Sordos a nivel estatal y nacional, para que así la evolución de la LSV y las nuevas señas creadas tengan una verdadera formalidad de uso.

Finalmente cabe destacar que no hay mayor satisfacción que hacer algo que es de utilidad para el mundo, aunque las adversidades de la sociedad en la que nos desenvolvemos no ayuden, siempre se debe tener como motivación el porque se empieza hacer algo, esas razones nos dan valor para afrontar las dificultades en cualquier ámbito de la vida.

5.2 Recomendaciones a futuro

La primera recomendación es seguir con este proyecto, TAMtALUm es solo una primera iteración, un proyecto en el que, seguramente con el uso continúe irá mostrando deficiencias, por lo tanto, es necesario que se invierta tiempo en el estudio de mejoras, por ejemplo, Freeling ofrece funciones y clases para el reconocimiento del idioma y también reconocimiento de discursos, el cual puede ser utilizado para una futura iteración. Además queda pendiente la construcción del módulo de reconocimiento de voz y el animador, como propuestas de grado factibles.

Como posibles mejoras a TAMtALUm se plantea el uso de algoritmos de reconocimiento de emociones, lo que ayudaría a hacer una verdadera traducción de español a LSV, la gestualidad es tan importante como la sintaxis, un traductor de este tipo que no reconoce expresiones que vienen cargadas de está destinado a equivocarse en la interpretación, se puede decir que TAMtALUm nunca va a estar listo, siempre se va a encontrar en proceso de desarrollo con una versión funcional vigente, por lo tanto la creación de un equipo de trabajo que siga iterando sobre el traductor en si es necesaria.

Un proyecto con tanta utilidad puede ser también un método ingreso de capital. La financiación de investigaciones y desarrollos como el presente es posible si se presenta ante los entes competentes y se piden los recursos necesarios para su crecimiento.

Además se puede plantear la optimización de las reglas existentes, de la mano de lingüistas interesados en la LSV, para aplicar algoritmos de aprendizaje automático que ampliarían el lenguaje traducible, este es un paso para volver a TAMtALUm un traductor completo y no solo un traductor para caso de estudio.

En la escuela de ingeniería de sistemas se debe mostrar más interés por el desarrollo de proyectos de servicio comunitario y de grado que estén orientados a solventar problemas actuales y reales. En los que se busque solventar de alguna manera la deuda que existe hacia nuestra universidad y hacia nuestro país por recibir educación gratuita.

Es necesario además que con el apoyo de la ULA se introduzcan al sistema educativo primario y secundario el estudio de la LSV en paralelo con el inglés como lengua complementaria, estas acciones son verdaderas herramientas de inclusión, necesitamos

abrir los salones clases a todos los venezolanos por igual, sin discriminar a ninguna persona por su condición física.

www.bdigital.ula.ve

C.C. Reconocimiento

Bibliografía

- [1] Lluís Padró. Analizadores multilingües en freeling. *Linguamatica*, 3(2):13–20, December 2011.
- [2] Ana María Morales García. La lengua de señas en la vida de los sordos o el derecho de apalabrar su realidad. Disponible en <http://www.cultura-sorda.org/lengua-de-senas-en-la-vida-de-los-sordos/>, 2015.
- [3] Lourdes Pietrosevoli. Primer seminario de linguistica de la lengua de señas venezolana (l.s.v). In *I SEMINARIO DE LINGUISTICA DE LA LENGUA DE SEÑAS VENEZOLANA (LSV)*, Mérida, Venezuela, 1989.
- [4] INE. Boletín de mesa de discapacidad. Disponible en <http://www.ine.gov.ve/documentos/SEN/menuSEN/pdf/subcomitedemografica/Discapacidad/PoblacionDiscapacidadVzla.pdf>, 2014.
- [5] Luis Terán. Software educativo para el uso del alfabeto manual como herramienta de enseñanza del español como segunda lengua en niños sordos del subsistema de educacion inicial. Technical report, Universidad Católica Andrés Bello, 2008.
- [6] Benito Estrada Aranda. La vulneración de los derechos humanos de las personas sordas en México. Disponible en <http://www.cultura-sorda.org/la-vulneracion-de-los-derechos-humanos-de-las-personas-sordas-en-mexico/>, 2008.
- [7] Asamblea Nacional Venezuela. Constitución de la república bolivariana de Venezuela. Disponible en http://www.asambleanacional.gob.ve/documentos_archivos/constitucion-nacional-7.pdf, 1999.

- [8] María Eugenia Dominguez Mujica. Los verbos de la lsv fundamentos para comprender la morfología verbal de una lengua de señas. Tesis de maestría, Universidad de los Andes, Biblioteca Tulio Febres Cordero, Universidad de Los Andes, Mérida, Venezuela, digitalizado por SERVIULA, 1996.
- [9] Rafael Alejandro Oviedo Palomares. El discurso narrativo de la lengua de señas venezolana. dos ensayos. Trabajo presentado como requisito parcial para ascender a la categoría de agregado, Universidad de los Andes, Disponible en línea, digitalizado por SERVIULA, 1995.
- [10] Prodeaf. Prodeaf traductor. Disponible en <http://www.prodeaf.net/>, 2016.
- [11] APP ¡HÁBLAME! Háblame app. Disponible en http://www.hablameapp.co/#feature_1, 2014.
- [12] Pichardo et al. Sistema de traducción directa de español a lsm con reglas marcadas. *Research in Computing Science*, 1(115):29–41, Mayo 2016.
- [13] Mario Chacón-Rivas Johan Serrato-Romero. Traductor de documentos y texto a lengua de señas. Conference papel, TEC Digital, Instituto Tecnológico de Costa Rica, Costa Rica, En línea: <https://www.researchgate.net/publication/299545941>, 2015.
- [14] Matías Laíno. Integración de herramientas para procesamiento de lenguaje natural. Proyecto de grado, Instituto de Computación - Facultad de Ingeniería. Universidad de la República, Montevideo, Uruguay, En línea, 2015.
- [15] Noam Chomsky. *Syntactic structures*. Walter de Gruyter, 2002.
- [16] Jurafsky & Martin. *Speech and Language Processing*. Prentice Hall, 1999.
- [17] M Hernández. J Gómez. Aplicaciones del procesamiento del lenguaje natural. Artículo de revista, Escuela Politécnica Nacional, Facultad de Ingeniería en Sistemas, Quito, Ecuador. Universidad de Alicante, Departamento de Lenguajes y Sistemas Informáticos, Alicante, España, Revista Politécnica Julio 2013, Vol. 32, No. 1, Páginas: 87 , 96, Julio 2013.

- [18] Lluís Padró, Miquel Collado, Samuel Reese, Marina Lloberes, and Irene Castellón. Freeling 2.1: Five years of open-source language processing tools. In *Proceedings of 7th Language Resources and Evaluation Conference (LREC'10)*, La Valletta, Malta, May 2010.
- [19] Jordi Atserias, Bernardino Casas, Elisabet Comelles, Meritxell González, Lluís Padró, and Muntsa Padró. Freeling 1.3: Syntactic and semantic services in an open-source nlp library. In *Proceedings of the fifth international conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy, May 2006. ELRA.
- [20] Xavier Carreras, Isaac Chao, Lluís Padró, and Muntsa Padró. Freeling: An open-source suite of language analyzers. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*, 2004.
- [21] Lluís Padró and Evgeny Stanilovsky. Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey, May 2012. ELRA.
- [22] Luis Padró. Freeling 4.0 user manual. Disponible en <https://legacy.gitbook.com/book/talp-upc/freeling-4-0-user-manual/details>, 2018.
- [23] Lourdes G. Pietrosevoli. La lengua de senas analisis linguistico. Trabajo realizado como credencial para optar a la categoría de profesor titular, Facultad de Humanidades y Educacion Escuela de Letras, Departamento de Linguistica, Universidad de Los Andes, Merida, Venezuela, Biblioteca Tulio Febres Cordero, Universidad de Los Andes, Mérida, 1991.
- [24] María Blanco. Repositorio tamtalum. Disponible en https://github.com/maferblancor/LSV_Translator, 2018.
- [25] Yolanda Pérez. La norma en la lengua de señas venezolana. *Sapiens, Revista Universitaria de Investigación*, 8(2):105–121, Diciembre 2007.
- [26] Arturo Curiel. Reconocimiento automático de la lengua de señas. Tesis de doctorado, Universite Paul Sabatier, Toulouse III, RIT, Disponible en linea en:

http://www.mufm.fr/sites/mufm.univtoulouse.fr/files/arturo_tlacaelel_curiel_diaz.pdf, SF.

- [27] Lourdes Pietrosevoli et al. Marcadores discursivos en hablantes sanos y afásicos: El caso especial de y. Technical Report 2, Julio 2006.
- [28] Luis Padró. Freeling 4.1 user manual. Disponible en <https://talp-upc.gitbooks.io/freeling-4-1-user-manual/content/basics.html>, 2018.
- [29] Jupyter Steering Council. Proyecto jupyter. Disponible en <http://jupyter.org/about>, 2018.
- [30] Kivy community. Kivy. Disponible en <https://kivy.org/#home>, 2018.

www.bdigital.ula.ve