Forma y Función, vol. 36, núm. 2, 2023, pp. 1-24.

UnderRL Tagger: un etiquetador gramatical para lenguas infrasoportadas tecnológicamente y lenguas minoritarias.

Pemberty Tamayo, José Luis, Molina Mejia, Jorge Mauricio y Vallejo Zapata, Víctor Julián.

Cita:

Pemberty Tamayo, José Luis, Molina Mejia, Jorge Mauricio y Vallejo Zapata, Víctor Julián (2023). *UnderRL Tagger: un etiquetador gramatical para lenguas infrasoportadas tecnológicamente y lenguas minoritarias. Forma y Función, 36 (2), 1-24.*

Dirección estable: https://www.aacademica.org/jorge.mauricio.molina.mejia/74

ARK: https://n2t.net/ark:/13683/pqc6/4Zr

10.15446/fyf.v36n2.101984
Artículos

UNDERRL TAGGER: UN ETIQUETADOR GRAMATICAL PARA LENGUAS INFRASOPORTADAS TECNOLÓGICAMENTE Y LENGUAS MINORITARIAS*

UNDERRL TAGGER: A GRAMMAR TAGGER FOR TECHNOLOGICALLY UNDER-SUPPORTED AND MINORITY LANGUAGES

José Luis Pemberty Tamayo[1]
Jorge Mauricio Molina Mejía[2]
Víctor Julián Vallejo Zapata[3]

Cómo citar este artículo:
Pemberty Tamayo, J. L., Molina Mejía, J. M., & Vallejo Zapata, V. J. (2023). UnderRL Tagger: un etiquetador gramatical para lenguas infrasoportadas tecnológicamente y lenguas minoritarias. *Forma y Función, 36*(2). https://doi.org/10.15446/fyf.v36n2.101984

Recibido: 2022-04-04, aceptado: 2023-02-24

* Este artículo está derivado del desarrollo del software UnderRL Tagger (Pemberty Tamayo et al., 2020) en el marco del trabajo de grado de Pemberty Tamayo (2020) y con el apoyo del semillero de investigación «Corpus Ex Machina» de la Universidad de Antioquia.

1 ORCID https://orcid.org/0000-0001-9498-874X Universidad de Antioquia, Medellín, Colombia. jose.pemberty@udea.edu.co

2 ORCID https://orcid.org/0000-0002-1430-6364 Universidad de Antioquia, Medellín, Colombia. jorge.molina@udea.edu.co

3 ORCID https://orcid.org/0000-0002-5965-4880 Universidad de Antioquia, Medellín, Colombia. victor.vallejo@udea.edu.co

Resumen

En este artículo se presenta UnderRL Tagger, un programa informático de acceso libre diseñado para el etiquetado morfosintáctico (*POS tagging*) en lenguas que no cuentan con etiquetadores automáticos. El programa busca facilitar el trabajo con corpus en estas lenguas infrasoportadas tecnológicamente y en las lenguas minoritarias, aportando así a los procesos de revitalización basada en investigación descriptiva y herramientas computacionales. UnderRL Tagger permite que el proceso manual de etiquetado se convierta poco a poco en automático gracias a un sistema que permite recordar y reutilizar las etiquetas, manejar grandes cantidades de textos y generar archivos de salida en formato XML con etiquetas basadas en el sistema estandarizado EAGLES. Este artículo muestra el proceso de modelado y elaboración del sistema, sus diferentes funcionalidades y las perspectivas de trabajos posteriores.

Palabras clave: *etiquetado morfosintáctico; lenguas infrasoportadas tecnológicamente; lenguas minoritarias; corpus textuales; procesamiento del lenguaje natural.*

Abstract

This paper presents UnderRL Tagger, a freely available software program designed for morphosyntactic tagging (POS tagging) in languages that do not have automatic taggers. The program aims to facilitate working with corpora in these technologically under-supported languages and in minority languages, thus contributing to revitalization processes based on descriptive research and computational tools. UnderRL Tagger allows the manual tagging process to gradually become automatic thanks to a system that allows remembering and reusing tags, handling large amounts of text and generating output files in XML format with tags based on the standardized EAGLES system. This article shows the process of modeling and development of the system, its different functionalities and the prospects for further work.

Keywords: *morphosyntactic tagging; technologically under-supported languages; minority languages; text corpora; natural language processing.*

1. Introducción

La investigación lingüística de las lenguas minoritarias, que abarca las actividades de descripción, documentación y etnoeducación, es un trabajo inacabado y en constante renovación que presenta distintos retos: qué, cómo, para qué y para quién documentar (Ramírez-Cruz & Chaparro, 2021). Dentro de este panorama, la articulación entre lingüística de corpus y lingüística computacional ha resultado sumamente relevante, pues en su seno han surgido avances tecnológicos que permiten registrar y procesar grandes cantidades de muestras de uso, sistematizar las reglas gramaticales y léxicas de su funcionamiento, y crear bases de datos para su acceso y empleo en distintas aplicaciones (Woodbury, 2014).

Sin embargo, el desarrollo de herramientas tecnológicas lingüísticas no está a la par de las necesidades urgentes de todas las lenguas minoritarias. Como ocurre respecto de la desigualdad en la vitalidad de las lenguas, donde, por ejemplo, el 37.7% de hablantes nativos está distribuido en ocho lenguas de un total de 7111 lenguas vivas (Eberhard et al., 2019), la mayoría de los investigadores y programadores se orienta a aquellos idiomas predominantes en las relaciones comerciales y académicas como el inglés, el mandarín o el español (Ostler, 2014; Camacho & Zevallos, 2020; Cunliffe et al., 2022).

Esta circunstancia deriva en poco desarrollo de soportes tecnológicos para la mayoría de las lenguas, es decir, en la carencia de recursos informáticos para la descripción y documentación lingüísticas, que incluye los insumos de datos para la construcción de corpus, las herramientas para su procesamiento y el diseño de aplicaciones con distintas finalidades (Somers, 2003; Maxwell & Hughes, 2006).

Denominamos las lenguas en esta situación como *lenguas infrasoportadas tecnológicamente* (LIT)[1]. Para que una lengua pueda incluirse dentro de esta categoría debe presentar una o varias de las siguientes características (Somers, 2003; Besacier et al., 2014):

- Carencia de un único sistema de escritura o una ortografía estable.
- Presencia limitada en la internet, tanto de páginas sobre la lengua y su cultura, como de páginas publicadas en dicha lengua.
- Carencia de recursos electrónicos para el procesamiento del habla y la lengua, como corpus monolingües, diccionarios bilingües electrónicos, corpus orales transcritos, diccionarios de pronunciación, colecciones terminológicas, herramientas de reconocimiento de voz y de caracteres.
- Carencia de herramientas de exploración, codificación y manipulación de corpus, como herramientas de anotación, etiquetado y lematización.

- Carencia de *hardware* específico para el procesamiento de la lengua, como teclados y sistemas de entrada ajustados al sistema.
- Poca representación académica en la figura de lingüistas, científicos sociales o educadores expertos en dicha lengua.

Nótese que los tipos de limitaciones planteados abarcan tanto la investigación lingüística (descripción, documentación y etnoeducación) como los espacios de interacción tecnológica y comunicativa para los usuarios de las lenguas en juego. De esta manera, encontramos que las LIT plantean un diagnóstico que atraviesa las distintas dimensiones de investigación y acción lingüística, en consonancia con las propuestas que abogan por abordajes complejos de la diversidad lingüística articulando lo académico, lo social y lo político (Unesco, 2014; Ramírez-Cruz & Chaparro, 2021).

El concepto de *infrasoporte tecnológico* llama la atención sobre nuevas formas de minorización y desigualdad ante la actual revolución de las Tecnologías del Lenguaje Humano, es decir, la última generación de Tecnologías de la Información y la Comunicación (TIC) consistente en sistemas que permiten a los usuarios interactuar en su propio idioma, donde «los dispositivos podrán encontrar automáticamente las noticias y la información más importantes de todo el mundo simplemente utilizando comandos de voz fáciles de usar». Así, «la tecnología del lenguaje podrá traducir automáticamente o ayudar a los intérpretes, resumir conversaciones y documentos, además de apoyar a los usuarios en tareas de aprendizaje» (Camacho & Zevallos, 2020, p. 188).

De esta manera, las LIT afrontan distintos retos en lo que toca a las diversas aplicaciones tecnológicas que se vienen desarrollando en las últimas décadas, desde las herramientas para la sistematización de corpus hasta las interfaces de usuario en sistemas de uso cotidiano. En ese sentido, la revitalización —esto es, los procesos tendientes al mantenimiento y supervivencia social de la diversidad lingüística— demanda el apoyo en tecnologías tanto para los investigadores como para los hablantes (Ostler, 2014; Cunliffe et al., 2022).

Finalmente, debemos resaltar que las LIT no equivalen, necesariamente, a las lenguas minoritarias o a las variedades dialectales marginales. Si bien es posible, como ocurre en el caso colombiano, que las lenguas minoritarias —aquellas que son usadas por un grupo social numéricamente menor al resto de la población y que sufren procesos de exclusión o afectación de sus derechos básicos (Núñez, 2013)— se encuentren también en situación de infrasoporte tecnológico, también se pueden encontrar lenguas mayoritarias infrasoportadas tecnológicamente, como el hindi (Somers, 2003), o lenguas minoritarias bien soportadas tecnológicamente, como el catalán (Besacier et al., 2014).

Siguiendo a Cunliffe et al. (2022), podemos sintetizar la relación entre infrasoporte tecnológico, minorización lingüística y revitalización como sigue:

> La tecnología del lenguaje se está volviendo cada vez más importante en una variedad de dominios de aplicaciones que se han convertido en un lugar común en las lenguas mayoritarias y con buenos recursos. Sin embargo, existe el peligro de que las lenguas minoritarias o infrasoportadas tecnológicamente sean empujadas cada vez más a los márgenes tecnológicos. Las lenguas infrasoportadas tecnológicamente enfrentan retos significativos a la hora de ofrecer los recursos lingüísticos subyacentes necesarios para sustentar el desarrollo de dichas aplicaciones. (p. 1)

Como presentamos en la sección 2 (Fundamentos), si bien existen etiquetadores gramaticales semiautomáticos y automáticos, todas estas herramientas tienen en común que han sido desarrolladas con la finalidad de sistematizar lenguas mayoritarias o que cuentan con grandes recursos lingüísticos (inglés, español, francés, alemán, chino mandarín, etc.). No encontramos, en nuestro rastreo, herramientas de este tipo para las LIT.

Teniendo en cuenta este panorama, el objetivo de este artículo es presentar el desarrollo de UnderRL Tagger, una herramienta para el etiquetado gramatical orientado a las LIT. Confiamos que, en el contexto de Colombia y otros países donde las lenguas minoritarias sufran también de infrasoporte tecnológico, este programa sirva como apoyo a las actividades de descripción lingüística en el nivel gramatical.

2. Fundamentos

Como se mencionó en el apartado anterior, la lingüística computacional y la lingüística de corpus han establecido un campo fructífero de colaboración en el desarrollo de herramientas informáticas. Dentro de este, resaltan las plataformas y aplicaciones para el etiquetado de corpus, que permiten etiquetar de manera automática grandes cantidades de textos en diferentes lenguas. Algunas herramientas de libre acceso conocidas en el medio son TreeTagger[2] (Schmid, 1994) y TagAnt[3] (Anthony, 2022), que pueden etiquetar y lematizar diversas lenguas a nivel gramatical, proceso conocido en inglés como *Part of Speech* (POS); Weisser (2018). Además de aplicaciones, se han desarrollado librerías orientadas al procesamiento del lenguaje natural como FreeLing[4] (Padró et al., 2010) y Stanford Parser[5] (Schuster & Manning, 2016), las cuales permiten etiquetar en diferentes niveles de análisis del lenguaje, por ejemplo el *parsing* (generación de árboles sintácticos a partir de la gramática de dependencias, o de constituyentes inmediatos), el reconocimiento de cadenas correferenciales (anáforas y catáforas), la elaboración

de grafos semánticos o el análisis de entidades nombradas. Todas estas herramientas sirven para la sistematización de lenguas mayoritarias, pues incluyen etiquetas morfosintácticas predeterminadas a dichos idiomas, sin posibilidad de modificarlas para adecuarlas a otras lenguas.

Sin embargo, esto no implica que sea imposible desarrollar herramientas para el etiquetado morfosintáctico de las lenguas minoritarias. Para ello se cuenta con los parámetros establecidos por el sistema EAGLES (Leech & Wilson, 1996), que sirve de estándar para las etiquetas de las diferentes lenguas. Se trata de una serie de convenciones adoptadas por distintos grupos en el trabajo de corpus, propuesto por el Expert Advisory Group on Language Ingineering Standards y que consta de una serie de normativas para la asignación de códigos a los posibles accidentes gramaticales que ocurran en las lenguas. Como ilustra la Tabla 1 respecto de los códigos para el caso de los adjetivos, nuestro programa se acoge a esta estandarización para la definición de los algoritmos. La finalidad de esta elección es que se pueda contar con un sistema de etiquetas que se encuentren reconocidas y compartidas por lingüistas de diferentes latitudes.

Tabla 1. Valores y caracteres de adjetivo incluidos por defecto en UnderRL Tagger

Categoría gramatical	Característica	Posibles valores		
A (Adjective)	Type	O (Ordinal)	Q (Qualificative)	P (Possessive)
	Degree	S (Superlative)	V (Evaluative)	
	Genere	F (Feminine)	M (Masculine)	C (Common)
	Number	S (Singular)	P (Plural)	I (Invariable)
	Possessor-person	1	2	3
	Possessor-number	S (Singular)	P (Plural)	I (Invariable)

Fuente: Pemberty Tamayo (2020, p. 49).

Respecto del tratamiento computacional de las LIT, encontramos enfoques tan variados como el etiquetado de lenguas concretas, sean mayoritarias como el árabe o el somalí (El-Haj et al., 2015; Mohammed, 2020) o minoritarias como el irlandés (McCrae & Doyle, 2019), el desarrollo de herramientas de automatización terminológica (McCrae et al., 2021), el reconocimiento del habla (Besacier et al., 2014) o la recolección de corpus por medio de la obtención de textos directamente desde la web (Sene-Mongaba, 2015). En el contexto colombiano, resaltamos, en primer lugar, la investigación adelantada por la Universidad del Cauca[6] para el nasa yuwe y el nam trik que incluye el desarrollo de sistemas de reconocimiento de habla, corpus gramaticales y materiales pedagógi-

cos (Rojas Curieux, 2017); también encontramos otras propuestas para otras lenguas minoritarias, como el diccionario electrónico sáliba-español (Dueñas & Gómez, 2015), el proyecto de diccionario bilingüe para lenguas minoritarias y en peligro a través de Mediawiki (Dueñas & Gómez, 2016), la digitalización y desarrollo tecnológico del Atlas Lingüístico-Etnográfico de Colombia (Bonilla et al., 2020), la recolección de un corpus oral del criollo sanandresano (Sanmiguel Ardila et al., 2021) o el desarrollo de sistemas de reconocimiento de fonemas para el tanikuma (Montenegro & Guaquetá, 2020). Estos antecedentes comparten con UnderRL Tagger su preocupación por las LIT y las lenguas minoritarias, aunque se diferencian de este en que no se ocupan propiamente de la asistencia automatizada en el etiquetado manual de corpus y sus enfoques son, en gran parte de los casos, monolingües, con excepción, cómo es posible apreciar, de los trabajos propuestos por los colegas del Instituto Caro y Cuervo, en los cuales se apuesta por diccionarios bilingües (Dueñas & Gómez, 2015, 2016) y entornos tecnológicos plurilingües (Bonilla et al., 2020).

Además de esto, encontramos que el Summer Institute of Linguistics (SIL) ha desarrollado dos herramientas para la gestión de corpus en diferentes lenguas, independientemente de su soporte tecnológico. Por un lado, Field Linguist's ToolBox[7] (SIL, 2019), orientado a la etiquetación gramatical y a la transcripción, y por el otro FieldWorks Language Explorer[8] (SIL, 2021), que se compone de varias aplicaciones para la gestión, anotación y sistematización de información lingüística y cultural con fines lexicográficos. Sin embargo, dada la amplitud de su campo de aplicación, pueden dificultar la tarea de obtener un corpus etiquetado en una lengua determinada, además de que carecen de una estandarización en el trabajo de la lingüística de corpus como las mencionadas etiquetas EAGLES; el uso de estas etiquetas permite que se trabaje con un estándar establecido en el mundo de la lingüística computacional actual, a diferencia de los sistemas que proponen las herramientas del SIL, cuyas etiquetas se enfocan más en información etnográfica, semántica o lexicográfica.

Volviendo sobre la relación entre soporte tecnológico y revitalización de lenguas, podemos afirmar que la disponibilidad de este tipo de herramientas en una lengua, sumada a otros factores de carácter extralingüístico, puede influir fuertemente en la decisión de un investigador de trabajar con ella: la carencia de herramientas hace que la investigación en una LIT sea menos frecuente y, por ello, la creación de las mismas herramientas sea lenta y difícil (Maxwell & Hughes, 2006; Cunliffe et al., 2022). La disponibilidad de estos elementos, a su vez, hace posible que los hablantes de la lengua puedan contar con diferentes aplicaciones de las tecnologías de la información y la comunicación, tales como la traducción automática o los diccionarios digitales. Es

por esto que subsanar la brecha de herramientas para el procesamiento computacional en estas lenguas es un interés tanto académico como social, toda vez que beneficia a las comunidades en que se usa la lengua (Pemberty Tamayo, 2020).

Partiendo de todos los temas explorados en este apartado, se evidencia la necesidad de contar con herramientas para el etiquetado de corpus en las LIT. La herramienta UnderRL Tagger (Pemberty Tamayo et al., 2020) propone un sistema que permite etiquetar grandes cantidades de textos en diferentes lenguas de forma manual, con asistencia computacional, agilizando el proceso en una proporción importante. Este proceso puede, además, producir contenido susceptible de ser reutilizado para etiquetar otros corpus en la misma lengua que sirvan como base para la creación de aplicaciones que permitan el etiquetado completamente automático (Pemberty Tamayo, 2020).

3. Marco metodológico

Presentamos en este apartado el procedimiento de construcción del programa. Teniendo en cuenta que se ha seleccionado el nivel de POS en el etiquetado como objetivo principal de la aplicación, se acogió para este el uso del sistema de etiquetas EAGLES (Leech & Wilson, 1996) que, como adelantamos previamente, permite codificar de manera corta, a través de diferentes números y letras, información como la categoría gramatical, el género, el número, entre otros. Veamos a continuación una oración de tres palabras y sus etiquetado EAGLES correspondiente:

Tabla 2. Ejemplo de etiquetas EAGLES

YO	COMPRABA	PAN
PP1CSN0	VMII3S0	NCMS000
P = Pronombre	V = Verbo	N = Nombre
P = Personal	M = Principal	C = Común
1 = Primera persona	I = Indicativo	M = Masculino
C = De género neutro o indeterminado	I = Pretérito imperfecto	S = Singular
S = Singular	3 = Tercera persona	0 = Podría tomar otro valor si fuera un nombre propio clasificable
N = Caso nominativo	S = Singular	0 = Valor también asociado a la clasificación de nombres propios
0 = Registro informal	0 = De género neutro o indeterminado	0 = Valor asociado a la presencia de rasgos apreciativos en sustantivos y adjetivos

Fuente: Pemberty Tamayo (2020, p. 21).

En la Tabla 2 puede apreciarse de qué manera se utilizan las etiquetas EAGLES para precisar la información gramatical de cada una de las palabras. Estas series de letras y números, sin embargo, deben convertirse todavía a un lenguaje de etiquetado que pueda ser procesado y analizado a nivel computacional. Para lograr esto, el programa utiliza el lenguaje de marcado extensible XML (Extensible Markup Language), que permite asignar a elementos individuales dentro una serie de características que lo definen (ver Sección 4). Así, en este lenguaje puede asignarse la etiqueta correspondiente a cada uno de los componentes del texto. Tanto las etiquetas EAGLES como el lenguaje XML corresponden a estándares que se utilizan de manera amplia en el medio del etiquetado de corpus, por lo que su utilización asegura la comprensión por parte de una amplia variedad de investigadores en el campo, así como su fácil integración con otros trabajos (Passonneau & Mani, 2014; Pemberty Tamayo, 2020; Rojas Curieux, 2017).

3.1. Descripción de la estructura del programa

UnderRL Tagger consta principalmente de una ventana con la que se puede interactuar para navegar entre los archivos del corpus, fijar las etiquetas y guardar o recuperar sesiones previas. Esta ventana está constantemente interactuando con otros archivos y carpetas que registran toda la información necesaria para que el proceso de etiquetado sea eficiente y correcto (Figura 1).

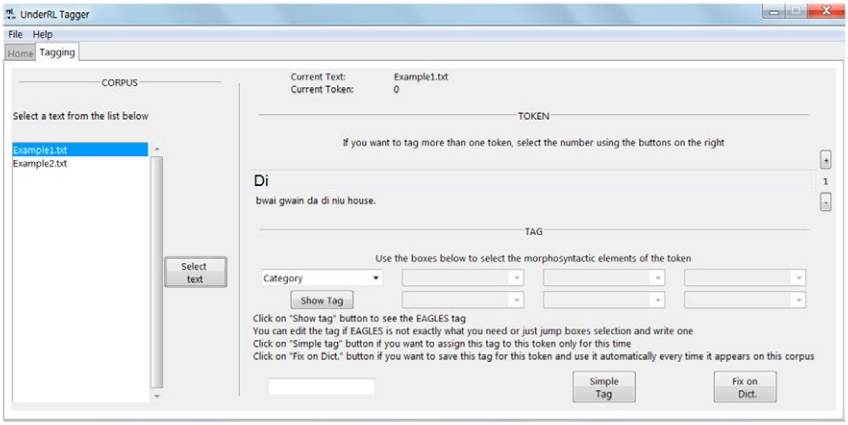

Figura 1. Ventana del programa: pantalla de etiquetado

Fuente: Pemberty Tamayo (2020, p. 32).

Esta ventana cuenta con textos, instrucciones e indicaciones en inglés. Aunque el código del programa puede modificarse para que se muestre en otros idiomas, se ha escogido el inglés para estar disponible en esta primera versión debido a que el alcance del programa no está circunscrito a los trabajos en países cuya lengua mayoritaria es el español, sino que se plantea como una herramienta que puede ser utilizada en proyectos a nivel mundial. Este aspecto, sumado al hecho de que el inglés es actualmente una lengua común entre los académicos del mundo y de que son estos los llamados a realizar los procesos de etiquetado de corpus textuales en diferentes lenguas, son las razones de esta elección en la construcción de la interfaz.

Una de las carpetas es utilizada por el sistema para guardar los datos de los diferentes *diccionarios* que se vayan creando. El *diccionario* es un archivo en el que se guardan las etiquetas que pueden ser reutilizadas en un determinado corpus, de manera que no sea necesario introducirlas de nuevo manualmente. Otra ubicación importante es la carpeta en que se guardan los archivos XML que contienen los textos ya etiquetados; esta carpeta se crea automáticamente en el mismo directorio en que se encuentran los textos originales del corpus. Además, también se cuenta con un grupo de archivos que registran en todo momento cuáles son los proyectos de etiquetado que se encuentran en ejecución y cuál es su avance, de manera que sea sencillo interrumpir la tarea de etiquetado en cualquier momento y retornar a ella posteriormente.

Partiendo de esto, en el programa pueden ingresarse todos los textos que componen el corpus, que deben encontrarse en formato de texto plano (TXT) y con codificación en formato de transformación Unicode de 8 bits UTF-8, que permite reconocer una amplia variedad de caracteres. Todos ellos deben estar guardados en una sola carpeta, cuya dirección será ingresada en la aplicación.

Una vez que se cuenta con los textos, el programa procederá a pasar por cada uno de ellos, según lo seleccione el usuario, y a realizar un proceso que consiste en separar el texto por palabras. Cuando se cuenta con las palabras separadas, en la ventana principal se le muestra al usuario cada una de ellas, permitiéndole también seleccionar más de una cuando lo necesite. Para cada palabra, el usuario podrá seleccionar, a través de varios controles, las características de la palabra que se etiquetarán, y el programa se encarga de representarlas según el modelo de EAGLES. Además de esto, un espacio en la interfaz permite la creación de etiquetas nuevas o la edición de las predeterminadas; de esta manera es posible extender las posibilidades del etiquetado según necesidades ajenas al POS. Finalmente, cuando se haya fijado una etiqueta, el usuario podrá guardarla en el archivo XML final, donde quedará dispuesta con el resto del texto, contando con su respectiva etiqueta y un identificador único.

Además de simplemente etiquetar la palabra, el usuario puede elegir guardar esa etiqueta en el *diccionario*, de manera que cada vez que aparezca nuevamente la misma palabra en el corpus será etiquetada automáticamente sin intervención del usuario; es así como este programa ayuda a automatizar en gran medida el etiquetado, pues permite que la intervención humana sea reducida a los momentos en los que sea realmente necesaria. Cada vez que el etiquetador llega a una nueva palabra, la busca en el diccionario antes de mostrarla en la pantalla, por lo que un solo texto puede pasar por fragmentos considerables antes de requerir la atención de un usuario humano.

Como consecuencia de este procedimiento, el *diccionario* puede robustecerse a medida que se avanza en el etiquetado, lo que permite que la automatización sea cada vez mayor y también contar con un archivo que puede servir para etiquetar otros textos en la misma lengua o como base para otros programas que requieran del conocimiento de estas nociones para el procesamiento del lenguaje.

Cuando un usuario percibe que no podrá automatizarse el etiquetado de una palabra, porque esta pueda presentar variaciones en sus etiquetas a lo largo del corpus, simplemente puede elegir no guardarla en el *diccionario*, por lo que cada vez que aparezca le será presentada en la ventana principal y se le permitirá elegir la etiqueta que considere adecuada para cada ocasión.

4. Análisis de los algoritmos

UnderRL Tagger es un programa escrito en lenguaje Python que puede utilizarse para el etiquetado manual de POS, poniendo los métodos de la lingüística computacional al servicio de la lingüística de corpus y permitiendo que el proceso de etiquetado de las LIT se acelere notablemente al permitir automatizar varias de sus etapas.

El flujo de procesos del programa es el siguiente (Figura 2): cuando un usuario introduce adecuadamente la dirección de una carpeta que contiene los textos de un corpus, el programa verifica la existencia de los textos y crea los archivos y carpetas necesarios (i.e., *diccionarios*) para llevar los registros que implica el proceso, tal y como se describió en la Sección 3.

Toda la información que el sistema guarda, además de los textos etiquetados en XML, se encuentra en carpetas que deben estar en el mismo directorio en que el programa se ejecuta, y se utilizan para ello archivos que también tienen el formato TXT, por lo que pueden ser leídos y modificados fácilmente, en caso de que se haya cometido algún error, por ejemplo, en la creación de una etiqueta errada en el *diccionario*.

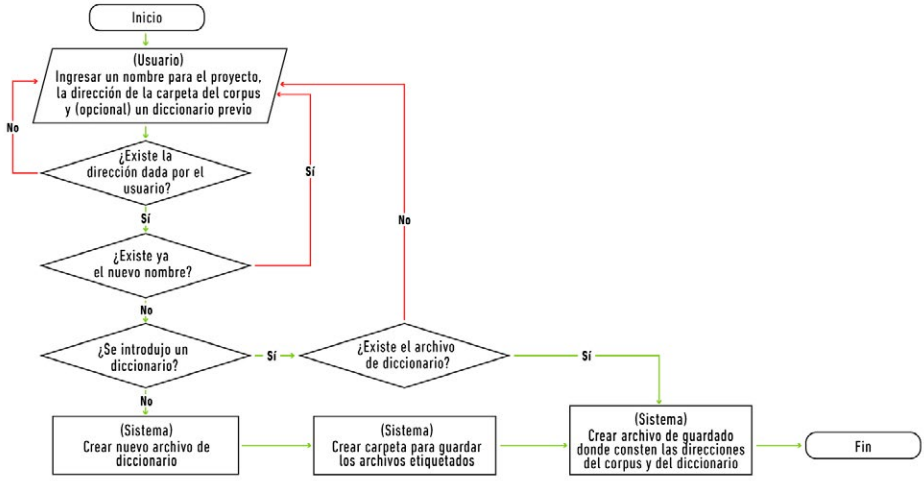

Figura 2. Diagrama de flujo: inicio de un nuevo proyecto

Fuente: adaptado de Pemberty Tamayo (2020, p. 29).

Teniendo preparados estos archivos, la herramienta pasa a etiquetar los textos. Para ejemplificar lo que sucederá en cada uno de los pasos, tomemos una oración en lengua *kriol* del archipiélago de San Andrés:

Tabla 3. Descripción de una oración en *kriol* por token, categoría gramatical y glosa

ORACIÓN A							
Token (palabra)	Di	bwai	gwain	da	di	niu	house
Categoría gramatical	Artículo	Sustantivo	Verbo	Preposición	Artículo	Adjetivo	Sustantivo
Glosa	El	niño	va	a	la	nueva	casa

Fuente: Pemberty Tamayo, (2020, p. 31).

Antes de mostrar al usuario los textos para etiquetar y las diferentes opciones, es necesario que el texto sea procesado de una manera específica. En apartados anteriores se ha hablado de que el texto se divide en palabras y a cada una de ellas se le asignan las categorías. Pues bien, es momento de precisar que el concepto adecuado en el contexto de la lingüística de corpus y computacional no es el de palabra, sino el de *token*. Siguiendo a Mikheev (2014), un *token* es una unidad lingüística mínima que puede corresponder a una palabra, un número o un signo de puntuación. Una diferencia importante entre

token y palabra es que esta sigue siendo una sola, independientemente de que aparezca varias veces en uno o en muchos textos, mientras que el primero se corresponde con una ocurrencia única, por lo que cada uno de ellos debe de diferenciarse con respecto a los otros. Al proceso de dividir un texto en los *tokens* que lo componen se le denomina *tokenizado*. Podemos ilustrar el flujo del procesamiento del texto como sigue (Figura 3):

Como puede verse en el diagrama anterior, el programa comprueba si en el sistema de archivos se encuentra información preexistente del mismo texto, para poder recuperarla y continuar en el punto en que se había dejado el trabajo, y verifica desde el primer *token* del texto si existe para él una etiqueta establecida en el *diccionario*. Asumiendo que se trata de un proyecto nuevo que no cuenta con etiquetas en su *diccionario*, el resultado de este proceso será simplemente el texto tokenizado.

También es importante considerar que los *tokens* suelen identificarse a través del espacio en blanco que media entre dos palabras; sin embargo, hay también muchas unidades que se componen de dos o más palabras separadas por espacios y que sería erróneo etiquetar como *tokens* diferentes. Estas unidades se denominan *palabras multitoken*; por ejemplo, las locuciones o algunas maneras de referirse a los números (Mikheev, 2014). Para poder etiquetar estas unidades, el sistema brinda la posibilidad de encadenar unos *tokens* con otros, pudiendo crear una unidad compuesta entre este tipo de elemento y el que le sigue.

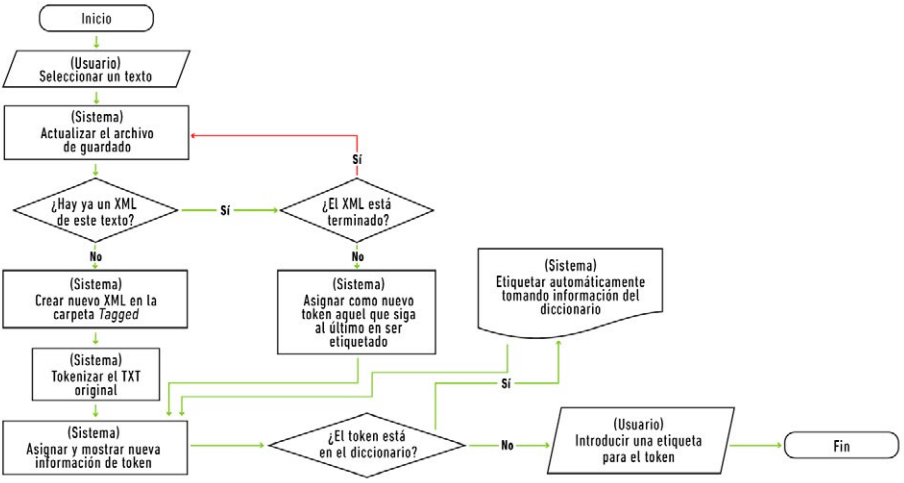

Figura 3. Diagrama de flujo: procesamiento previo de un texto seleccionado

Fuente: adaptado de Pemberty Tamayo (2020, p. 34).

El sistema realiza automáticamente todas las comprobaciones presentadas en la Figura 3, por lo que, para el usuario, apenas transcurre un instante entre la selección de un texto para su etiquetado y la posterior visualización en la ventana, en la que aparecen los primeros *tokens* y los controles para fijar las etiquetas.

Una vez realizado este proceso, podemos decir que el programa ha etiquetado una unidad, visualizado por el usuario de la siguiente manera:

Como se observa en la Figura 4, el programa presenta al usuario la palabra «Di», es decir, el primer *token* de la oración «Di bwai gwain da di niu house», junto con los demás que sirven para entender el contexto de ocurrencia. Así mismo, se habilitan para el usuario una serie de listas desplegables que le permitirán elegir entre diferentes categorías que pueden asignarse al *token* que se encuentra seleccionado. A partir de las diferentes selecciones, se creará la etiqueta en formato EAGLES, a saber: DA3CNS-.

Las diferentes posibilidades que tiene el usuario varían dependiendo de la primera selección que debe realizar, la de la categoría gramatical que desee atribuir al *token*, de la que se desprenden las demás. Por esa razón, la cantidad de información necesaria y su tipo cambian cuando se selecciona una de estas categorías.

Cuando se han seleccionado los elementos adecuados en las listas desplegables, se debe pulsar en el botón *Show Tag*, que permite visualizar, en la barra de texto de la parte inferior, la etiqueta que se creó a partir de la información ingresada y siguiendo

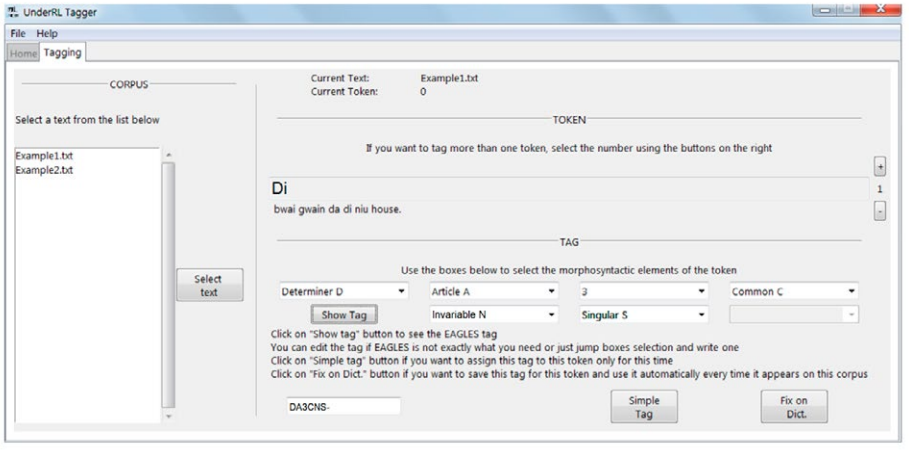

Figura 4. Ventana del programa con una unidad etiquetada con el sistema de listas desplegables

Fuente: Pemberty Tamayo (2020, p. 33).

el sistema EAGLES. Como se puede notar, en las listas desplegables las opciones se expresan con palabras de uso común en el campo de la lingüística, mientras que en la etiqueta solamente se muestra su equivalente en el sistema de etiquetado; de esta manera, no es necesario que el usuario conozca a la perfección las EAGLES para poder usarlas, pues el programa se encarga de establecer cuáles son los caracteres necesarios.

El usuario ahora puede establecer la etiqueta para ese *token*; sin embargo, también puede editarla, en caso de que requiera añadir a ella información adicional y de interés para su trabajo. Así, el etiquetador permite crear etiquetas propias que se basen en EAGLES o que sean completamente nuevas, por lo que puede ser utilizado tanto para las LIT como para etiquetar fenómenos ajenos al nivel de POS en otras lenguas. Esta flexibilidad permite al usuario trabajar según el enfoque o teoría lingüística que prefiera o necesite.

También se cuenta con dos opciones para fijar la etiqueta y llevarla definitivamente al archivo XML de salida. La primera de ellas es *Simple Tag*, que toma lo que se encuentre en la barra donde se muestra la etiqueta y la fija en el archivo de salida asociada a ese *token* en particular y a su número de identificación.

Por otra parte, el botón de *Fix on Dict.* permite que aquello que se encuentre escrito en la barra de la etiqueta se fije en el archivo de *diccionario* asociado al *token* que se encuentra seleccionado; además de ello, realiza el procedimiento de fijar esa ocurrencia del *token* en el archivo XML.

Esta segunda opción debe aplicarse únicamente cuando se tenga seguridad de que la misma etiqueta podrá ser utilizada en todas las ocasiones en que se presente la misma palabra o combinación de palabras y caracteres en el *token*. Esto puede aplicarse fácilmente a artículos, signos de puntuación, preposiciones o adverbios e incluso a una gran mayoría de sustantivos, adjetivos y verbos. Es de esta manera como se alimenta el diccionario, que será usado para etiquetar automáticamente los *tokens* que coincidan con la información que contiene. Para los casos en que la etiqueta puede variar, se debe utilizar la primera opción, pues la ausencia de esa etiqueta en el *diccionario* hará que siempre se pida al usuario seleccionar las categorías adecuadas de manera manual. A continuación, presentamos un ejemplo de archivo de *diccionario*:

Como lo muestra la Figura 5, este archivo consta de varias líneas de texto que asocian cada *token* con la etiqueta que se le ha asignado. Los caracteres que se encuentran al principio y en medio de cada línea son usados por el sistema para diferenciar entre estos dos elementos. La búsqueda en el *diccionario* consiste en recorrer este conjunto de líneas, que se encuentran ordenadas alfabéticamente, y tomar de ellas la etiqueta si se encuentra una coincidencia, para llevarla luego al archivo de salida.

```
entry_ . ***** Fp
entry_ bwai ***** NCMS---
entry_ di ***** DA-CNS-
entry_ house ***** NCFS---
entry_ niu ***** AQ-CS--
```

Figura 5. *Tokens* y entradas en diccionario

Fuente: Pemberty Tamayo, (2020, p. 38).

Al repetir constantemente el proceso de alimentar el *diccionario* con nuevos *tokens* y etiquetas se permite que el etiquetador busque y fije automáticamente la mayor cantidad posible de ocurrencias de las palabras; de este modo se logra una reducción importante del esfuerzo requerido para tener un corpus completamente etiquetado en XML.

Por último, la Figura 6 ilustra cómo se vería la oración «Di bwai gwain da di niu house» etiquetada con el sistema UnderRL Tagger en su archivo de salida. El archivo XML cuenta con una identificación del texto del que se trata y con todos los *tokens* que lo componen. Para cada uno de esos *tokens* se cuenta con la información de *form*, que es la manera exacta en que aparece en el texto; *tag*, que es la etiqueta que se estableció para ella, y finalmente *id*, que es un número que lo identifica y diferencia de todos los demás *tokens* del texto. Este *id* se compone de la letra *t*, un número entero que se refiere a la posición del *token* en el texto y otro número entero que se refiere a la cantidad de palabras que conforman el *token*, que varía en el caso de las *palabras multitoken*.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<text name="Ejemplo1.txt">

<token form="Di" tag="DA3CNS-" id="t.0.1"/>
<token form="bwai" tag="NCMS—" id="t.1.1"/>
<token form="gwain" tag="VMIP3SC" id="t.2.1"/>
<token form="da" tag="SP——" id="t.3.1"/>
<token form="di" tag="DA3CNS-" id="t.4.1"/>
<token form="niu" tag="AQ-FS-S" id="t.5.1"/>
<token form="house" tag="NCFS—" id="t.6.1"/>
<token form="." tag="Fp" id="t.7.1"/>
</text>
```

Figura 6. Ejemplo de XML final

Fuente: Pemberty Tamayo (2020, p. 37).

Veamos ahora un ejemplo que refleje otras de las capacidades de la aplicación. En este caso se utilizará como corpus una oración de la lengua tukano, en la que se encuentran caracteres diferentes a los que se acostumbra utilizar en español, inglés o *kriol*, así como algunos elementos que se pueden etiquetar manualmente. La oración es la siguiente (Tabla 4):

Tabla 4. Descripción de una oración en tukano por token, categoría gramatical y glosa

Oración B						
Token (palabra)	¿	Noá	wa'a-tí	cu̱	me'rã̄	?
Categoría gramatical	Puntuación	Pronombre Interrog.	Verbo	Pronombre	Preposición	Puntuación
Glosa	¿	Quién	fue (Interrogativo)	él	con	?

Como puede notarse, esta oración cuenta con caracteres que representan fonemas nasalizados y diferentes tipos de acentos y apóstrofes, así como una marca de interrogación en el verbo principal, que es una característica no contemplada por el sistema EAGLES para esta categoría gramatical. Así pues, al abrir este texto con UnderRL Tagger, se encuentra lo siguiente (Figura 7):

Lo primero que puede notarse es que todos los caracteres se muestran adecuadamente en el programa, porque también están comprendidos dentro de la codificación UTF-8, que incluye además gran parte de los utilizados en el Alfabeto Fonético Internacional.

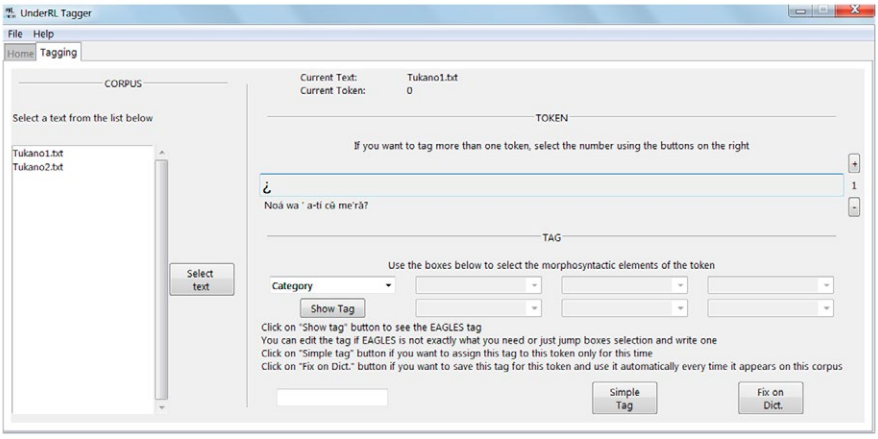

Figura 7. Ejemplo de lengua tukano con diferentes caracteres

Vale la pena aclarar aquí que, en caso de que sea necesario utilizar un sistema diferente de codificación del texto, esto depende principalmente de que sea un sistema que ya se encuentre estandarizado y en uso para equipos informáticos. Algunos ejemplos, además del UTF-8, pueden ser el SJIS para el japonés y el ISO-8859-9 para el turco, que permiten la representación de los caracteres de esos sistemas en nuestras pantallas. Así, si en algún momento se quisiera trabajar con un sistema diferente al establecido en el programa, un usuario experimentado podría modificar en código en Python (ver lista de referencias al final del artículo) para permitirlo. Si no se cuenta con la experiencia adecuada en este lenguaje de programación, esta no es actualmente una configuración disponible en la interfaz de usuario.

Otra característica de la Oración B es que posee una marca de interrogación en el verbo principal, haciendo necesario que un investigador interesado en esta información deba añadirla manualmente (Figura 8):

La Figura 8 muestra una posible solución para este inconveniente; en ella, un usuario introdujo todas las características del verbo por medio del sistema de listas desplegables y posteriormente personalizó la etiqueta generada para añadirle la información de su interés. Como ha mencionado, esta es otra de las facilidades que el sistema ofrece.

Finalmente, pueden verse en el archivo XML resultante del etiquetado que constan todos los elementos, desde los caracteres que se introdujeron como corpus de ejemplo hasta las etiquetas generadas de forma personalizada (Figura 9).

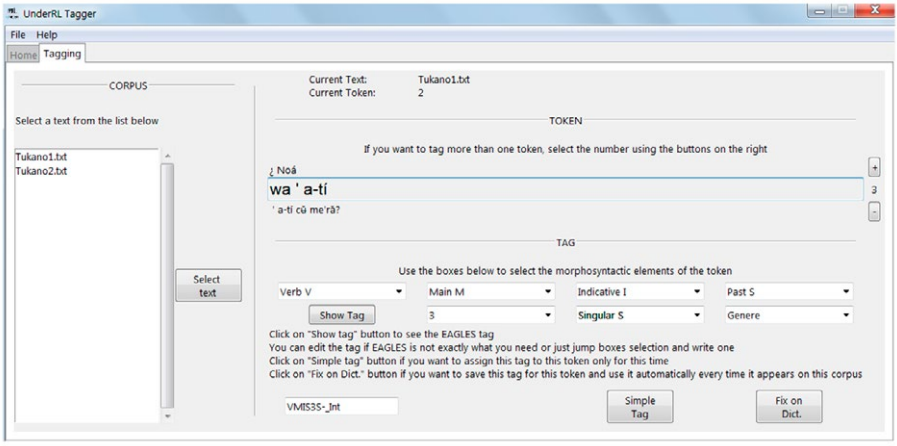

Figura 8. Ejemplo de etiqueta personalizada en la interfaz

```
<?xml version="1.0" encoding="UTF-8" standalone= "yes"?>

<text name="Tukano1.txt">

<token form="¿" tag="Fia" id="t.0.1"/>
<token form="Noá" tag="DT--SS-" id="t.1.1"/>
<token form="wa'a-tí" tag="VMIS3S-_Int" id="t.2.3"/>
<token form="cʉ̃" tag="DA3MS--" id="t.5.1"/>
<token form="me'rã" tag="SP-----" id="t.6.1"/>
<token form="?" tag="Fit" id="t.7.1"/>
</text>
```

Figura 9. Ejemplo de XML con etiquetas de Oración B

Es muy importante notar que las herramientas para visualización de archivos .txt o archivos XML no dependen únicamente del sistema que aquí se presenta, y por ello pueden no estar debidamente configuradas para mostrar caracteres UTF-8 en la pantalla del usuario, aunque la gran mayoría, incluidas las nativas de los diferentes sistemas operativos, lo soportan de manera adecuada.

Este segundo ejemplo presenta las maneras en que puede utilizarse la aplicación para enfrentar casos más complejos, tanto en la codificación de la lengua que se utilice como en la generación de etiquetas propias, ya sea porque no es suficiente con el sistema EAGLES o porque el investigador desea etiquetar en los textos cuestiones ajenas al etiquetado de POS.

5. Conclusiones y perspectivas

A través de una interfaz basada en ventanas, menús y controles sencillos, UnderRL Tagger (Pemberty Tamayo et al., 2020) permite realizar un proceso de etiquetado manual asistido y automatizado, lo que posibilita a los investigadores en descripción gramatical y léxica acogerse a estándares internacionales de la lingüística de corpus, elegir un sistema de etiquetas propio e incluso realizar etiquetados por fuera del POS con cualquier otro fenómeno que se desee. De la misma manera, permite el manejo de archivos de diccionario que pueden servir en un futuro para seguir etiquetando textos en la misma lengua o compartirlos con otros investigadores.

Nuestra propuesta ilustra de qué manera es posible que las aplicaciones de la lingüística computacional se utilicen en el etiquetado de corpus en lenguas que no cuentan aún con acceso a las herramientas de etiquetado automático, lo que permite la automatización progresiva de procesos cuya ejecución manual resulta dispendiosa.

Aunque UnderRL Tagger es un programa de uso especializado para el trabajo lingüístico, su interfaz de usuario y sus procedimientos de almacenamiento y recuperación de información están diseñados de manera accesible; así, presenta una curva de aprendizaje suave y eficiente que no demanda conocimientos informáticos previos, por lo que los investigadores y aprendices en lingüística pueden aprovechar sus virtudes sin una formación intensiva previa.

En síntesis, UnderRL Tagger se ofrece como una herramienta que promueve el acceso de las LIT a las tecnologías de la información y la comunicación; de esta manera, este programa se suma a los esfuerzos por simplificar el acceso de estas lenguas como objeto de investigación (Cunliffe et al., 2022).

Finalmente, es importante reiterar que se trata de un programa informático de acceso libre, lo cual es otro aporte a la equidad y la democracia en la generación de conocimiento. Desde la Universidad de Antioquia, ofrecemos esta herramienta a los investigadores en lenguas minoritarias e infrasoportadas tecnológicamente de nuestro país y del mundo; así, atendiendo a las invitaciones de los expertos en el campo (Ostler, 2014; Rojas Curieux, 2017), confiamos en que UnderRL Tagger resulte un apoyo pertinente para los procesos de revitalización de lenguas.

6. Referencias

Anthony, L. (2022). *TagAnt (Version 2.0.5)* [Software]. https://www.laurenceanthony.net/software

Besacier, L., Barnard, E., Karpov, A., & Schultz, T. (2014). Automatic speech recognition for under-resourced languages: A survey. *Speech Communication*, *56*, 85-100. https://doi.org/10.1016/j.specom.2013.07.008

Bonilla, J. E., Rubio López, R. Y., Llanos Chávez, A. L., Bejarano, D. E., & Bernal Chávez, J. A. (2020). Proyecto de Digitalización y Nuevas Perspectivas Tecnológicas del Atlas Lingüístico-Etnográfico de Colombia. En Gallego, A., & Roca Urgell, F. (eds.), *Dialectología digital del español*, 13, pp. 13-28. Santiago de Compostela: Universidad de Santiago de Compostela.

Camacho, L., & Zevallos, R. (2020). Lingüística computacional para la revitalización y el poliglotismo. *Letras*, *91*(134), 184-198. http://doi.org/10.30920/letras.91.134.9

Cunliffe, D., Vlachidis, A., Williams, D., & Tudhope, D. (2022). Natural language processing for under-resourced languages: Developing a Welsh natural language toolkit. *Computer Speech & Language*, *72*, 101311. https://doi.org/10.1016/j.csl.2021.101311

Dueñas, G., & Gómez, D. (2015). Diccionario electrónico sáliba-español: una herramienta interactiva para la documentación de la lengua y de la cultura sálibas. *Forma y Función*, *28*(2), 49-61. http://dx.doi.org/10.15446/fyf.v28n2.53539

Dueñas, G., & Gómez, D. (2016). Building Bilingual Dictionaries for Minority and Endangered Languages with Mediawiki. *CCURL 2016 Collaboration and Computing for Under-Resourced Languages: Towards an Alliance for Digital Language Diversity* (pp. 9-15).

Eberhard, D., Simons, G., & Fennig, C. (eds.) (2019). *Ethnologue: Languages of the World* (22 ed.). SIL International.

El-Haj, M., Kruschwitz, U., & Fox, C. (2015). Creating language resources for under-resourced languages: Methodologies, and experiments with Arabic. *Language Resources and Evaluation*, *49*(3), 549-580. https://doi.org/10.1007/s10579-014-9274-3

García, M., Gómez-Rodríguez, C., & Alonso, M. (2016). Creación de un treebank de dependencias universales mediante recursos existentes para lenguas próximas: el caso del gallego. *Procesamiento del Lenguaje Natural*, *57*, 33-40. https://www.redalyc.org/articulo.oa?id=515754424003

Leech, G., & Wilson, A. (1996). *EAGLES recommendations for the morphosyntactic annotation of corpora.* Istituto di Linguistica Computazionale. http://www.ilc.cnr.it/EAGLES96/annotate/node1.html

Maxwell, M., & Hughes, B. (2006). Frontiers in linguistic annotation for lower-density languages. *Proceedings of the workshop on frontiers in linguistically annotated corpora 2006*, 29-37. https://minerva-access.unimelb.edu.au/handle/11343/34524

McCrae, J., & Doyle, A. (2019). Adapting Term Recognition to an Under-Resourced Language: the Case of Irish. En Lynn, T., Prys, D., Batchelor, C., & Tyers, F. (eds.), *Proceedings of the Celtic Language Technology Workshop* (pp. 48-57). European Association for Machine Translation. https://aclanthology.org/W19-6907/

McCrae, J., Ojha, A., Chakravarthi, B., Kelly, I., Buffini, P., Tang, G., Paquin, E., & Locria, M. (2021). Enriching a terminology for under-resourced languages using knowledge graphs. En Kosem, I., Cukr, M., Jakubíček, M., Kallas, J., Krek, S., & Tiberius, C. (eds.), *Electronic lexicography in the 21st century: post-editing lexicography. Proceedings of the eLex 2021 conference* (pp. 560-571). Lexical Computing CZ. https://elex.link/elex2021/wp-content/uploads/eLex_2021-proceedings_compressed.pdf

Mikheev, A. (2014). Text Segmentation. En Mitkov, R. (ed.), *The Oxford Handbook of Computational Linguistics 2nd edition*. Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199573691.013.34

Mohammed, S. (2020). Using machine learning to build POS tagger for under-resourced language: the case of Somali. *International Journal of Information Technology*, *12*, 717-729. https://doi.org/10.1007/s41870-020-00480-2

Montenegro, A., & Guaquetá, M. (2020). Identificación de los posibles fonemas vocálicos de la lengua tanimuka aplicando Machine Learning. En Molina, A., Valdivia, P., & Venegas, R. (eds.), *III Congreso Internacional de Lingüística Computacional y de Corpus: Una mirada desde las tecnologías del lenguaje y las Humanidades Digitales (CILCC 2020) y V Workshop en Procesamiento Automatizado de Textos y Corpus (WOPATEC_2020)* (p. 52). Universidad de Antioquia/University of Groningen. https://dialnet.unirioja.es/servlet/libro?codigo=786453

Núñez, E. (2013). Minorías lingüísticas y derecho a las lenguas. *Revista Internacional d'Humanitats*, *27*, 7-28. http://www.hottopos.com/rih27/

Ostler, N. (2014). Endangered languages in the New Multilingual Order *per genus et differentiam*. En Jones, M. C. (ed.), *Endangered Languages and New Technologies* (pp. 1-13). Cambridge University Press. https://doi.org/10.1017/CBO9781107279063.002

Padró, L., Collado, M., Reese, S., Lloberes M., & Castellón, I. (2010). FreeLing 2.1: Five Years of Open-Source Language Processing Tools. En Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odij, J., Piperidis, S., Rosner, M., & Tapias, D. (eds.), *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2010)* (pp. 931-936). ELRA.

Passonneau, R., & Mani, I. (2014). Evaluation. En Mitkov, R. (ed.), *The Oxford Handbook of Computational Linguistics 2nd edition*. Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199573691.013.014

Pemberty Tamayo, J. L. (2020). *Concepción y elaboración de un sistema de etiquetado semiautomático para under-resourced languages* [tesis de pregrado, Universidad de Antioquia]. http://bibliotecadigital.udea.edu.co/handle/10495/16570

Pemberty Tamayo, J. L., Molina Mejía, J. M., & Marín Morales, M. I. (2020). *UnderRL Tagger* (Versión 1.0) [Software]. Universidad de Antioquia. https://github.com/jluispemberty/UnderRlTagger

Pereira, J. H. (2018). *Implementación de un lematizador para una lengua de escasos recursos: caso shipibo-konibo* [tesis de pregrado, Pontificia Universidad Católica del Perú]. https://tesis.pucp.edu.pe/repositorio/handle/20.500.12404/13495

Ramírez-Cruz, H., & Chaparro Rojas, J. F. (2021). Introducción: la diversidad lingüística y la investigación de lenguas en peligro. *Forma y Función*, *34*(2). http://doi.org/10.15446/fyf.v34n2.96558

Rojas Curieux, T. (comp.) (2017). *Corpus lingüísticos: estudio y aplicación en revitalización de lenguas indígenas*. Universidad del Cauca.

Sanmiguel Ardila, R., Schoch Angel, M., & Loaiza-Camacho, B. S. (2021). Retos y oportunidades contemporáneos del kriol en el archipiélago de San Andrés, Providencia y Santa Catalina, Colombia. *Forma y Función*, *34*(2). https://doi.org/10.15446/fyf.v34n2.88613

Schmid, H. (1994). *TreeTagger - a part-of-speech tagger for many languages* [Software]. https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

Schuster, S., & Manning, C. D. (2016). Enhanced English Universal Dependencies: An Improved Representation for Natural Language Understanding Tasks. En Calzolari, N., Choukri, J., Declerck, T., & Moreno, A. (eds.), *10th conference on International Language Resources and Evaluation (LREC'16)*. LREC. https://aclanthology.org/L16-1376/

Sene-Mongaba, B. (2015). The Making of Lingala Corpus: An Under-resourced Language and the Internet. *Procedia - Social and Behavioral Sciences*, *198*, 442-450. https://doi.org/10.1016/j.sbspro.2015.07.464

Somers, H. (2003). Translation technologies and minority languages. En. Somers, H. (ed.), *Computers and Translation* (pp. 87-103). John Benjamins. https://doi.org/10.1075/btl.35.09som

Summer Institute of Linguistics (SIL). (2019). *Field Linguist's ToolBox* (Version 1.6.4). [Software] https://software.sil.org/toolbox/

Summer Institute of Linguistics (SIL). (2021). *FieldWorks Language Explorer* (Version 9.0). [Software] https://software.sil.org/fieldworks/

UNESCO. (2014). *Una cátedra al servicio de comunidades indígenas.* Cátedra Unesco en Tecnologías Lingüísticas. https://www.teclin.org/Una-catedra-al-servicio-de-comunidades-indigenas_a6.html

Weisser, M. (2018). Automatically Enhancing Tagging Accuracy and Readability for Common Freeware Taggers. En Tono, Y., & Isahara, H. (eds.), *Proceedings of APCLC 2018* (pp. 502-505). Asia Pacific Corpus Linguistics Association.

Woodbury, A. C. (2014). Archives and audiences: Toward making endangered language documentations people can read, use, understand, and admire. *Language documentation and description*, *12*, 19-36.

NOTAS

1 El término original en inglés es under-resourced languages, que suele alternar con otros términos equivalentes como low-density languages, resource-poor languages, low-data languages o less-resourced languages (Besacier et al., 2014, p. 87). En español, aunque aparecen otras traducciones como lenguas de pocos recursos (García et al., 2016) y lenguas de escasos recursos (Pereira, 2018), preferimos lenguas infrasoportadas tecnológicamente pues, a nuestro criterio, expresa con mayor claridad la situación referida, además de que el término infrasoportado es la elección de la Cátedra UNESCO en Tecnologías Lingüísticas al referir a las comunidades infrasoportadas en sus lenguas maternas (UNESCO, 2014).

2 https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

3 https://www.laurenceanthony.net/software/tagant/

4 http://nlp.lsi.upc.edu/freeling/node/1

5 https://nlp.stanford.edu/software/lex-parser.shtml

6 Proyecto construido en la articulación interdisciplinaria del Grupo de Estudios Lingüísticos, Pedagógicos y Socioculturales del suroccidente colombiano (GELPS) y el Grupo I + D Tecnologías de la Información y la Comunicación (GTI) de la misma universidad.

7 https://software.sil.org/toolbox/

8 https://software.sil.org/fieldworks/