

Creación de distractores para preguntas de opción múltiple mediante técnicas de incrustación (Word Embedding)

Creation of Distractors for Multiple Choice Questions Using Embedding Techniques (Word Embedding)

Jammil Israel Ramos Alvarez¹ <https://orcid.org/0009-0006-2828-2857>,
Rommel Vicente Torres¹ <https://orcid.org/0000-0003-2313-0118>, Verónica Segarra Faggioni¹
<https://orcid.org/0000-0002-7275-7411>, Eliana Ivanova Pinza Tapia¹ <https://orcid.org/0000-0002-9717-0721>

¹Universidad Técnica Particular de Loja, Loja, Ecuador
jiramos@utpl.edu.ec, rovitor@utpl.edu.ec,
vasegarra@utpl.edu.ec, eipinza@utpl.edu.ec



Esta obra está bajo una licencia internacional
Creative Commons Atribución-NoComercial 4.0.

Enviado: 2024/04/12

Aceptado: 2024/12/23

Publicado: 2024/12/30

Resumen

La generación automática de distractores para preguntas de opción múltiple en documentos académicos se basa en técnicas avanzadas de incrustación de palabras, como Word Embedding. Estas técnicas representan palabras como vectores en un espacio semántico, lo que facilita encontrar relaciones y similitudes entre ellas. A través de estos métodos, se pueden crear distractores plausibles pero incorrectos, que reflejan conceptos erróneos comunes o relacionados semánticamente con la pregunta. Este enfoque ha mejorado la calidad de las evaluaciones al proporcionar opciones de respuesta más desafiantes y realistas. Además, ha automatizado el proceso de creación de cuestionarios, ahorrando tiempo a los educadores y garantizando una mayor coherencia. Sin embargo, es necesario vigilar y ajustar este proceso para evitar respuestas incorrectas poco plausibles o sesgos implícitos. Esta innovadora metodología está transformando la forma en que se diseñan y administran las evaluaciones en entornos educativos y en documentos académicos como guías, cuestionarios y evaluaciones escritas. En este artículo, presentamos una introducción al tema, seguida de una revisión bibliográfica exhaustiva. A continuación, describimos en detalle los métodos y algoritmos empleados. Los resultados obtenidos a partir de la experimentación son analizados y discutidos, culminando en conclusiones sólidas y recomendaciones prácticas.

Palabras clave: Distractores, preguntas de opción múltiple, técnicas de incrustación, documentos académicos, palabras.

Sumario: Introducción, Revisión Literaria, Materiales y Métodos, Resultados y Discusión, Conclusiones.

Como citar: Ramos, J., Torres, R., Segarra, V. & Pinza, E. (2024). Creación de distractores para preguntas de opción múltiple mediante técnicas de incrustación (Word Embedding). *Revista Tecnológica - Espol*, 36(2), 95-110. <https://rte.espol.edu.ec/index.php/tecnologica/article/view/1161>

Abstract

The automatic generation of distractors for multiple-choice questions in academic documents relies on advanced word embedding techniques, such as Word2Vec. These techniques represent words as vectors in a semantic space, facilitating the discovery of relationships and similarities between them. Through these methods, plausible but incorrect distractors can be created that reflect common misconceptions or are semantically related to the question. This approach has enhanced the quality of assessments by providing more challenging and realistic answer options. Additionally, it has automated the process of creating quizzes, saving time for educators and ensuring greater consistency. However, it is necessary to monitor and adjust this process to avoid implausible incorrect answers or implicit biases. This innovative methodology is transforming the way assessments are designed and administered in educational settings and in academic documents such as guides, quizzes, and written evaluations. In this paper, we present an introduction to the topic, followed by an exhaustive literature review. Subsequently, we describe in detail the methods and algorithms employed. The results obtained from experimentation are analysed and discussed, culminating in robust conclusions and practical recommendations.

Keywords: Distractors, multiple-choice questions, embedding techniques, academic documents, works.

Introducción

La evaluación del aprendizaje es un componente fundamental en el ámbito educativo. Los exámenes de opción múltiple son una herramienta ampliamente utilizada para este propósito, ya que ofrecen una forma eficiente de evaluar el conocimiento de los estudiantes sobre un tema determinado. Sin embargo, la creación de preguntas de opción múltiple de alta calidad puede ser un proceso laborioso y lento para los profesores. Una de las principales dificultades radica en la generación de distractores que sean plausibles y discriminativos, es decir, que puedan diferenciar entre los estudiantes que dominan el tema y aquellos que no (Liu et al., 2023).

Según Jurado Núñez et al. (2013), las técnicas de incrustación (Word Embedding) han demostrado ser una herramienta poderosa para el procesamiento del lenguaje natural, ya que permiten representar las palabras como vectores numéricos que capturan su significado semántico.

Jurado Núñez et al. (2013) mencionan que la técnica de incrustación de palabras es una herramienta clave en el aprendizaje no supervisado, capaz de capturar el contexto, la similitud semántica y sintáctica, así como la relación entre palabras. En este trabajo, se presenta una contribución única en la aplicación de estas técnicas para la generación automática de distractores en preguntas de opción múltiple a partir de documentos académicos, considerado como el conjunto de datos (evaluaciones, libros, guías educativas, etc.). La propuesta se enfoca en utilizar la información semántica de las incrustaciones de palabras para crear distractores relevantes al tema de la pregunta y que sean difíciles de distinguir de la respuesta correcta. Se evalúan tres algoritmos ampliamente utilizados que permitan predecir distractores de respuesta a preguntas de opción múltiple. Modelos como Word2Vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014) y FastText (Bojanowski et al., 2019) son ejemplos destacados de enfoques que explotan el contexto de palabras en textos específicos.

A diferencia de otros estudios previos, esta investigación realizó una evaluación semántica y sintáctica del lenguaje por medio de los algoritmos descritos en el párrafo anterior.

Se analizó sus enfoques, ventajas y limitaciones, proporcionando comparativas que resaltan su impacto en tareas de Procesamiento de Lenguaje Natural (NLP) y aplicaciones del mundo real.

A continuación, se presenta una estructura general del trabajo:

- Revisión de la literatura sobre las técnicas de incrustación y su aplicación en el ámbito educativo.
- Materiales y métodos para seleccionar dos técnicas de incrustación de palabras, y su posterior desarrollo de un entorno web para la generación automática de distractores a partir de un conjunto de datos.
- Evaluación de la eficacia del método propuesto mediante experimentos con diferentes tipos de documentos académicos.
- Discusión de los resultados obtenidos y las posibles aplicaciones del método en el contexto educativo.
- Conclusiones y recomendaciones para futuras investigaciones.
- Materiales y Métodos.

Revisión Literaria

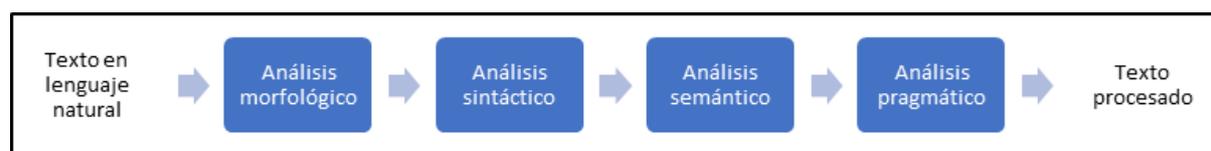
Procesamiento de Lenguaje Natural (PLN)

El Procesamiento del Lenguaje Natural (PLN), también conocido como Natural Language Processing (NLP) en inglés, es un campo de estudio dentro de la Inteligencia Artificial (IA) que se centra en la interacción entre las computadoras y el lenguaje humano. Su objetivo principal es dotar a las máquinas de la capacidad de entender, procesar y generar lenguaje natural de forma similar a como lo hacen los humanos (Manjarrés-Betancur & Echeverri-Torres, 2020).

El PLN, como se ve en la Figura 1, contempla cuatro tipos de análisis sobre un texto en lenguaje natural:

Figura 1

Procesamiento de lenguaje natural PLN



Técnicas de incrustación (Word Embedding)

Según Talamé et al. (2022), las técnicas de incrustación (Word Embedding) son un conjunto de métodos que permiten representar las palabras como vectores numéricos. Estos vectores capturan el significado semántico de las palabras al codificar las relaciones entre ellas en un espacio vectorial.

Las técnicas de incrustación se han utilizado con éxito en una amplia gama de tareas de procesamiento del lenguaje natural, como la traducción automática, la clasificación de textos, la detección de emociones y la generación de lenguaje natural.

Existen diferentes métodos para obtener incrustaciones de palabras, por ejemplo, Word2Vec, GloVe y FastText. Estos métodos se basan en el aprendizaje automático para analizar grandes cantidades de texto y determinar las relaciones entre las palabras.

Las incrustaciones de palabras tienen una amplia variedad de aplicaciones en el procesamiento del lenguaje natural, tales como:

- Análisis de sentimientos
- Traducción automática
- Recuperación de información
- Generación de texto

Word2Vec

En el trabajo de Mikolov et al. (2013) se define a word2vec como una arquitectura de red neuronal de dos capas que aprende representaciones vectoriales (embeddings) de palabras. Su objetivo es capturar relaciones semánticas entre palabras, entrenando en un corpus grande de texto. Los dos enfoques principales dentro de word2vec se lo puede observar en la Figura 2.

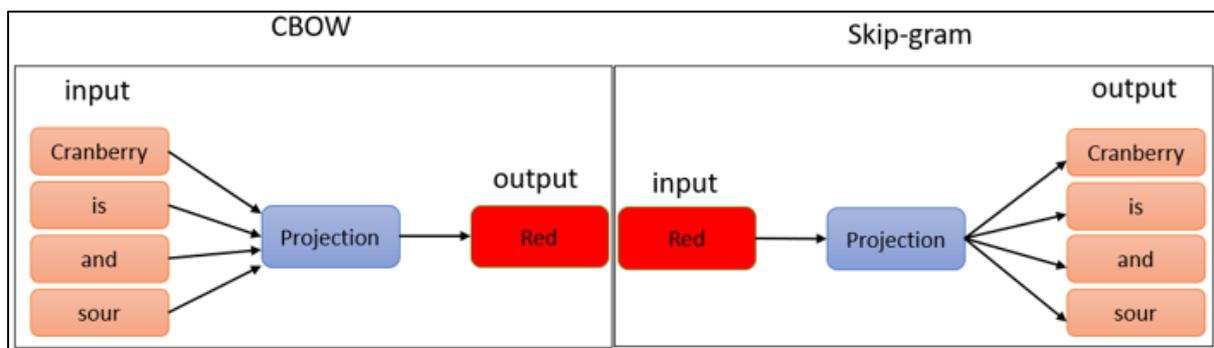
- Bag-of-Words continuo (CBOW): Dado un centro de palabra, el modelo predice las palabras que lo rodean dentro de una ventana de contexto.
- Skip-gram: Dado un entorno de palabra, el modelo predice la palabra central.

Algunos puntos clave:

- Aprende embeddings de palabras que capturan relaciones semánticas.
- Ofrece dos enfoques de entrenamiento: CBOW y skip-gram.
- Se utiliza ampliamente para diversas tareas de procesamiento del lenguaje natural.

Figura 2

Word2Vec con CBOW y Skip-gram

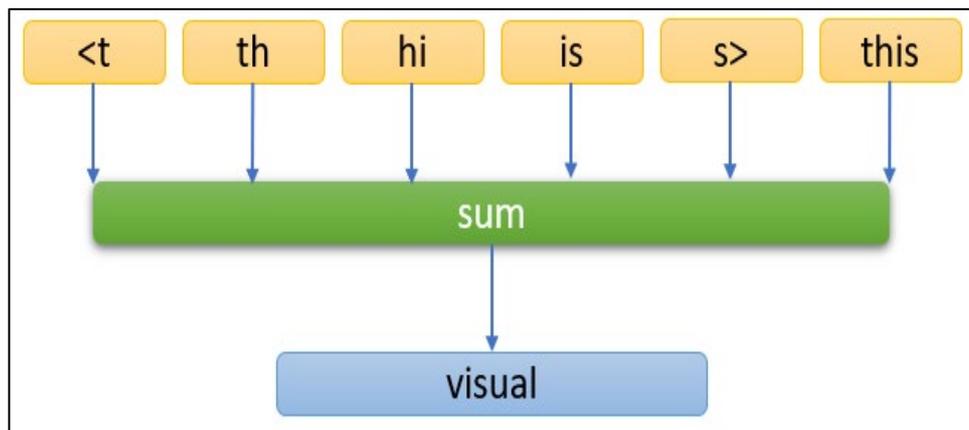


FastText

De acuerdo con Bojanowski et al. (2019), fastText es una extensión de word2vec que aborda el problema de las palabras fuera del vocabulario (OOV) al considerar información de subpalabras (n-gramas de caracteres) además de palabras completas. Esto le permite representar palabras conocidas y desconocidas de manera más efectiva (Véase Figura 3).

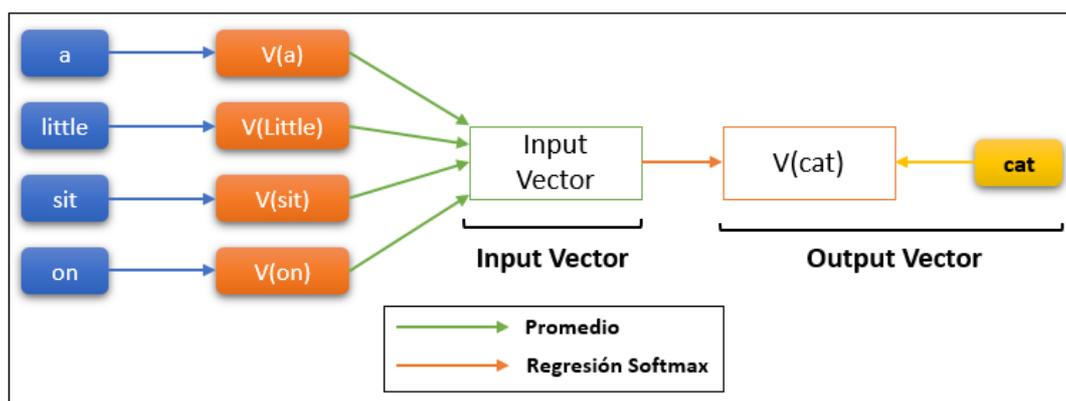
Algunos puntos clave:

- Maneja palabras OOV usando información de subpalabras (n-gramas de caracteres).
- Mejora la representación de palabras raras y morfológicamente complejas.
- Adecuado para idiomas con morfología rica y recursos de vocabulario limitados.

Figura 3*Algoritmo FastText***Glove**

GloVe (Global Vectors for Word Representation), según Pennington et al. (2014), es un método estadístico para aprender embeddings de palabras que aprovecha las estadísticas de coocurrencia de un corpus de texto grande. Toma en cuenta tanto el contexto local (palabras circundantes) como los patrones de uso global de las palabras, así se muestra en la Figura 4. Algunos puntos clave son:

- Utiliza estadísticas de coocurrencia para capturar el significado de las palabras.
- Considera tanto el contexto local como global para obtener representaciones más completas.
- A menudo funciona bien en tareas como analogía y similitud de palabras.

Figura 4*Algoritmo GloVe***Conjunto de datos**

Un conjunto de datos es una colección organizada de información, generalmente representada en forma tabular que puede incluir números, texto, imágenes, sonido u otros tipos de datos. Cada fila en el conjunto de datos representa una observación o instancia, y cada columna representa una variable o atributo específico asociado con esa observación. Los conjuntos de datos se utilizan en diversos campos como la estadística, la ciencia de datos, la informática y la investigación, para realizar análisis, modelado y toma de decisiones basada en datos. Los conjuntos de datos word2vec-google-news-300 (Google, 2024) y cc.en.300.bin

(Facebook OS, 2024) se seleccionaron por su robustez y confiabilidad en tareas de procesamiento del lenguaje natural (PLN). Word2vec-google-news-300, entrenado por Google, ofrece representaciones densas de palabras que capturan contextos semánticos profundos, mientras que cc.en.300.bin de Facebook (FastText) extiende la cobertura al incluir subpalabras, mejorando la representación de términos fuera del vocabulario.

Word2vec-google-news-300, utilizado para el algoritmo Word2Vec, es un modelo de incrustación de palabras preentrenado desarrollado por Google, diseñado específicamente para capturar relaciones semánticas entre palabras en un espacio vectorial de alta dimensión. El modelo "word2vec", abreviatura de "word to vector" (palabra a vector), es una de las arquitecturas pioneras en el campo del procesamiento del lenguaje natural (PLN) para generar incrustaciones de palabras.

Cc.en.300.bin, utilizado en el algoritmo FastText, es un archivo binario relacionado con el procesamiento del lenguaje natural, un modelo de incrustación de palabras que contiene vectores de palabras que capturan relaciones semánticas entre estas basadas en el contexto en el que aparecen en los datos de entrenamiento.

Es importante considerar que este trabajo también contempla la generación de un conjunto de datos aportado por el usuario, que se compone de documentos académicos como evaluaciones, guías de estudio, libros, etc. Este conjunto se utilizará para entrenar un modelo de aprendizaje automático.

Preguntas de opción múltiple

Las preguntas de opción múltiple en documentos académicos son una herramienta de evaluación que presenta al estudiante una pregunta o enunciado seguido de un conjunto de opciones de respuesta. Solo una de las opciones es la respuesta correcta, mientras que las demás son distractores, que pueden ser plausibles o no.

Trabajos relacionados

En la búsqueda de trabajos relacionados, se han definido dos preguntas principales de investigación:

- Pregunta 1 ¿Cuáles son las mejores técnicas de incrustación de palabras?
- Pregunta 2 ¿Cuál es el mejor algoritmo de entrenamiento en documentos de evaluación académica?

Para dar respuesta a estas preguntas, se han analizado y seleccionado veinte trabajos relacionados, la mayoría emplea herramientas como el entrenamiento de datos, técnicas de incrustación de palabras y la generación de datos a partir de un conjunto de datos. Estas herramientas permiten evaluar las palabras dentro de un documento o conjunto de documentos, analizando su similitud tanto semántica como sintáctica. Se ha determinado que, en la mayoría de los trabajos revisados, los algoritmos más comúnmente aplicados son:

- FastText
- Word2Vec
- GloVe

Algunos trabajos que sobresalen del estudio son:

- Automatic generation of multiple-choice items for prepositions based on Word2vec (Xiao et al., 2018).
- Automatic Chinese Multiple-Choice Question Generation for Human Resource Performance Appraisal (Quan et al., 2018).
- Ranking Multiple Choice Question Distractors using Semantically Informed Neural Networks (Sinha et al., 2020).
- Generating Adequate Distractors for Multiple-Choice Questions (Zhang et al., 2020).

Materiales y Métodos

Esta sección evaluó y desarrolló una solución, que empleó representaciones vectoriales de palabras como base para llevar a cabo la generación de distractores para evaluaciones de opción múltiple. Como se exploró en el presente trabajo, los algoritmos Word2Vec, FastText y GloVe fueron evaluados para seleccionar los dos más eficientes y considerarlos en el desarrollo de la aplicación web.

Comparación

De acuerdo a la literatura revisada, se realizó una comparación general de las técnicas de incrustación en términos generales que se expone en la Tabla 1.

Tabla 1

Características de algoritmos

Característica	Word2Vec	FastText	GloVe
Maneja palabras fuera de su vocabulario	No	Sí	No (pero puede generalizar)
Manejo morfológico de las palabras	Limitado	Mejor	Limitado (depende del corpus)
Velocidad de entrenamiento de palabras	Más rápido	Más lento	Más lento
Rendimiento en analogía	Bueno	Bueno	Bueno
Rendimiento en similitud semántica	Bueno	Bueno	Bueno

Evaluación de Algoritmos

Para evaluar los algoritmos, se analizaron en profundidad los aspectos, particularidades y métricas más relevantes identificadas en recientes investigaciones sobre el tema.

Aspectos generales

La Tabla 2 presenta otras características de los algoritmos analizadas en la literatura.

Tabla 2

Aspectos generales de los algoritmos

Característica	Word2Vec	FastText	GloVe
Compañía Desarrolladora	Facebook Research	Google	Stanford University
Año	2015	2013	2014
Representación	Vectores	Vectores	Vectores
Estructura de sub-palabras	Sí	No	No
Capacidad para manejar palabras desconocidas	Sí	No	No
Operaciones algebraicas con palabras	Sí	Sí	No
Complejidad computacional y tiempo de entrenamiento	Alta	Alta	Alta
Palabras fuera del vocabulario	Si	No	No

Similitud semántica

- Objetivo: Determinar el grado de correlación entre la similitud de pares de palabras y la similitud de sus vectores en las incrustaciones, como se detalla en la Tabla 3 y la Figura 5.
- Conjuntos de datos: WordSim353 y SimLex-999.
- Interpretación:
 - Alta similitud semántica con resultado cercano a 1.
 - Baja similitud semántica con resultado cercano a 0.
- Cálculo: Coseno del ángulo entre los vectores de las palabras. Fórmula:

$$\cos(\theta) = (v1 * v2) / \|v1\| \|v2\|$$

Donde:

$v1$ y $v2$ son los vectores de las dos palabras.
 * fue el producto escalar.

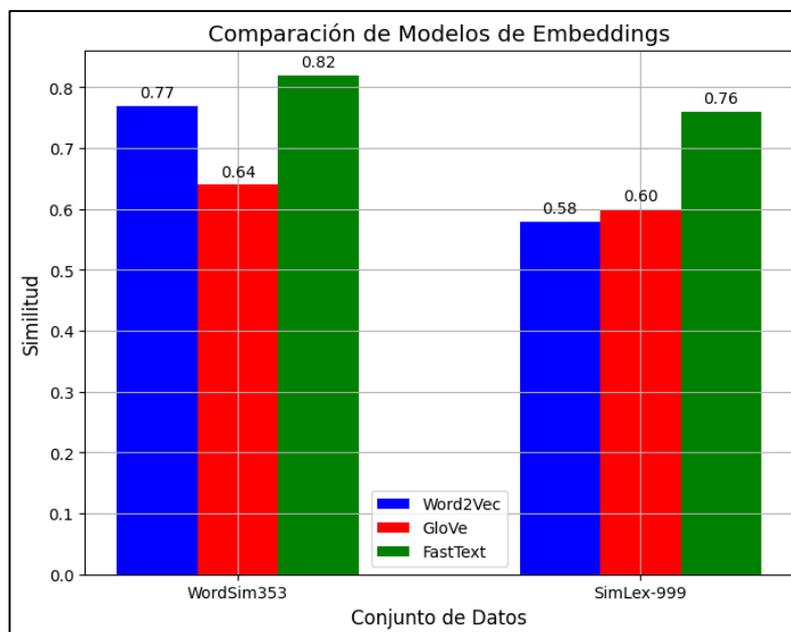
Tabla 3

Similitud semántica

Conjunto de datos	Word2Vec	GloVe	FastText
WordSim353 ¹	0.77	0.64	0.82
SimLex-999 ²	0.58	0.60	0.76

Figura 5

Similitud semántica



¹ Tomado de <https://acortar.link/wdqzJG>

² Tomado de <https://acortar.link/vSixxz>

Analogías de palabras

- Objetivo: Encontrar relaciones semánticas entre palabras, según los resultados en la Tabla 4. La idea fue completar analogías de la siguiente forma:
- A es a B como C es a D
- Conjuntos de datos: Google Analogy y Microsoft Research Paraphrase (MSR).
- Cálculo: Se calculó la similitud coseno entre las representaciones vectoriales de A-B y C-D.

$$A : B :: C : D$$

Donde:

A,B,C,D son palabras.

Tabla 4

Analogía de palabras

Algoritmo	Método	Analogía 1			Analogía 2
		Palabra	Palabra	Palabra	Palabra resultado
Word2Vec	Google Analogs ³	education teacher intelligence	learning student creativity	reading coach artist	read, knowledge Head_Coach, athletic_director painter, canvas
FastText		education teacher intelligence	learning student creativity	reading coach artist	knowledge, re-reading athletic, coaches canvas, artists
Glove		education teacher intelligence	learning student creativity	reading coach artist	read, listening coaches, team artists, painter
Word2Vec	MSR ⁴	education teacher intelligence	learning student creativity	reading coach artist	reading_aloud Head_Coach artists
FastText		education teacher intelligence	learning student creativity	reading coach artist	re-reading student-athlete artists
Glove		education teacher intelligence	learning student creativity	reading coach artist	read team artists

Analogía semántica

- Objetivo: Evaluar la capacidad de un modelo de lenguaje para comprender las relaciones semánticas entre palabras. El detalle en la Tabla 5.
- Conjuntos de datos: 3CosAdd y 3CosMul.
- Interpretación: Las fórmulas 3CosAdd y 3CosMul calcularon la similitud entre dos palabras (a y b) como el coseno del ángulo entre sus vectores de representación (v_a , v_b y v_c).
- Cálculo: Fueron necesario vectores de representación de las palabras. Su fórmula fue:

$$\text{SimAdd}(a,b) = \cos(v_a) * \cos(v_b) + \cos(v_a) * \cos(v_c) + \cos(v_b) * \cos(v_c)$$

$$\text{SimMul}(a,b) = \cos(v_a) * \cos(v_b) * \cos(v_c)$$

³ Tomado de <https://arxiv.org/pdf/1301.3781.pdf>

⁴ Tomado de <https://www.microsoft.com/en-us/download/details.aspx?id=52398>

Donde:

va, vb y vc fueron los vectores de las dos palabras.

Tabla 5

Analogía semántica

Algoritmo	Método	Analogía 1		Analogía 2		
		Palabra	Palabra	Palabra	Analogía	Semántica
Word2Vec	3CosAdd	man	woman	king	king queen monarch princess crown prince	0.84 0.73 0.65 0.62 0.58
FastText		man	woman	king	king queen monarch throne princess	0.80 0.69 0.56 0.56 0.55
Glove		man	woman	king	king queen queen-mother king- queen-consort	0.82 0.76 0.69 0.68 0.66
Word2Vec	3CosMul	man	woman	king	queen monarch princess Queen Consort queens	0.93 0.86 0.85 0.82 0.81
FastText		man	woman	king	queen princess throne monarch elizabeth	0.92 0.84 0.83 0.82 0.80
Glove		man	woman	king	queen queen-mother king- queen-consort child-king	0.94 0.91 0.89 0.88 0.87

Selección de algoritmos

Según la evaluación realizada en el punto anterior, Word2Vec y FastText resultaron ser los algoritmos más eficientes para generar distractores.

FastText (2015) sobresalió por su facilidad de uso y capacidad para manejar palabras fuera del vocabulario entrenado. Ofreció mejores resultados en tareas de similitudes semánticas y analogías, adaptándose de manera más efectiva al contexto del estudio. Su principal ventaja radicó en la generación de resultados con palabras no presentes en el corpus original.

Word2Vec (2013), a pesar de ser más antiguo, capturó de manera más efectiva relaciones semánticas y permitió calcular la similitud entre palabras. Alcanzó un alto porcentaje de similitud en la generación de distractores y demostró ser computacionalmente eficiente.

En comparación, FastText superó a GloVe en la capacidad de generalizar palabras OOV, gracias a su habilidad para crear vectores a partir de subpalabras. Aunque Word2Vec no fue tan eficaz en este aspecto, demostró una mayor capacidad de generalización que GloVe cuando se entrenaba adecuadamente. En términos de eficiencia computacional, Word2Vec resultó más rápido que GloVe, que requiere calcular y factorizar una gran matriz de co-ocurrencias. Además, Word2Vec capturó mejor las relaciones semánticas y sintácticas, mientras que FastText destacó en el tratamiento de la morfología y las palabras raras, mostrando mayor flexibilidad en idiomas o dominios especializados.

Resultados y Discusión

En el presente estudio se exploró la aplicación de los algoritmos de incrustación de palabras Word2Vec y FastText para generar distractores. Estos distractores se basaron en la proximidad de las palabras en el espacio de incrustación, considerando una pregunta y una respuesta correcta como punto de partida. Algunos aspectos a considerar fueron:

- Área de estudio: Se seleccionaron como área de estudio diversos documentos académicos, como evaluaciones, guías de estudio, entre otros.
- Idioma: El idioma utilizado fue el inglés.
- Implementación: La implementación de los algoritmos y el diseño del sistema se realizó en Python.
- Conjunto de datos: Se utilizó cc.en.300.bin para FastText y word2vec-google-news-300 en Word2Vec. También se permitió la opción de utilizar datos ingresados por el usuario para realizar la experimentación.
- Cantidad de experimentos: Se llevaron a cabo veinte experimentos, de los cuales se consignan tres en el presente trabajo.

Diseño de Experimentación

Se desarrolló un generador de distractores con los modelos Word2Vec y FastText para preguntas de opción múltiple, como se ve en la Figura 6.

Figura 6

Generador de distractores

The screenshot shows a web application titled "Generador de Distractores". It features a blue header with the title. Below the header, there are three main columns: "Pregunta", "Respuesta Correcta", and "Distractores". The "Pregunta" column contains a text area for input. The "Respuesta Correcta" column contains a text input field. The "Distractores" column contains a list of generated distractors, with a trash icon at the top right. In the center, there is a "Modelo" section with two radio buttons: "fasttext" and "word2vec". At the bottom left, there is a file upload section with a "Seleccionar archivo" button and a text input for "Archivo para entrenar el modelo (opcional)". A "Generar" button is located at the bottom center.

Funcionamiento del generador

El usuario ingresó una pregunta y la respuesta correcta. El usuario pudo elegir un modelo de lenguaje para generar los distractores. Así también, pudo cargar un archivo de texto para entrenar el modelo de lenguaje. El generador devolvió una serie de distractores para la pregunta.

Modelos de lenguaje

De acuerdo a la experimentación realizada, Word2Vec y FastText son dos modelos de lenguaje que se pueden utilizar para generar distractores. FastText es generalmente más preciso que Word2Vec, pero Word2Vec es más rápido en general.

Word2Vec genera distractores utilizando las siguientes técnicas: coseno similar y vecinos más cercanos. En cambio, FastText genera distractores empleando las siguientes técnicas: predicción de la siguiente palabra y búsqueda de vecinos.

Evaluación de la calidad

La calidad de los distractores generados por Word2Vec y FastText se evaluó utilizando las siguientes técnicas:

- Análisis por expertos: Grupo de expertos evaluó la calidad de los distractores.
- Análisis estadístico: Se utilizaron técnicas estadísticas para evaluar distractores.

Resultados

Es importante considerar que los resultados obtenidos mostraron palabras (con letras mayúsculas o con errores de tipeo) que presentaron un mayor porcentaje de similitud con la respuesta correcta y que se encontraron dentro del conjunto de datos evaluado. Los resultados de las pruebas realizadas con el generador de distractores se muestran en la Tabla 5, Tabla 6, Tabla 7, Tabla 8; así como en la Figura 7, Figura 8 y Figura 9:

Tabla 6

Ejemplo 1

Pregunta	What is the main purpose of education?			
Respuesta correcta	Knowledge			
Algoritmo	FastText		Word2Vec	
Conjunto de Datos	cc.en.300.bin	Datos Usuario	W2V-google-news-300	Datos Usuario
Distractor	knowlege - 0.81	performance - 0.75	knowlege - 0.65	one - 0.83
	knowldege - 0.76	relationships - 0.81	expertise - 0.65	following - 0.81
	knowldge - 0.75	educators - 0.76	understanding - 0.63	worry - 0.81

Figura 7

Ejemplo 1

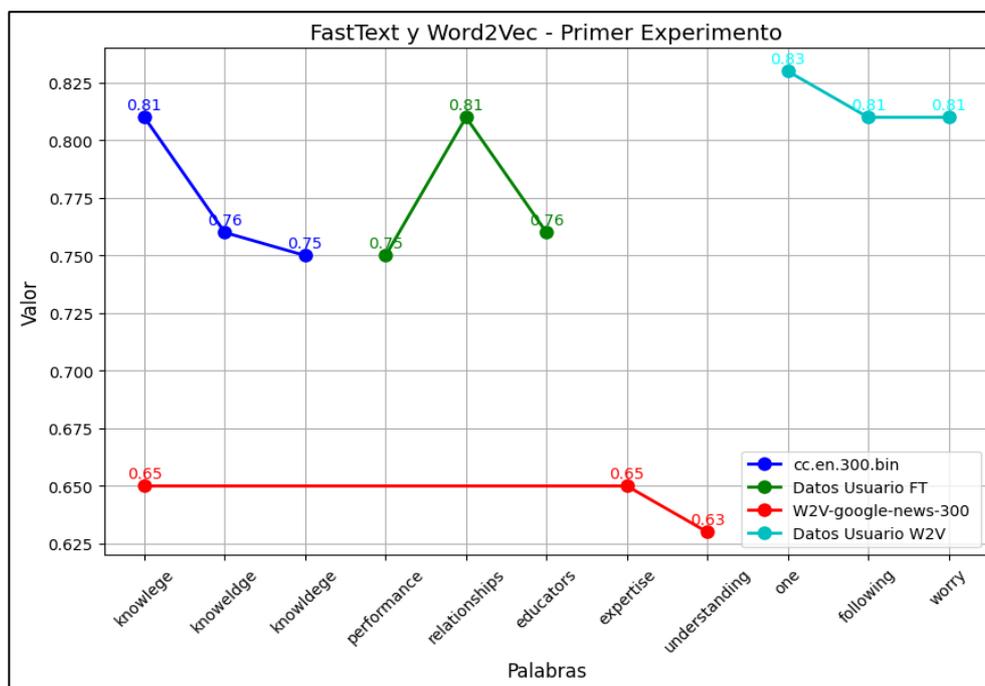


Tabla 7
Ejemplo 2

Pregunta	What can convey a wealth of meaning in an instant?			
Respuesta correcta	Pictures			
Algoritmo	FastText		Word2Vec	
Conjunto de Datos	cc.en.300.bin	Datos Usuario	W2V-google-news-300	Datos Usuario
Distractor	Photos - 0.90	successfully - 0.75	Photos - 0.91	abilities - 0.35
	Picutres - 0.81	concepts - 0.73	Photographs - 0.88	pupils - 0.33
	compete - 0.32	Pictues - 0.80	single - 0.72	Images - 0.79

Figura 8
Ejemplo 2

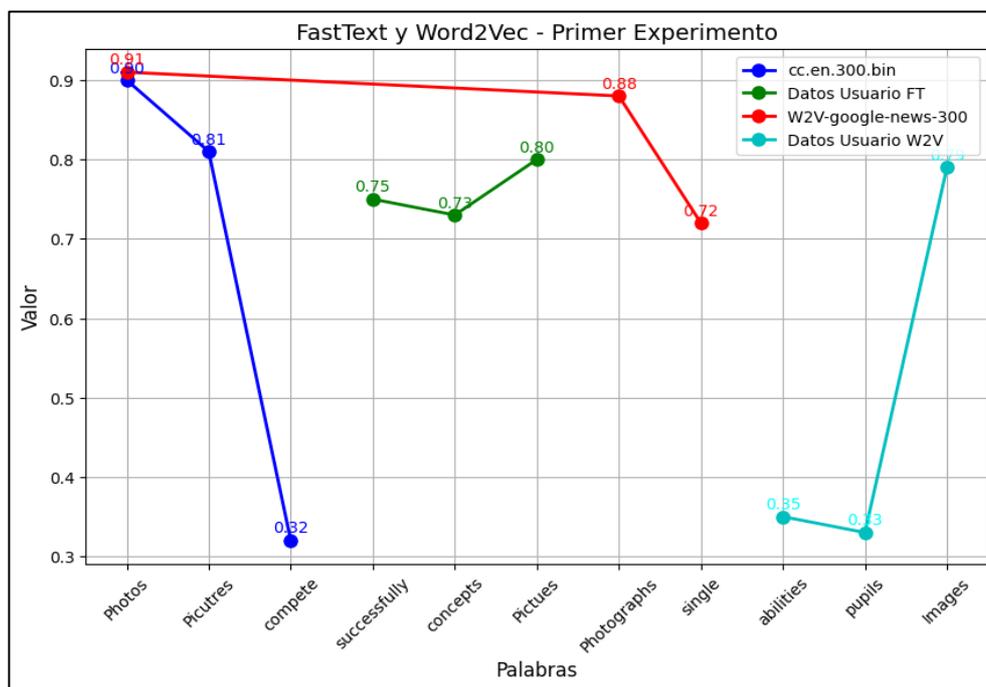
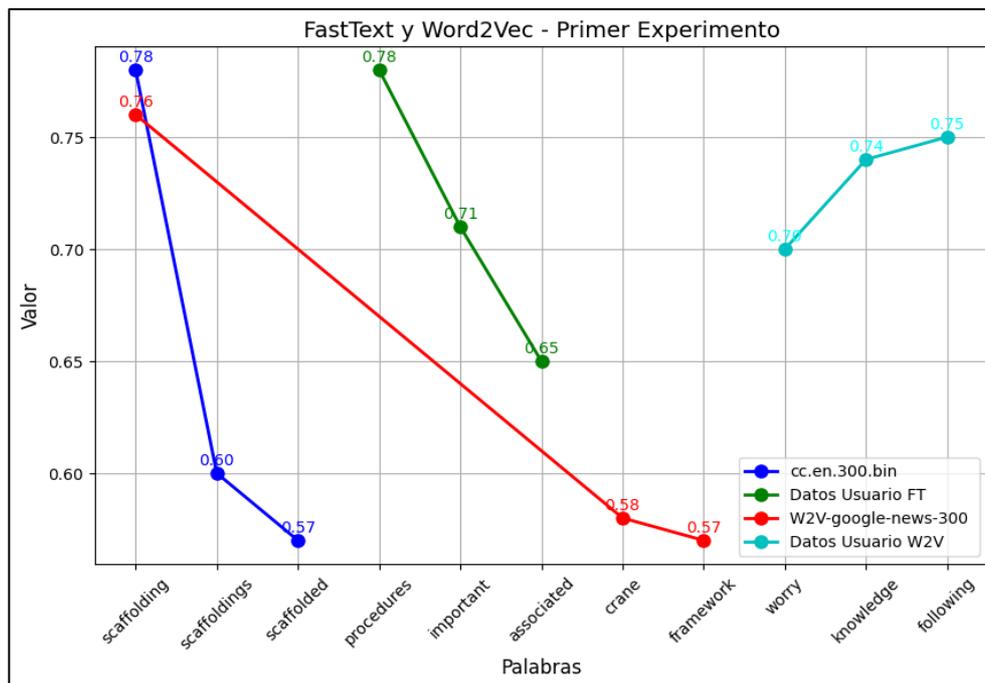


Tabla 8
Ejemplo 3

Pregunta	What term is related to frameworks that support the learning process?			
Respuesta correcta	Scaffold			
Algoritmo	FastText		Word2Vec	
Conjunto de Datos	cc.en.300.bin	Datos Usuario	W2V-google-news-300	Datos Usuario
Distractor	scaffolding - 0.78	procedures - 0.78	scaffolding - 0.76	worry - 0.70
	scaffoldings - 0.60	important - 0.71	crane - 0.58	knowledge - 0.74
	scaffolded - 0.57	associated - 0.65	framework - 0.57	following - 0.75

Figura 9

Ejemplo 3



Discusión de resultados

En la Tabla 5, se identifica una tarea que consistió en seleccionar la palabra relacionada con la educación de entre un conjunto de palabras distractoras. La palabra más relevante para "education" fue "knowledge", tanto para FastText (0,81) como para Word2Vec (0,83). Otras palabras, como "relationships" (0,81) y "expertise" (0,65), también estuvieron relacionadas con la educación, pero no de manera tan fuerte como "knowledge". Esto demostró cómo se puede utilizar un modelo de aprendizaje automático para identificar palabras asociadas con conceptos como "education", comparándolas con otras palabras de un gran conjunto de datos.

En la Tabla 6, se exponen los resultados de la generación de distractores para la pregunta "What can convey a wealth of meaning in an instant?". Los distractores con las puntuaciones más altas fueron "Photos" (0,90 en FastText y 0,91 en Word2Vec), lo que sugiere que el modelo fue efectivo en la generación de distractores. Sin embargo, también se generó un distractor ("compete") que no está relacionado con la pregunta, lo que evidenció que el modelo puede producir distractores irrelevantes.

Finalmente, en la Tabla 7, se presentan los resultados para una pregunta sobre marcos que apoyan el proceso de aprendizaje. En Word2Vec, la palabra que más distrajo fue "scaffolding", con una similitud coseno de 0,76. Otros distractores incluyeron "following" (0,75), "important" (0,71) y "knowledge" (0,74). De manera similar, en FastText, "scaffolding" fue el distractor más relevante (0,78), seguido de "procedures" (0,78) y "associated" (0,65).

Conclusiones

Según la experimentación realizada, la efectividad de FastText y Word2Vec depende principalmente de la calidad y amplitud de los datos con los que se entrenan. Cuanto más completo sea el conjunto de datos, mejor podrán estos modelos captar el significado del lenguaje.

De los resultados obtenidos, los distractores formados por respuestas de una sola palabra suelen ser más efectivos, ya que permiten generar opciones que se parecen más a la respuesta correcta.

FastText es útil para tareas que requieren una comprensión detallada del lenguaje, gracias a su capacidad para analizar subpalabras. Por otro lado, Word2Vec es más adecuado para escenarios con grandes volúmenes de datos y donde se necesita una capacitación eficiente.

Aunque se han logrado avances, la generación automática de distractores sigue siendo un reto importante, con problemas en cuanto a precisión y la capacidad de los sistemas para generar y clasificar distractores de forma adecuada.

El uso de algoritmos de procesamiento de lenguaje natural facilita la creación de distractores, lo que puede ser una herramienta útil para educadores, ayudándoles a ahorrar tiempo y esfuerzo en la preparación de evaluaciones.

Reconocimientos

Los autores declaran la contribución y participación equitativa de roles de autoría para esta publicación.

Referencias

- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2019). *Enriching Word Vectors with Subword Information*. *Transactions of the Association for Computational Linguistics*, 7, 135-146. https://doi.org/10.1162/tacl_a_00051
- Facebook OS. (2024). *Word vectors for 157 languages · fastText*. <https://fasttext.cc/docs/en/crawl-vectors.html>
- Google. (2024). *Google Code Archive - Long-term storage for Google Code Project Hosting*. <https://code.google.com/archive/p/word2vec/>
- Jurado Núñez, A., Flores Hernández, F., Delgado Maldonado, L., Sommer Cervantes, H., Martínez González, A., & Sánchez Mendiola, M. (2013). *Distractores en preguntas de opción múltiple para estudiantes de medicina: ¿cuál es su comportamiento en un examen sumativo de altas consecuencias?* *Revista de Educación en Ciencias de la Salud*, 10(3), 173-185.
- Liu, Q., Wald, N., Daskon, C., & Harland, T. (2023). Multiple-choice questions (MCQs) for higher-order cognition: Perspectives of university teachers. *Innovations in Education and Teaching International*, 60(3), 314-323. <https://doi.org/10.1080/14703297.2023.2222715>
- Manjarrés-Betancur, R. A., & Echeverri-Torres, M. M. (2020). Asistente virtual académico utilizando tecnologías cognitivas de procesamiento de lenguaje natural. *Revista Politécnica*, 16(31), 85-95. <https://doi.org/10.33571/rpolitec.v16n31a7>
- Mikolov, T., Le, Q. V., & Sutskever, I. (2013). *Exploiting Similarities among Languages for Machine Translation*. <https://code.google.com/p/word2vec/>
- Pennington, J., Socher, R., & Manning, C. D. (2014). *GloVe: Global Vectors for Word Representation*. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, 1532-1543. <http://nlp.stanford.edu/pubs/glove.pdf>
- Quan, P., Shi, Y., Niu, L., Liu, Y., & Zhang, T. (2018). Automatic Chinese Multiple-Choice Question Generation for Human Resource Performance Appraisal. *Procedia Computer Science*, 139, 165-172. <https://doi.org/10.1016/j.procs.2018.10.235>

- Sinha, M., Dasgupta, T., & Mandav, J. (2020). Ranking Multiple Choice Question Distractors using Semantically Informed Neural Networks. *Proceedings of the 29th International Conference on Information and Knowledge Management (CIKM 2020)*, 3329-3332. <https://doi.org/10.1145/3340531.3417468>
- Talamé, M. L., Monge, A., Nicolás Amor, M., & Cardoso, C. (2022). *Creation of a corpus of embedded words from tweets generated in Argentina*. *Revista de Lingüística y Lenguas Aplicadas*, 17(2), 123-139. <http://revistas.upm.es/>.
- Xiao, W., Wang, M., Zhang, C., Tan, Y., & Chen, Z. (2018). Automatic generation of multiple-choice items for prepositions based on Word2vec. *Communications in Computer and Information Science*, 902, 81–95. https://doi.org/10.1007/978-981-13-2206-8_8
- Zhang, C., Sun, Y., Chen, H., & Wang, J. (2020). *Generating Adequate Distractors for Multiple-Choice Questions*. *arXiv Preprint*. <https://arxiv.org/abs/2010.12658>