

Application of Language Technologies to Assist English Teaching

1st Bernabé González
Instituto de Computación
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
bernabe.gonzalez@fing.edu.uy

2nd Isabel Ivagnes
Instituto de Computación
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
isabel.ivagnes@fing.edu.uy

3rd Joaquín Lejtregger
Instituto de Computación
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
joaquin.lejtregger@fing.edu.uy

4th Luis Chiruzzo
Instituto de Computación
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
luischir@fing.edu.uy

5th Aiala Rosá
Instituto de Computación
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
aialar@fing.edu.uy

Abstract—This work explores the use of standard Natural Language Processing (NLP) tools for the development of teaching support tools. We use these methods for developing an application that allows to automatically create different types of English exercises aimed at students at the beginners level. The web application contains two interfaces, where teachers can generate four types of exercises starting from English texts, and students can solve them. The four types of exercises are: verb form exercises, noun definition exercises, use of English exercises, and lexical relations exercises. Some of the NLP techniques and tools explored during development include: POS-tagging, language models, word embeddings, Levenshtein distance and WordNet. Developing this tool allowed us to test the potential of Natural Language Processing techniques to generate teaching activities. As a way of validating the tool, we tested it with local teachers and found that there is an important demand for these type of tools in our country.

Index Terms—NLP, AI in Education, English Learning, Word Embeddings, Language Models

I. INTRODUCCIÓN

El Procesamiento de Lenguaje Natural (PLN) es un área de investigación que en los últimos años ha tenido un importante desarrollo debido al aumento de la cantidad de texto disponible, principalmente en Internet, que conlleva la necesidad de automatizar el procesamiento de esa gran masa de información. Este desarrollo del PLN ha potenciado otras áreas de aplicación de estas tecnologías, como la enseñanza, y en particular la enseñanza de lenguas, en donde las tecnologías pueden dar apoyo al trabajo del docente y aportar a un mejor desempeño de los estudiantes.

En esta área específica de aplicación del PLN se enmarca este trabajo, en el que se describe el desarrollo de

una herramienta informática para ser utilizada por maestros que enseñan inglés en sus clases. El trabajo se desarrolló en colaboración con el Programa de Políticas Lingüísticas de ANEP (Administración Nacional de Enseñanza Pública, Uruguay)¹, que lleva adelante un proyecto de enseñanza de inglés en el medio rural, orientado a niños de cuarto a sexto año de enseñanza primaria. En este proyecto, maestros de aula, asistidos en modalidad *e-coaching* por profesores especializados en enseñanza de inglés, enseñan el idioma en sus clases. En este contexto la asistencia a los maestros en la generación de actividades y ejercicios para trabajar con los alumnos es fundamental.

Las herramientas desarrolladas incluyen actividades para maestros y niños. La aplicación que describimos en este trabajo permite la generación de ejercicios clásicos de práctica del idioma, como completar espacios en blanco, unir palabras con definiciones y asociar palabras con categorías. Este desarrollo se llevó adelante con el fin de explorar el potencial del PLN para la generación de actividades educativas. Se hizo una encuesta primaria a docentes para evaluar el interés en herramientas de este tipo, quedando pendiente una evaluación de su efectividad para mejorar los aprendizajes de los niños.

En lo que sigue se describe el trabajo desarrollado: se presentan brevemente el área PLN y algunas de sus aplicaciones a la enseñanza (sección II), luego se describe la aplicación desarrollada (sección III), los cuatro tipos de ejercicios (sección IV), y los resultados obtenidos en las pruebas de validación con usuarios (sección V). Finalmente se presentan conclusiones y líneas futuras de trabajo (sección VI).

¹<https://www.anep.edu.uy/codicen/politicas-linguisticas/focus-on-first>

II. PROCESAMIENTO DE LENGUAJE NATURAL APLICADO A LA ENSEÑANZA

El Procesamiento de Lenguaje Natural (PLN) es una subdisciplina de la Inteligencia Artificial que intenta resolver con computadoras tareas vinculadas al lenguaje humano, permitiendo la comunicación entre el humano y la computadora a través del lenguaje natural o resolviendo diferentes tareas que implican algún tipo de procesamiento de texto o habla [1].

El desarrollo de aplicaciones de PLN orientadas a colaborar con tareas de enseñanza es un tema de estudio en el área desde sus inicios. El vínculo entre el PLN y la enseñanza puede verse desde diferentes ángulos [2]. Es posible construir herramientas que asistan al docente en sus tareas, tanto en la generación de ejercicios como en la corrección de los trabajos de los estudiantes, o herramientas para asistir a los estudiantes en su aprendizaje. En los dos casos, puede tratarse de enseñanza de lenguas o de otras disciplinas en donde el lenguaje es solamente el medio para transmitir las ideas. También es posible aplicar técnicas de PLN, principalmente las basadas en métodos estadísticos, para el procesamiento de diferentes tipos de información digital que se genera en torno a la enseñanza, como los intercambios que se dan en los foros en donde estudiantes y docentes interactúan; o todos los datos que se generan sobre los estudiantes, sus características y desempeño, a lo largo de cada año lectivo. Esta gran cantidad de información digital, que aumenta a medida que aumenta el uso de tecnologías en las aulas, es imposible de procesar en forma manual.

Un ejemplo de aplicación en donde el PLN permite generar ejercicios para la práctica de inglés es Language Muse² [3], que permite seleccionar un texto de un catálogo, o utilizar textos propios, y generar a partir de ellos ejercicios que evalúan conceptos morfológicos, sintácticos o semánticos, ofreciendo además la posibilidad de gestionar los resultados de un grupo de estudiantes. Estas ideas van en línea con otros desarrollos que se están realizando en la actualidad para otros idiomas [4]. En [5] los autores intentan construir recursos para el aprendizaje a partir de transformaciones de cuentos infantiles utilizando herramientas de PLN. Un trabajo que tiene un enfoque muy relacionado con el nuestro es [6], donde se intenta construir ejercicios de inglés de diferentes tipos (algunos similares a los que presentamos aquí), pero principalmente enfocándose en el aprendizaje de colocaciones en inglés para niveles más avanzados.

Entre los recursos necesarios para el desarrollo de este tipo de herramientas, se encuentran textos y léxicos de nivel y contenido apropiados, ontologías, diccionarios con definiciones orientadas a hablantes no nativos, entre otros. Para obtener textos acordes al nivel deseado se ha trabajado en diferentes líneas: definición de índices de complejidad, como por ejemplo el Automated Readability Index (ARI) [7], entrenamiento de herramientas para simplificación de texto [8], [9], utilización de modelos entrenados para construir textos de un nivel apropiado [10].

²<https://languagemuse.org/>

Otra línea de investigación importante es la corrección automática de trabajos de estudiantes. Es posible pensar en contar con verificadores automáticos de ortografía y gramática, que asistan al docente en la tarea de corrección de textos o que permitan a un estudiante corregir oraciones agramaticales u oraciones que no suenen naturales [11], [12]. Para estas tareas, es usual la utilización de herramientas de análisis lingüístico, como analizadores morfológicos y sintácticos. En relación a esta tarea, se han organizado algunas campañas de evaluación (en formato de competencias), en el marco de reconocidos eventos internacionales (SemEval y ConNLL) orientados a la aplicación de técnicas estadísticas sobre conjuntos de datos con ejemplos ya resueltos, provistos por los organizadores. En SemEval 2013 [13] se presentó una tarea de corrección automática de textos, en base a grandes corpus de textos escritos por estudiantes que incluyen correcciones de docentes. En ConNLL 2013 [14] se presentó una tarea de corrección automática de respuestas a preguntas, también en base a un corpus de ejemplos. Por otro lado, en 2018, en el marco del BEA 13, se desarrolló una campaña de evaluación basada en un conjunto de datos proveniente de la aplicación Duolingo [15], orientada a la adquisición de lenguas extranjeras. Estas tareas no alcanzan aún altos niveles de precisión, por lo que existe un amplio margen para investigar en estos temas.

En cuanto al análisis semántico del lenguaje, que es fundamental para la evaluación del dominio de un idioma por parte de un estudiante, es posible aplicar algunas técnicas usuales en PLN como los modelos de lenguaje basados en n-gramas [1], los modelos neuronales de lenguaje [16] o las representaciones distribuidas de palabras (*word embeddings*) [17], de gran uso en la actualidad. Los modelos de lenguaje permiten estimar la probabilidad de que ciertas secuencias de palabras sean sintáctica y semánticamente correctas para un cierto lenguaje. Por otro lado, los *word embeddings* permiten representar las palabras como vectores y calcular distancias entre ellas de modo de encontrar conjuntos de palabras semánticamente relacionadas.

III. DESCRIPCIÓN DE LA APLICACIÓN

En esta sección se describe la aplicación desarrollada y los recursos que fue necesario recopilar o construir.

A. Generación de ejercicios a partir de textos

La aplicación construida es una plataforma para la generación de ejercicios de inglés que permite generar cuatro tipos de ejercicios, todos ellos a partir de un texto de entrada. De este modo un docente puede trabajar con un texto de un nivel que considere apropiado para su clase y mediante esta herramienta generar uno o varios ejercicios que estén relacionados con el contenido del texto.

El sistema puede funcionar en dos modalidades. La modalidad docente permite crear y editar ejercicios basados en textos de entrada o eligiendo un texto de una base precargada (ver Fig. 1). Los docentes pueden también ver los ejercicios generados anteriormente y explorar o editar las listas de palabras y definiciones. La modalidad estudiante solamente despliega

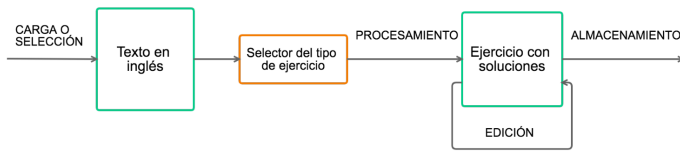


Fig. 1: Flujo para la generación de un ejercicio.

los ejercicios y permite que el estudiante los resuelva, pero no tiene las opciones de generar ejercicios nuevos o editarlos.

Los cuatro tipos de ejercicios que se generan son: conjugación verbal, definiciones de sustantivos, uso de lenguaje, y relaciones léxicas. Su funcionamiento se detalla en la sección IV.

B. Recursos utilizados

El objetivo de la herramienta es dar soporte a la enseñanza de inglés en escuelas, lo cual restringe en cierta medida el tipo de textos que se pueden utilizar, tanto en su complejidad como en el tipo de temáticas que pueden abordar. Contamos con un conjunto pequeño de textos en inglés de nivel principiante y un léxico acotado que corresponde con el nivel que se quiere alcanzar. Este léxico (que denominaremos *léxico básico*) es un vocabulario basado en los niveles correspondientes a *Starters* y *Movers* del Cambridge Assessment English, o niveles pre-A1 y A1 del MCER.

Se obtuvieron más textos y recursos léxicos de diferentes fuentes de internet: conjuntos de textos simples y diccionario para niños de nivel principiante. En particular, se utilizó Simple Wikipedia³, que consiste en una traducción simple de 142.910 artículos de English Wikipedia, creada para facilitar la lectura a niños y personas que están aprendiendo el idioma o que tengan dificultades para leer. Otro recurso interesante es el Proyecto Gutenberg⁴, que contiene una gran variedad de libros cuyo copyright expiró. Si bien son textos que contienen lenguaje antiguo, también contienen muchas palabras adecuadas para niños ya que se incluyen libros infantiles. También se utilizó una lista de palabras frecuentes tomada de Wiktionary⁵, un proyecto colaborativo que desarrolla diccionarios para diferentes idiomas. La lista está basada en los libros del proyecto Gutenberg.

Por otra parte, se utilizaron diferentes herramientas de análisis lingüístico. En particular, se usó el analizador morfosintáctico (POS-tagger) provisto por el grupo NLP de Stanford [18]. El POS-tagger brinda, para cada palabra de un texto, su lema, su categoría gramatical y ciertos rasgos morfológicos, como número, género, etc., dependiendo de la categoría. También se utilizó WordNet [19], un recurso léxico que organiza las palabras en una ontología, en donde se representan los conceptos como clases de sinónimos (*synsets*) y se establecen entre ellos diferentes relaciones léxicas como hiponimia, meronimia, antonimia, etc. Para cada concepto se

incluye una glosa que lo define. En esta ontología, las palabras polisémicas pertenecen a diferentes (*synsets*).

Como se describe en las secciones IV-C1 y IV-C2, también fue necesario construir recursos nuevos: un modelo de lenguaje y una colección de word embeddings adaptados para nuestra aplicación.

IV. EJERCICIOS

En esta sección se describen en detalle los cuatro tipos de ejercicios que se pueden generar utilizando la aplicación.

A. Ejercicios de conjugación verbal

Los ejercicios de conjugación verbal son ejercicios de tipo completar espacios en blanco (*fill in the blanks*) donde un estudiante debe elegir la conjugación correcta de un verbo a utilizar en un texto. El generador de ejercicios obtiene todos los verbos del texto de entrada y genera diferentes opciones para cada uno, seleccionando diferentes conjugaciones. Solo se tienen en cuenta los verbos pertenecientes al léxico básico. Estos verbos son sustituidos por espacios en blanco en el texto, el estudiante deberá elegir la conjugación correcta para completar cada espacio. En cada espacio en blanco se aclara cuál es el tiempo verbal del verbo a completar de modo de evitar posibles ambigüedades.

Se utiliza la herramienta Stanford POS-tagger para etiquetar los verbos y analizar sus características de forma de generar las opciones correctas. Para generar los distractores, se utiliza la librería Pattern⁶ que permite obtener todas las posibles conjugaciones de un verbo. Para cada una de las palabras se eligen tres conjugaciones (diferentes a la conjugación original del verbo). Se puede ver un ejemplo de ejercicio generado en la Fig. 2.

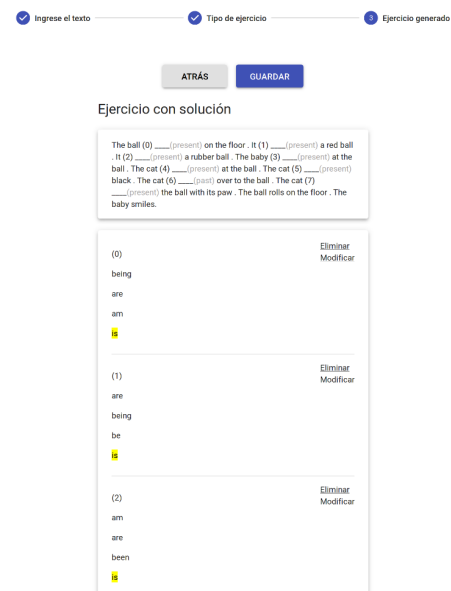


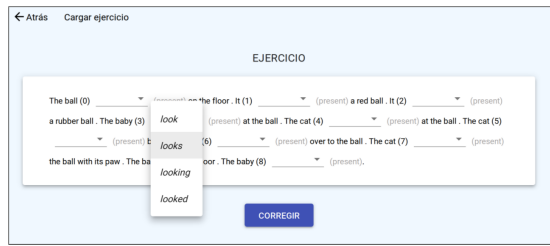
Fig. 2: Captura de la generación de un ejercicio de conjugación verbal.

³<https://simple.wikipedia.org/>

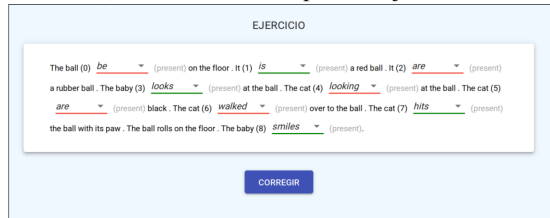
⁴<https://www.gutenberg.org/>

⁵<https://www.wiktionary.org/>

⁶<https://www.clips.uantwerpen.be/pages/pattern>



(a) El estudiante completa el ejercicio.



(b) La aplicación indica los errores.

Fig. 3: Capturas de la vista de estudiante de un ejercicio de conjugaciones.

La Fig. 3 muestra la vista de estudiante de este tipo de ejercicios. Inicialmente el estudiante se encontrará con un texto con espacios en blanco. El estudiante completa cada espacio con alguna de las opciones disponibles y elige “corregir” cuando haya terminado. La aplicación contrastará las respuestas del estudiante con las válidas y dará un feedback inmediato indicando qué opciones están bien (subrayadas en verde) y cuáles presentan error (subrayadas en rojo).

B. Ejercicios de definiciones de sustantivos

Los ejercicios de definiciones de sustantivos son ejercicios de correspondencias (*match*) para establecer la relación entre un conjunto de sustantivos y sus respectivas definiciones. El generador de ejercicios busca todos los sustantivos del texto que se encuentren dentro de una base de definiciones precargadas. El ejercicio presenta una lista con los sustantivos detectados y otra lista con sus definiciones. El estudiante deberá marcar la correspondencia entre unos y otros. En este ejercicio el docente puede sustituir las definiciones por otras que considere más adecuadas.

La definición de cada sustantivo se busca en el diccionario precargado, si existe una única definición, esta es seleccionada, pero si existe más de una definición, se aplica el algoritmo de Lesk [20] para buscar en la ontología WordNet el *synset* que tenga más sentido para el contexto. Este algoritmo se basa en el supuesto de que dos palabras son similares si comparten palabras en sus respectivas definiciones. En este caso, lo que hace el algoritmo de Lesk es seleccionar, entre todos los *synsets* posibles, el que tiene en su definición más palabras en común con el contexto en donde aparece la palabra que estamos definiendo.

A pesar de que el algoritmo de Lesk no es del todo preciso, como se trata de textos y definiciones simples funciona bastante bien en este contexto. Por otro lado, si bien existen otros enfoques para la desambiguación del sentido de las palabras,

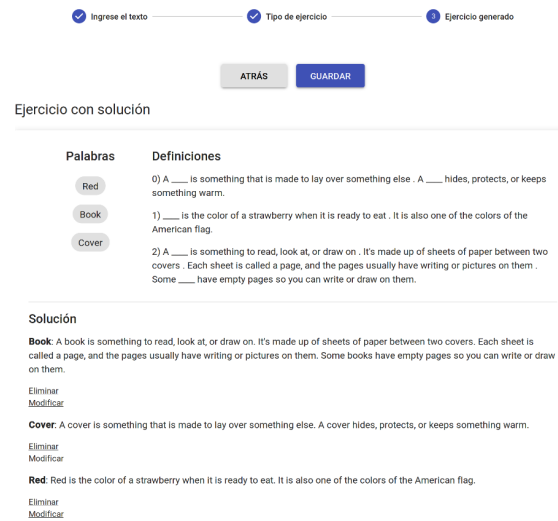


Fig. 4: Captura de la generación de un ejercicio de definiciones de sustantivos.

la mayoría requieren un gran esfuerzo computacional, lo cual se consideró innecesario ya que en estos textos simples la definición suele ser encontrada en el diccionario y además el docente tiene la oportunidad de corregir la definición encontrada por el algoritmo, en caso de no ser apropiada.

La Fig. 4 muestra la interfaz de generación de un ejercicio de definiciones de sustantivos para un docente. Como se puede observar, es posible modificar o eliminar cada definición propuesta.

La Fig. 5 muestra la vista de estudiante de uno de estos ejercicios. Al estudiante se le presentan los posibles sustantivos y las definiciones, y se espera que escriba los sustantivos en las posiciones correctas. Notar que en este caso, como el estudiante escribe un texto libre, se indica mediante un código de colores (rojo, amarillo o verde) qué tan correcta es la respuesta. La clasificación de una respuesta como correcta, incorrecta, o semi-correcta se hace utilizando la distancia de edición de Levenshtein de la respuesta del estudiante respecto a la esperada.

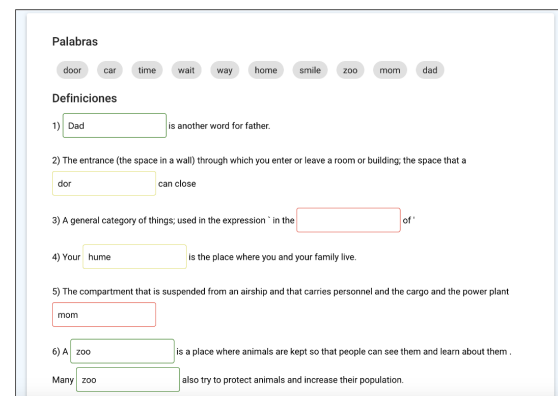


Fig. 5: Captura de la vista de estudiante de un ejercicio de definiciones de sustantivos.

C. Ejercicios de uso del lenguaje

Los ejercicios de uso del lenguaje “*Use of English*” están pensados para que el estudiante pueda detectar palabras y frases de uso frecuente. El generador de ejercicios toma un texto de entrada, elige ciertas palabras para sustituirlas por espacios en blanco y da opciones candidatas para su sustitución. No tiene sentido elegir palabras al azar como distractores, ya que serían fácilmente descartables.

Por ejemplo, si tomamos la oración “*Max eats a sandwich for breakfast*”, el generador de ejercicios puede extraer la palabra *sandwich* y generar distractores que cumplan con las siguientes condiciones:

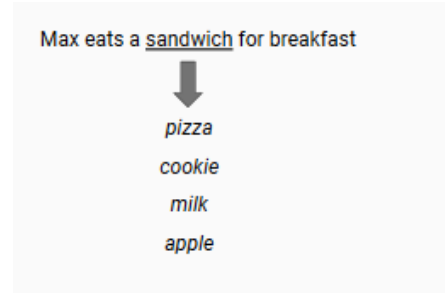
- 1) Debe ser una palabra similar a *sandwich*, en particular, es deseable que sean otros sustantivos.
- 2) Debe ser lo suficientemente distinta de la palabra *sandwich* como para que la oración no tenga un sentido apropiado con otra palabra, ya que no debería haber más de una opción correcta.

Para poder cumplir con las dos condiciones a la vez, se utilizan el modelo de lenguaje que se describe en la sección IV-C1 y la colección de word embeddings que se describe en la sección IV-C2. Utilizando word embeddings se obtiene una colección de palabras cercanas a *sandwich*, que probablemente incluirá comestibles como *pizza* o *cookie* que podrían sustituirse correctamente en el contexto, y otras candidatas con significados cercanos pero que no quedan bien en este contexto como *milk* o *apple* (Fig. 6a). Luego se calcula la probabilidad de cada una de las palabras en el contexto según el modelo de n-gramas (Fig. 6b), finalmente se eligen las palabras de menor probabilidad para ser usadas como distractores (Fig. 6c). De este modo, es muy poco probable que entre las opciones queden otras palabras que generen oraciones correctas. Un enfoque similar es utilizado en [6] para la generación de distractores en ejercicios de múltiple opción, pero en ese caso solamente se utiliza la colección de embeddings y heurísticas de distancia de vectores para evitar el problema de la alta similitud, mientras que en nuestro caso utilizamos un segundo criterio probabilístico mediante el modelo de lenguaje de n-gramas, lo cual puede hacer más robusto el proceso.

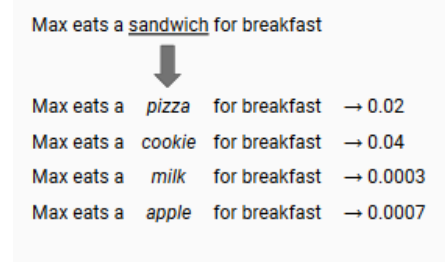
La Fig. 7 muestra una captura de la aplicación al generar un ejercicio de uso de lenguaje. En este modo el docente puede eliminar algunas de las palabras o modificar los distractores sugeridos.

1) *Modelo de Lenguaje de n-gramas*: Se construyó un modelo de lenguaje para evaluar los distractores de los ejercicios de *use of English*. El modelo de lenguaje construido es un modelo de n-gramas, en particular trigramas. Para esto se utilizó la librería KenLM⁷, que además de construir el modelo ofrece un método para obtener la probabilidad de una cadena de palabras. Se realizaron dos experimentos: uno entrenando con el corpus de Wikipedia Simple y otro con la Wikipedia en inglés estándar.

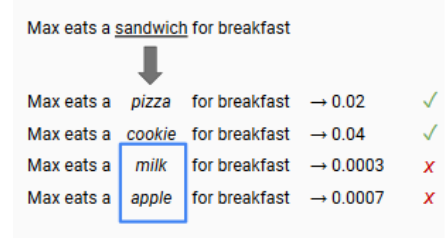
El modelo generado usando la Wikipedia simple era aceptable para buscar, entre varias oraciones, las dos más distantes



(a) Palabras más cercanas según word embeddings.



(b) Probabilidades según modelo de lenguaje.



(c) Se eligen como distractores las menos probables.

Fig. 6: Generación de distractores en un ejercicio de *use of English*.

en probabilidad (aplicación necesaria para generar los ejercicios), pero no se comportaba muy bien cuando las oraciones eran similares. Por ejemplo, la oración “*He eats his dog*” tiene una mejor probabilidad que “*He eats his breakfast*”. Entrenando con la Wikipedia estándar obtuvimos mejores resultados. En el cuadro I se pueden ver varios ejemplos y sus valores según los dos modelos probados.

El trabajo con modelos de n-gramas genera problemas de uso de disco y memoria, el modelo generado ocupaba inicialmente 22GB y además consumía demasiada memoria al cargarlo. Dado que el uso que se pretende dar a estas aplicación

TABLE I: Probabilidades (\log_{10}) obtenidas con el modelo de lenguaje entrenado con la Simple English Wikipedia y con la Wikipedia en inglés estándar.

Texto	Simple Wiki	English Wiki
<i>he eats his breakfast</i>	-15.59	-14.03
<i>he eats his dog</i>	-14.35	-15.41
<i>he eats his bed</i>	-15.13	-17.23
<i>he eats his food</i>	-15.01	-14.15
<i>he eats his meal</i>	-16.08	-14.21

⁷<https://github.com/kpu/kenlm>



Fig. 7: Captura de la generación de un ejercicio de uso de lenguaje.

es en escuelas rurales en donde los recursos informáticos son limitados, decidimos reducir nuestro modelo de lenguaje para que ocupe menos espacio y memoria. KenLM provee una herramienta para filtrar los n-gramas luego de haber sido generados. Lo más adecuado para nuestro caso fue filtrar por vocabulario, es decir, si el trigramma no pertenece al vocabulario entonces se descarta. El vocabulario utilizado es el de las palabras más frecuentes más el léxico básico, incluyendo todas las conjugaciones verbales contenidas en estas listas. Nuestro modelo reducido pasó de 22GB a tan sólo 2.4GB. Calculando las mismas oraciones de los ejemplos del cuadro I en el modelo reducido obtuvimos exactamente los mismos valores que en el modelo no reducido. Esto ocurre porque todas las palabras de la oración pertenecen al vocabulario, entonces ningún trigramma que influye en la probabilidad fue eliminado. Esta decisión no afecta negativamente en nuestra plataforma porque las palabras que consideramos en nuestros ejercicios son solamente las que están en el vocabulario.

2) *Modelo de word embeddings*: Entrenamos un modelo de word embeddings para poder encontrar palabras similares a la palabra seleccionada en la oración, de modo de generar un conjunto de candidatos a distractores.

Para entrenar el modelo utilizamos la librería Gensim⁸, que, además de entrenar, permite utilizar el modelo de vectores. Manejamos tres conjuntos de vectores distintos, uno utilizando el corpus de Simple Wikipedia y dos con Wikipedia en inglés completa. Para evaluar con cuál quedarnos hicimos un balance entre el rendimiento y la capacidad de procesamiento que demanda cada uno de los modelos. En el cuadro II mostramos algunos ejemplos de palabras más cercanas según los diferentes modelos a modo de comparación.

⁸<https://radimrehurek.com/gensim/>

El modelo de Simple Wikipedia contiene un vocabulario menos abundante y a veces devuelve palabras que no son para nada parecidas. Los vectores de tamaño 400 dieron resultados marginalmente mejores que los de 150, pero priorizamos reducir el consumo de memoria de la plataforma, por lo que optamos por utilizar los de 150.

El tiempo de procesamiento para la generación de vectores oscila entre ocho y doce horas para cada experimento, dependiendo de los parámetros a utilizar. El modelo final pesa 500MB y los parámetros utilizados para su construcción son los siguientes: tamaño de vector 150, tamaño de ventana 5 (palabras de contexto al entrenar los vectores), las palabras a considerar tienen que aparecer un mínimo de 30 veces en el corpus.

D. Ejercicios de relaciones léxicas

Los ejercicios de relaciones léxicas consisten en clasificar palabras dentro de diferentes categorías. El generador de ejercicios detecta los sustantivos y adjetivos del texto de entrada utilizando el Stanford POS-tagger y luego utiliza la ontología WordNet para identificar la clase de cada uno.

Entre las diferentes relaciones léxicas definidas en la ontología, en este ejercicio se trabaja solamente con la relación de hiponimia. Se analizan todos los sustantivos y adjetivos del texto, buscando elementos que sean hipónimos de alguna de las categorías siguientes: animales, vehículos, colores, ropa o deportes. El ejercicio mostrará la lista de palabras detectadas para su clasificación y las categorías posibles. La Fig. 8 muestra la vista de docentes y de estudiantes para uno de estos ejercicios.

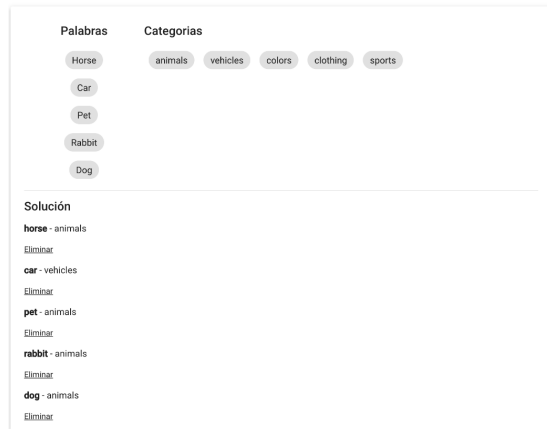
La implementación actual de este ejercicio no es del todo satisfactoria ya que los textos generalmente contienen muchas palabras de una misma categoría, es decir que no es común que un texto abarque categorías diferentes. Esto implica que el ejercicio puede trabajar con demasiadas palabras pertenecientes a la misma categoría (por ejemplo: si el texto habla de animales entonces es probable que la mayoría de sus palabras sean de la categoría animales). Como las categorías son acotadas, también puede suceder que no se encuentren suficientes palabras en el texto para que el ejercicio tenga sentido.

Otro problema no del todo resuelto aún es la forma de elegir el synset más apropiado para cada palabra, dado que es probable que una palabra tenga varios synsets asociados. Este paso es fundamental, ya que según el synset que se elija es posible llegar a diferentes clases, siguiendo las relaciones de hiponimia. En la versión actual, se implementó la opción más rápida y simple: tomar el primer synset. Como el vocabulario y las categorías con los que trabajamos son muy básicos, esta solución es bastante buena, debido a que para la mayoría de los casos la primera opción se corresponde con el sentido más frecuente dentro del vocabulario. Queda pendiente explorar enfoques basados en técnicas de desambiguación de sentidos, como el algoritmo de Lesk ya mencionado u otras.

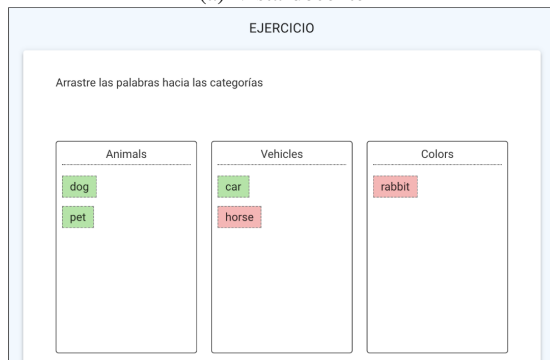
TABLE II: Ejemplos de palabras más cercanas encontradas utilizando diferentes colecciones de word embeddings generadas. En los ejemplos se utilizan palabras pertenecientes al léxico inicial.

Palabra	Simple Wikipedia 150	English Wikipedia 400	English Wikipedia 150
bear	shark, pale, bird, whale, deer, purple, cat, panda, elephant	bears, lion, rabbit, paw, ruxpin, dog, wolf, cat, panda, elephant	bears, dog, lion, rabbit, paw, toothy, monkey, hyena, wolf, elephant
be	remain, being, handle, damper, happen, surviva, recrystallize, apply, fly, stay	being, have, remain, apply, render, respond, mean, correspond, produce, handle	being, have, been, remain, get, respond, apply, handle, render, produce
pizza	syrup, pudding, ginger, peanut, sausage, soy, tomato	pizzeria, taco, pizzas, burger, sandwiches, hamburgers, pepperoni, calzone, sushi, donut	taco, pizzeria, sushi, hamburgers, nandos, pepperoni, macaroni, burger
quickly	slowly, rapidly, eventually, gradually, slow, fast, weak, easily, soon, hard	eventually, swiftly, soon, slowly, immediately, ultimately, promptly	eventually, swiftly, soon, slowly, rapidly, gradually, finally, ultimately, immediately, easily

Ejercicio con solución



(a) Vista docente



(b) Vista estudiante

Fig. 8: Capturas de la generación de un ejercicio de relaciones léxicas.

V. ENCUESTAS A DOCENTES USUARIOS DE LA HERRAMIENTA

Para tener una primera aproximación a la validación de la aplicación, se solicitó a distintos docentes que probaran las diferentes funcionalidades y completaran una encuesta. Dentro de esta sección detallamos las preguntas que hicimos y explicamos de forma resumida la retroalimentación que recibimos por parte de los entrevistados.

Entre los entrevistados contamos con maestros con más de un año de experiencia en enseñanza de inglés, estudiantes de

magisterio y profesores de inglés. Uno de ellos es docente en una escuela rural. En total se obtuvieron respuestas de siete docentes. Si bien este número no es suficiente para una evaluación formal de la aplicación, al menos sirve como un comienzo para obtener validación del funcionamiento actual e ideas nuevas para continuar mejorando la herramienta.

A continuación se listan las preguntas incluidas en la encuesta:

- ¿Cuál es el nombre de la escuela o institución?
- ¿Te pareció útil la propuesta?
- ¿Usarías el ejercicio de conjugación de verbos con tus alumnos?
- ¿Usarías el ejercicio de sustantivos con tus alumnos?
- ¿Usarías el ejercicio de uso del lenguaje con tus alumnos?
- ¿Usarías el ejercicio de hipónimos con tus alumnos?
- ¿Por último, que le cambiarías o le agregarías a cualquiera de las dos aplicaciones?

A todos los docentes encuestados les pareció útil la propuesta, aunque los motivos fueron variados. Algunos plantearon que simplifica la tarea docente, destacando la rapidez con que genera ejercicios. Otros plantearon que sirve para ocupar espacios ociosos y para manejar mejor las diferencias entre niveles de estudiantes. Otro aspecto positivo que remarcaron es lo fácil de usar que es la aplicación. Entre las recomendaciones generales, sugirieron que incorporar algún tipo de material audiovisual serviría para poder trabajar con niños más pequeños. Este es un aspecto a tener en cuenta en versiones futuras de la aplicación.

Recibimos buena retroalimentación sobre el ejercicio de sustantivos y el de uso del lenguaje. Sobre el de sustantivos opinaron que se debe acompañar con imágenes para esclarecer conceptos, y el de uso del lenguaje debería ser revisado por los maestros para controlar que la dificultad no sea excesiva.

Por último, en el ejercicio de hipónimos algunos docentes nos manifestaron que no se encontraba ninguna palabra para las categorías o que el texto contenía pocas palabras como para que resultara útil para los estudiantes.

La Fig. 9 muestra un heatmap con los resultados acerca de si usarían o no los ejercicios. Se puede apreciar que la gran mayoría usaría los cuatro ejercicios, siendo el ejercicio de uso del lenguaje el que cuenta con la aprobación de todos los docentes, lo cual es muy alentador.

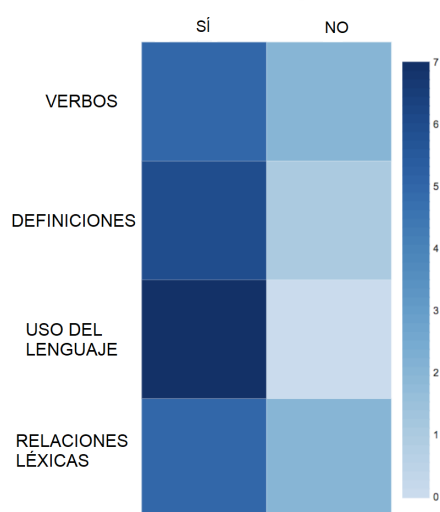


Fig. 9: Heatmap que muestra los resultados de la encuesta referidos a los ejercicios.

VI. CONCLUSIONES

Presentamos la construcción de una aplicación basada en técnicas de PLN para dar soporte a la enseñanza de inglés para estudiantes que recién están comenzando, en particular para escuelas primarias en contexto rural. La herramienta permite construir cuatro tipos de ejercicios: conjugación de verbos, definición de sustantivos, uso de lenguaje y detección de homónimos. Para su implementación se utilizaron diferentes técnicas de PLN, se recopilieron recursos lingüísticos y se generaron recursos nuevos, en particular un modelo de lenguaje de n-gramas y una colección de word embeddings. El desarrollo de esta herramienta nos permitió comprobar el potencial de las técnicas de Procesamiento de Lenguaje Natural para la generación de actividades educativas, línea en la que seguimos investigando en la actualidad. A modo de validación inicial se realizó una encuesta a docentes, la cual, si bien no constituye una evaluación formal, nos da la pauta de la utilidad de este tipo de herramientas, que pensamos serían muy bien recibidas en la comunidad de enseñanza.

REFERENCES

- [1] D. Jurafsky and J. H. Martin, "Speech and language processing: An introduction to speech recognition, computational linguistics and natural language processing," *Upper Saddle River, NJ: Prentice Hall*, 2008.
- [2] D. Litman, "Natural language processing for enhancing teaching and learning," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016, pp. 4170–4176.
- [3] J. Burstein, J. Sabatini, J. Shore, B. Moulder, and J. Lentini, "A user study: Technology to increase teachers' linguistic awareness to improve instructional language support for english language learners," in *Proceedings of the 2nd Workshop on Natural Language Processing for Improving Textual Accessibility (NLP4ITA)*. Atlanta, Georgia: Association for Computational Linguistics, 2013.
- [4] A. Katinskaia, J. Nouri, and R. Yangarber, "Revita: a language-learning platform at the intersection of its and call," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japón: European Language Resources Association (ELRA), 2018.

- [5] M. Agirrezabal, B. Altuna, L. Gil Vallejo, J. Goikoetxea, and I. Gonzalez Dios, "Creating vocabulary exercises through nlp," *Digital Humanities in the Nordic Countries. Proceedings, 2019*, 2019.
- [6] A. Fenogenova and E. Kuzmenko, "Automatic generation of lexical exercises," in *Proceedings of the Workshop on Computational Linguistics and Language Science*, 2016.
- [7] E. Smith and R. Senter, "Automated readability index," Aerospace Medical Research Laboratories AMRL-TR-66-22, Tech. Rep., 1967.
- [8] W. Coster and D. Kauchak, "Simple english wikipedia: a new text simplification task," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short papers-Volume 2*. Association for Computational Linguistics, 2011, pp. 665–669.
- [9] W. Xu, C. Callison-Burch, and C. Napoles, "Problems in current text simplification research: New data can help," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 283–297, 2015.
- [10] M. Shardlow, "A survey of automated text simplification," *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 1, pp. 58–70, 2014.
- [11] D. Flickinger, "Using a grammar implementation to teach writing skills," in *Workshop on Technology Enhanced Learning, GWC 2018, Singapore*, 2018.
- [12] L. Morgado da Costa, "TELL@NTU: Intelligent tutoring and error detection systems, bringing language teaching and learning to a technological enhanced reality," in *Workshop on Technology Enhanced Learning, GWC 2018, Singapore*, 2018.
- [13] T.-H. Kao, Y.-W. Chang, H.-W. Chiu, T.-H. Yen, J. Boisson, J.-C. Wu, and J. S. Chang, "Conll-2013 shared task: Grammatical error correction ntu system description," in *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, 2013, pp. 20–25.
- [14] M. O. Dzikovska, R. D. Nielsen, C. Brew, C. Leacock, D. Giampiccolo, L. Bentivogli, P. Clark, I. Dagan, and H. T. Dang, "Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge," NORTH TEXAS STATE UNIV DENTON, Tech. Rep., 2013.
- [15] B. Settles, C. Brust, E. Gustafson, M. Hagiwara, and N. Madnani, "Second language acquisition modeling," in *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, 2018, pp. 56–65.
- [16] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.
- [17] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.
- [18] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55–60.
- [19] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [20] M. Lesk, "Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone," in *SIGDOC '86: Proceedings of the 5th annual international conference on Systems documentation*. New York, NY, USA: ACM, 1986.