UNIVERSIDAD DE COSTA RICA
SISTEMA DE ESTUDIOS DE POSGRADO

USO DE WORD EMBEDDINGS PARA LA DETECCIÓN AUTOMÁTICA DE TITULARES CLICKBAIT EN IDIOMA ESPAÑOL

Trabajo Final de Investigación Aplicada sometido a la consideración de la Comisión del Programa de Estudios de Posgrado en Computación e Informática para optar al grado y título de Maestría Profesional en Computación e Informática

MARIO CABRERA VEGA

Ciudad Universitaria Rodrigo Facio, Costa Rica

2021

‘‘Este trabajo final de investigación aplicada fue aceptado por la Comisión del Programa de Estudios de Posgrado en Computación e Informática de la Universidad de Costa Rica, como requisito parcial para optar al grado y título de Maestría Profesional en Computación e Informática.”

Firmado digitalmente por JORGE ANTONIO LEONI DE LEON (FIRMA)
Fecha: 2022.02.11 11:20:00 -06'00'

Dr. Jorge Antonio Leoni de León

Representante de la Decana Sistema de Estudios de Posgrado

EDGAR ENRIQUE CASASOLA MURILLO (FIRMA)

Firmado digitalmente por EDGAR ENRIQUE CASASOLA MURILLO (FIRMA)
Fecha: 2022.02.12 17:30:34 -06'00'

Dr. Edgar Casasola Murillo

Profesor Guía

GABRIELA MARIN RAVENTOS (FIRMA)

Firmado digitalmente por GABRIELA MARIN RAVENTOS (FIRMA)
Fecha: 2022.02.11 10:11:21 -06'00'

Dra. Gabriela Marín Raventós

Lectora

JOSE ANDRES GUEVARA COTO (FIRMA)
PERSONA FISICA, CPF-01-1278-0555.
Fecha declarada: 11/02/2022 03:54:34 PM
Esta es una representación gráfica únicamente, verifique la validez de la firma.

Dr. José Guevara Coto

Lector

KRYSCIA RAMIREZ BENAVIDES (FIRMA)

Digitally signed by KRYSCIA RAMIREZ BENAVIDES (FIRMA)
Date: 2022.02.11 11:56:03 -06'00'

Dra. Kryscia Ramírez Benavides

Representante del Posgrado en Computación e Informática

MARIO ALBERTO CABRERA VEGA (FIRMA)

Digitally signed by MARIO ALBERTO CABRERA VEGA (FIRMA)
Date: 2022.02.11 08:47:53 -06'00'

Mario Cabrera Vega

Sustentante

Índice

Índice de figuras

Índice de cuadros

UNIVERSIDAD DE COSTA RICA

SEP Sistema de Estudios de Posgrado

Autorización para digitalización y comunicación pública de Trabajos Finales de Graduación del Sistema de Estudios de Posgrado en el Repositorio Institucional de la Universidad de Costa Rica.

Yo, Mario Cabrera Vega, con cédula de identidad 1-1501-0981, en mi condición de autor del TFG titulado "Uso de *word embeddings* para la detección automática de titulares *clickbait* en idioma español"

Autorizo a la Universidad de Costa Rica para digitalizar y hacer divulgación pública de forma gratuita de dicho TFG a través del Repositorio Institucional u otro medio electrónico, para ser puesto a disposición del público según lo que establezca el Sistema de Estudios de Posgrado. **SI** [X] **NO** *[ ]

***En caso de la negativa favor indicar el tiempo de restricción: _____ año (s).**

Este Trabajo Final de Graduación será publicado en formato PDF, o en el formato que en el momento se establezca, de tal forma que el acceso al mismo sea libre, con el fin de permitir la consulta e impresión, pero no su modificación.

Manifiesto que mi Trabajo Final de Graduación fue debidamente subido al sistema digital Kerwá y su contenido corresponde al documento original que sirvió para la obtención de mi título, y que su información no infringe ni violenta ningún derecho a terceros. El TFG además cuenta con el visto bueno de mi Director (a) de Tesis o Tutor (a) y cumplió con lo establecido en la revisión del Formato por parte del Sistema de Estudios de Posgrado.

INFORMACIÓN DEL ESTUDIANTE:

Nombre Completo: Mario Cabrera Vega.

Número de Carné: B11209 Número de cédula: 1-1501-0981.

Correo Electrónico: mario.cabrera.cr@gmail.com.

Fecha: 02/02/22. Número de teléfono: 8915-7615.

Nombre del Director (a) de Tesis o Tutor (a): Edgar Casasola Murillo.

MARIO ALBERTO CABRERA VEGA (FIRMA)

Digitally signed by MARIO ALBERTO CABRERA VEGA (FIRMA)
Date: 2022.02.11 08:48:24 -06'00'

FIRMA ESTUDIANTE

Nota: El presente documento constituye una declaración jurada, cuyos alcances aseguran a la Universidad, que su contenido sea tomado como cierto. Su importancia radica en que permite abreviar procedimientos administrativos, y al mismo tiempo genera una responsabilidad legal para que quien declare contrario a la verdad de lo que manifiesta, puede como consecuencia, enfrentar un proceso penal por delito de perjurio, tipificado en el artículo 318 de nuestro Código Penal. Lo anterior implica que el estudiante se vea forzado a realizar su mayor esfuerzo para que no sólo incluya información veraz en la Licencia de Publicación, sino que también realice diligentemente la gestión de subir el documento correcto en la plataforma digital Kerwá.

1. Introducción

El ciberanzuelo (*clickbait* en inglés) es un término peyorativo que se le da a cierto tipo de contenido en Internet. Este tipo de contenido se caracteriza por intentar atraer la atención del usuario, con el fin de que este acceda al contenido a través de un clic. Sobre todo, el objetivo principal del *clickbait* es aumentar los ingresos publicitarios a través de las visitas de los usuarios. Para esto, los creadores de contenido recurren generalmente al uso de publicidad engañosa.

En particular, los titulares *clickbait* buscan captar la atención del usuario mediante diferentes técnicas, como por ejemplo el uso de palabras sensacionalistas, frases engañosas o exageraciones. Algunos ejemplos de titulares *clickbait* son los siguientes:

- "19 fotos que no podrás borrar de tu cabeza"[1]
- "21 personas que no tienen perdón de dios"[2]
- "Estas ideas de la física acerca del tiempo te volarán la cabeza"[3]

Por lo general el contenido ligado a este tipo de titulares no brinda información de calidad para el usuario, haciendo que este se sienta decepcionado con los resultados obtenidos después de haber hecho clic.

Con el auge de las redes sociales, noticias falsas, y contenidos virales, se ha despertado el interés de detectar de forma automática los contenidos *clickbait* en Internet. Por ello, se han desarrollado herramientas, en su mayoría basadas en aprendizaje máquina supervisado, con el fin de combatir el uso de *clickbait*.

[1] https://www.buzzfeed.com/mx/philippjahner/fotos-que-guardaras-en-tu-cabeza-para-siempre

[2] https://www.buzzfeed.com/mx/raquelmiserachi/infierno-carcel-muerte

[3] https://pijamasurf.com/2019/04/estas_ideas_de_la_fisica_acerca_del_tiempo_te_volaran_la_cabeza/

A raíz de esto, se formula la pregunta de investigación: ¿Cómo pueden ser utilizados los *word embeddings* para la detección automática de *clickbait* en idioma español? Además se establece como pregunta metodológica: ¿es posible utilizar los *word embeddings* para diferenciar de forma efectiva titulares *clickbait* de los titulares *no-clickbait*?

Este documento tiene la siguiente estructura. La siguiente sección describe la justificación de la investigación, después se presentan el objetivo general y los objetivos específicos. Seguidamente se presenta el marco conceptual y se describe el estado del arte. Luego se describe la metodología, los resultados obtenidos, y finalmente se presentan las conclusiones y trabajo futuro.

2. Justificación

La detección automática de titulares *clickbait* es importante en diversas áreas tanto de computación como ciencias sociales. Algunas consecuencias negativas del *clickbait* son las siguientes:

- Desde el punto de vista de la ciberseguridad, este tipo de titulares podrían conducir a *malware*, o representar ataques de tipo *phishing* y causar daños irreparables en el sistema del usuario.
- El uso de titulares *clickbait* contribuye al esparcimiento de desinformación y noticias falsas. En particular se puede destacar su uso engañoso y deshonesto con el fin de influir en actitudes sociales y la opinión pública.
- Por último, este tipo de contenido busca generar una respuesta emocional en el usuario, por lo que puede generar una mayor polarización social sobre temas como política, religión, entre otros.

De acuerdo a los resultados obtenidos a partir de una revisión sistemática de literatura ejecutada en marzo del 2021; fue posible conocer la existencia de

herramientas clasificadoras para la detección de *clickbait* en idioma inglés. Sin embargo, no existen herramientas similares para el idioma español. Inclusive, no existe un conjunto de datos en idioma español que pueda ser utilizado por un modelo clasificador.

La presente investigación pretende realizar una serie de aportes tecnológicos. Primero, la creación de un conjunto de datos etiquetado que corresponde a titulares *clickbait* en idioma español. Segundo, la implementación de un modelo de aprendizaje de máquina para llevar a cabo detección automática de *clickbait*. Tercero y último, realizar una evaluación del desempeño del uso de *word embeddings* en la detección de *clickbait*.

A continuación se detallan los objetivos.

3. Objetivos

En esta sección se describe el objetivo general y los objetivos específicos de la investigación.

3.1. General

Evaluar a través de un modelo de aprendizaje máquina supervisado, el uso de *word embeddings* para la detección automática de titulares *clickbait* en idioma español.

3.2. Específicos

- Construir un corpus etiquetado de titulares *clickbait* y *no-clickbait* en idioma español.
- Implementar un modelo basado en aprendizaje máquina supervisado, capaz de detectar automáticamente titulares *clickbait* a partir de los *word embeddings* que los conforman.

- Medir el desempeño del modelo obtenido, utilizando métricas de precisión, para determinar la utilidad de los *word embeddings* al detectar *clickbait*.

A continuación se presenta el marco conceptual.

4. Marco conceptual

El *clickbait* es una técnica en la cual se manipula la curiosidad de una persona, con el objetivo de hacerle abrir más páginas web a través de un clic. Esto usualmente se logra escribiendo titulares exagerados o irreales [11]. Una manera de poder identificar automáticamente este tipo de titulares es por medio de un modelo de aprendizaje máquina.

El aprendizaje máquina consiste en encontrar patrones ocultos a partir de grandes volúmenes de datos, y así poder resolver problemas de clasificación u optimización. Dentro de esta área, el desafío de identificar titulares *clickbait* es un problema de clasificación.

El objetivo de un modelo clasificador es poder predecir a qué clase pertenece una instancia de un conjunto de datos. Por ejemplo, en el contexto de esta investigación, predecir si un titular pertenece a la clase *clickbait* o a la clase *no-clickbait*. Para construir y evaluar el modelo, primero se debe entrenar este con un conjunto de datos previamente etiquetados con su debida clase. Este conjunto de datos es posible obtenerlo realizando *web scraping*.

Web scraping es el proceso por el cual se extrae información o contenido de una página web de forma automática a través de un programa denominado "bot" o "*crawler*". Los titulares obtenidos a través de *web scraping* son luego etiquetados por un grupo de personas, para así obtener el conjunto de datos con clases asignadas.

Entrenado el modelo, este se valida con un conjunto de datos no etiquetados. Es decir, se realizan predicciones sobre instancias que el modelo no

haya "visto" con anterioridad. A partir de estas predicciones, se obtienen las métricas de interés. Por ejemplo, la métrica de precisión se define como:

$$\frac{TP}{TP + FP}$$

Donde TP son las predicciones verdaderos positivos, y FP son las predicciones falsos positivos.

Los modelos clasificadores de aprendizaje máquina requieren representar los titulares de forma tal que estos sirvan como datos de entrada. Para ello es posible representar las palabras escritas en lenguaje natural, como vectores de números reales, conocidos como *word embeddings*. Se ha demostrado que representar las palabras de esta manera pueden mejorar significativamente y simplificar las aplicaciones de procesamiento de lenguaje natural [20].

En [20] se introducen dos diferentes modelos para generar *word embeddings*: el modelo bolsa continua de palabras (*Continuos Bag-of-Words*) y el modelo Skip-gram. Estos modelos consisten en utilizar un corpus de datos grande, para entrenar las representaciones vectoriales de las palabras de acuerdo a las palabras vecinas que aparecen en un mismo contexto. Una implementación de estos modelos está disponible como un proyecto de código abierto, cuyo nombre es word2vec [19].

Uno de los métodos de aprendizaje máquina más populares para clasificación de texto son las redes neuronales. En particular, las redes neuronales recurrentes han sido ampliamente adoptadas por su habilidad para modelar datos secuenciales tal como discurso y texto.

Las redes neuronales recurrentes son una clase de red neuronal artificial que utiliza información secuencial para mantener la historia de los datos a través del tiempo mediante sus capas intermedias [2]. Además, un tipo particular de red neuronal recurrente, llamado *Long Short Term Memory* (LSTM), es aún más eficiente que las redes neuronales recurrentes regulares, ya que preserva la información de mejor manera si una palabra está muy

alejada de otra en el texto. Algunos hiperparámetros a tomar en cuenta al desarrollar una red neuronal recuerrente tipo LSTM son los mostrados en el cuadro 1.

Cuadro 1: Hiperparámetros de una red neuronal tipo LSTM

Hiperparámetro	**Descripción**
Número de épocas	Define cuántas iteraciones de entrenamiento se realizan utilizando todo el conjunto de entrenamiento.
Tamaño de los baches	Define la cantidad de instancias del conjunto de entrenamiento que se utilizan antes de ajustar los pesos de la red.
Tasa de aprendizaje	Define qué tan rápido la red neuronal ajusta sus parámetros.
Número de capas ocultas	Define la cantidad de capas tipo LSTM presentes en la red neuronal.
Número de neuronas ocultas (*hidden size*)	Define la dimensión de la salida oculta (que se propaga a través del tiempo) de la capa LSTM de la red.

En la siguiente sección se presenta el estado del arte de la detección automática de *clickbait*.

5. Estado del arte

Se realizó una revisión sistemática de literatura en el mes de marzo del 2021, con el objetivo de conocer el estado del arte de la detección automática de *clickbait*. La mayoría de los artículos recuperados estudian la detección de *clickbait* en idioma inglés. Por otra parte, no se encontraron artículos

relacionados a la detección de *clickbait* en idioma español.

Los artículos obtenidos se clasificaron en tres categorías de importancia para esta investigación: 1) tipo de entrada de datos, 2) método o algoritmo de clasificación empleado, y 3) precisión obtenida de acuerdo al año de publicación.

5.1. Tipos de entrada de datos

Para llevar a cabo el procesamiento de clasificación de titulares *clickbait*, estos deben ser representados como datos de entrada para un modelo clasificador. El 71,00 % de los estudios recolectados utilizan un tipo de *embeddings* para esta tarea.

La clasificación de los estudios de acuerdo a los tipos de entrada de datos, se puede observar en la figura 1.

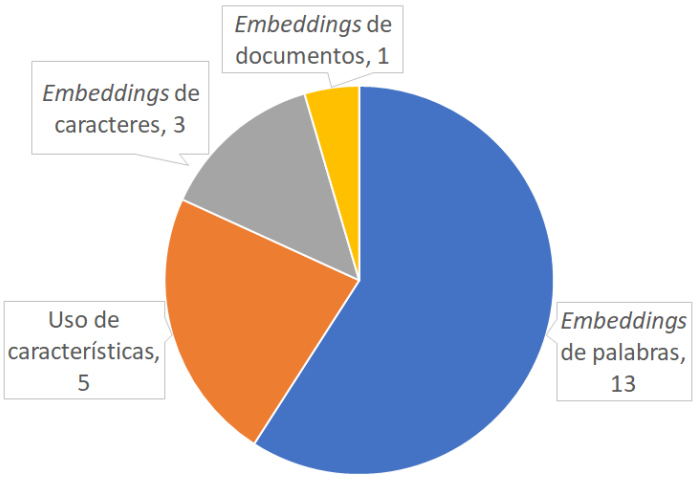


Figura 1: Clasificación de estudios de acuerdo a los tipos de entrada de datos

Los estudios [11], [15], [2], y [30] utilizan distintos tipos de *embeddings* al mismo tiempo para representar los datos de entrada. Por otra parte los

estudios que no utilizan *embeddings* ([14], [5], [31], [25], [23], [6]), utilizan extracción manual de *features* de los titulares. Algunos de estos *features* son: cantidad de palabras, cantidad total de caracteres, presencia de adjetivos, entre otros.

En [1] se presenta el único estudio en donde se utilizan *word embeddings* generados desde cero. Estos se generan como parte del modelo desarrollado, y se combinan con *word embeddings* pre-entrenados, a diferencia de estudios como [26], [16], [30], y [21], que utilizan únicamente *embeddings* pre-entrenados.

5.2. Métodos de clasificación

La figura 2 muestra la categorización de los estudios basada en los métodos utilizados para la creación de los modelos clasificadores. Los métodos más utilizados son las redes neuronales, que abarcan un 66,00 % de los estudios, seguido de las máquinas de soporte vectorial, con un 22,00 %.

Algunos estudios ([5], [31], [9], [24]) utilizan diversos métodos para comparar el rendimiento de la solución propuesta dentro del mismo artículo.

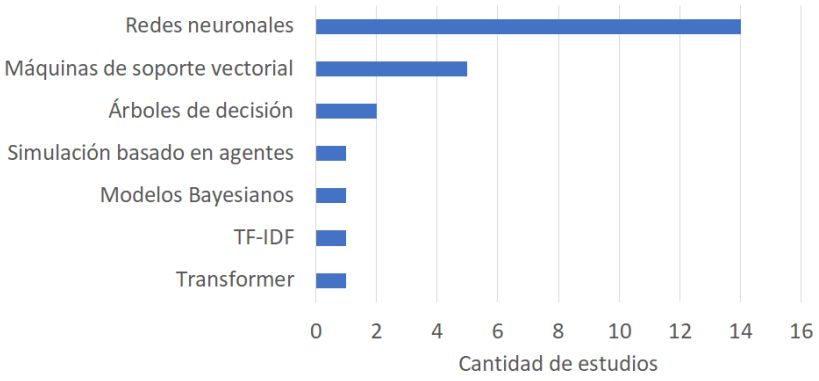


Figura 2: Caracterización de estudios de acuerdo a los métodos de clasificación

Por otra parte, se identificaron estudios que utilizan métodos menos convencionales. En [11] se utiliza un mecanismo de conteo de frecuencia de palabras entre el encabezado y el contenido, para determinar si el titular corresponde a *clickbait* o no. Relacionado a esto, en los estudios [7] y [30], también se estudia la similitud presente entre el encabezado y el contenido para realizar la clasificación. Por último, en [23] se utiliza un método basado en agentes, y en [30] se utiliza un modelo *Transformer*. Este último corresponde a un modelo que procesa datos de forma secuencial, en el cual se utilizan mecanismos de atención.

5.3. Métricas de precisión según año de publicación

La figura 3 muestra la distribución de los estudios de acuerdo a su año de publicación y la precisión obtenida por el método de clasificación respectivo. El estudio que reporta una mejor métrica es [18] con un 99,00 % de precisión. En segundo lugar se encuentra [17] con un 98,70 % de precisión, seguido de [3] con un 98,65 %. Por otro lado, la precisión más baja es reportada por el estudio [22], cuyo resultado es 73,20 %.

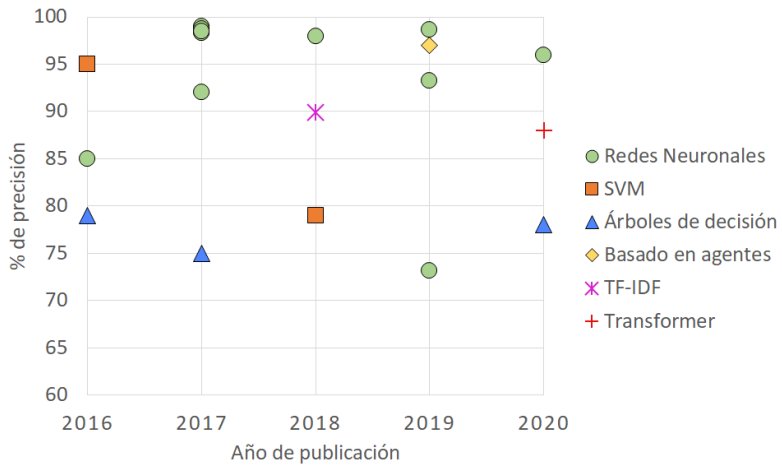


Figura 3: Distribución de estudios de acuerdo a precisión y año

Cabe mencionar además que algunos estudios más recientes, del año 2020, no presentan una precisión elevada debido a que buscan solucionar problemas más complejos. Por ejemplo, en [30] se busca detectar *clickbait* de acuerdo a si el encabezado se relaciona con el contenido de la publicación. Además en [6], la solución propuesta es independiente del lenguaje, es decir se busca que el modelo funcione a partir de titulares escritos en cualquier idioma.

Por otra parte, los estudios recolectados del año 2021, no fueron tomados en cuenta porque se enfocan en detectar *clickbait* en publicaciones multimedia como imágenes y videos. Por ejemplo los estudios [10], y [29], buscan detectar *clickbait* en los titulares presentes en publicaciones de videos de YouTube, realizando un análisis del titular y el contenido del video. Además el estudio presentado en [13] se enfoca en detectar *clickbait* en publicaciones con imágenes presentes en las redes sociales Instagram y Twitter.

Finalmente un 58,00 % de los estudios supera el 90,00 % de precisión obtenida, mientras que la media es de 90,08 %. Es importante mencionar que ni [24], ni [15] presentan resultados de métricas de precisión, por lo cuál no se incluyeron en la figura 3.

De acuerdo a la revisión de literatura, se corroboró que no existen estudios dedicados a la detección automática de *clickbait* en idioma español.

La siguiente sección describe la metodología de esta investigación.

6. Metodología

Esta sección presenta las actividades realizadas con el fin de cumplir los objetivos propuestos. El proceso metodológico se llevó a cabo de acuerdo a las ciencias del diseño, basado en el modelo presentado en [28]. El ciclo metodológico se muestra en la figura 4, y sus fases de describen a continuación:

- Reconocimiento del problema: identificar un problema y definir los objetivos de una solución. Como resultado de esta fase se obtiene una

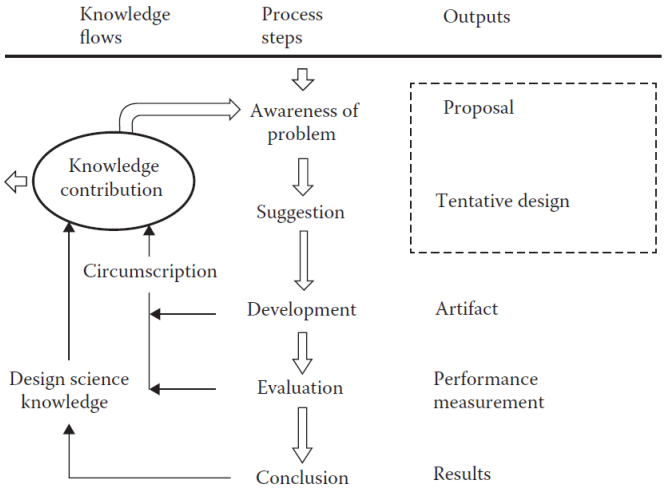


Figura 4: Ciclo de ciencias del diseño tomado de [28]

propuesta para un esfuerzo de investigación.

- Proposición: De acuerdo a la propuesta obtenida de la fase anterior, se crea un diseño tentativo con el cual se visualiza la solución del problema.

- Desarrollo: En esta fase el diseño es desarrollado e implementado, resultando en la construcción de un artefacto.

- Evaluación: El artefacto es evaluado de acuerdo a los criterios definidos en la propuesta de investigación. Los resultados obtenidos de esta evaluación y la información adicional obtenida a través de la construcción del artefacto, son utilizados para realizar otro ciclo de investigación con el conocimiento adquirido.

- Conclusión: Los resultados de la investigación son consolidados y documentados. Además el conocimiento adquirido es categorizado como "firme" –hechos que fueron aprendidos y pueden ser aplicados repetidamente– o "cabos sueltos" –comportamiento anómalo que carece de explicación y podría servir para investigación futura.

Figura 5: Metodología general de cada ciclo de investigación

De acuerdo al modelo metodológico de ciencias del diseño, se realizó el diseño tentativo, desarrollo, y evaluación de un prototipo. A partir de los conocimientos e información obtenidos del prototipo se realizó un nuevo ciclo metodológico acorde a lo aprendido.

El desarrollo y evaluación de cada ciclo metodológico se compone de una serie de actividades mostradas en la figura 5. Estas actividades consisten en 1) recolectar el conjunto de datos, 2) etiquetar el conjunto de datos, 3) obtener los *word embeddings*, 4) construir el modelo clasificador basado en aprendizaje máquina, y 5) entrenar y evaluar el modelo.

La recolección del conjunto de datos se realizó durante el curso PF3394 - Recuperación de Información. Por otra parte la investigación y evaluación del uso de *word embeddings* se realizó durante el curso PF3897 - Procesamiento de Lenguaje Natural. Mientras que la construcción y evaluación del modelo se llevó a cabo en el curso PF3115 - Técnicas Computacionales y Estadísticas de Aprendizaje de Máquinas.

6.1. Recolección del conjunto de datos

Con el fin de entrenar y evaluar el modelo clasificador se recolectaron titulares de publicaciones en distintos sitios web. Dicha recolección de datos se llevó a cabo de manera automática haciendo *web scraping* de fuentes en Internet seleccionadas de manera manual. Estas fuentes corresponden a las siguientes páginas web: Buzzfeed (`buzzfeed.com/mx`), BoredPanda (`boredpanda.es`),

PijamaSurf (pijamasurf.com), Vanidades (vanidades.com), y Wikinoticias (es.wikinews.org).

El principal criterio para la selección de las fuentes está basado en observación y popularidad. Es decir, de acuerdo a experiencia previa, estas fuentes ya habían sido identificadas como origen de contenido tanto *clickbait* como no *clickbait*. A su vez, estos sitios web se seleccionaron de modo que los temas de sus publicaciones sean similares. En este caso las publicaciones giran en torno a noticias, estilo de vida, moda, y entretenimiento.

En particular las fuentes Buzzfeed, y Wikinoticias son ampliamente utilizadas en la literatura para la creación de un corpus *clickbait*. Por ejemplo, estas fuentes son utilizadas en los estudios [2], [5], [9], [16], [17], [18], [23], [24], [25], [26].

Para realizar el proceso de *web scraping* se desarrolló un bot utilizando el *framework* de Scrapy[4], con el cual se obtuvieron los titulares de los sitios web mencionados anteriormente, de una forma sencilla y automática.

La figura 6 muestra el proceso de *web scrapping*. Este proceso consiste en tomar una serie de URLs predefinidas, y encolarlas en el módulo llamado "URL queue". Seguidamente para cada una de las URLs se solicita su contenido HTML de Internet, y se extraen los títulos e hipervínculos. Finalmente los títulos se escriben a un archivo de salida en formato CSV, y los hipervínculos son encolados en el módulo "URL queue" para repetir el proceso a partir de estas nuevas URLs.

6.2. Etiquetado del conjunto de datos

El proceso de etiquetar el conjunto de datos corresponde a determinar si un titular pertenece a la clase *clickbait* o a la clase *no-clickbait*. Este proceso se realizó de forma manual por un conjunto de personas que inspeccionan

[4] https://scrapy.org

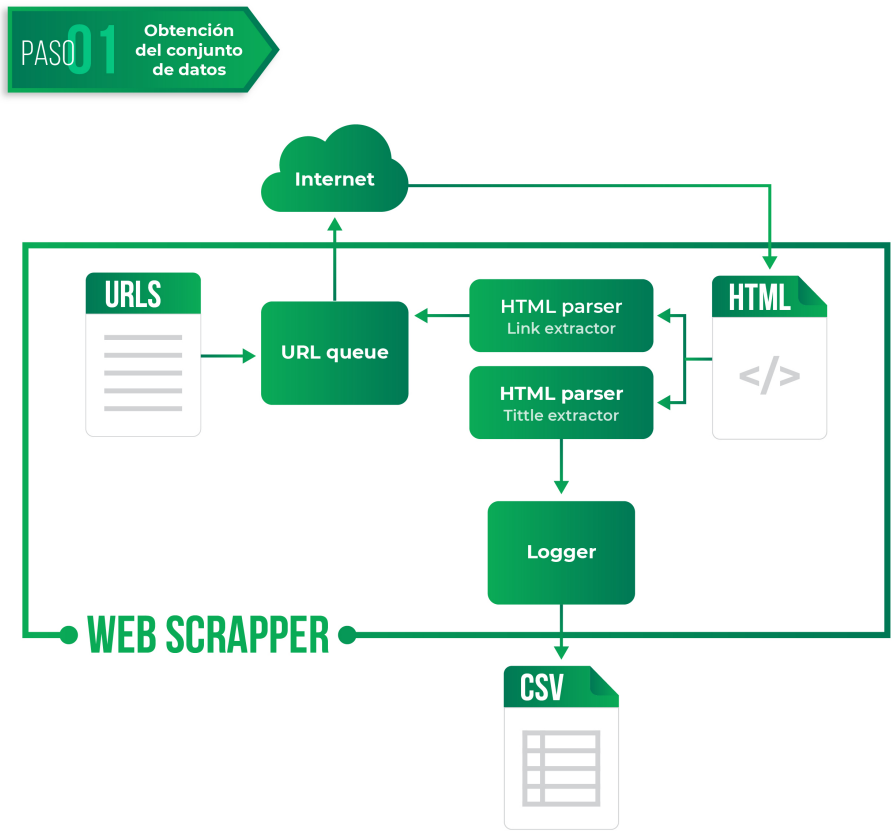


Figura 6: Recolección de titulares mediante *web scrapping*

cada titular y determinan, según su juicio, la clase de cada instancia del conjunto de datos.

Al realizar esta investigación se dispuso de catorce anotadores voluntarios que corresponden a estudiantes de posgrado de la Universidad de Costa Rica. Estos estudiantes cursan la carrera de Computación e Informática y la carrera de Lingüística.

Cada anotador tuvo la responsabilidad de etiquetar un subconjunto del conjunto de datos. Estos subconjuntos se distribuyen mediante el proceso mostrado en la figura 7, descrito a continuación:

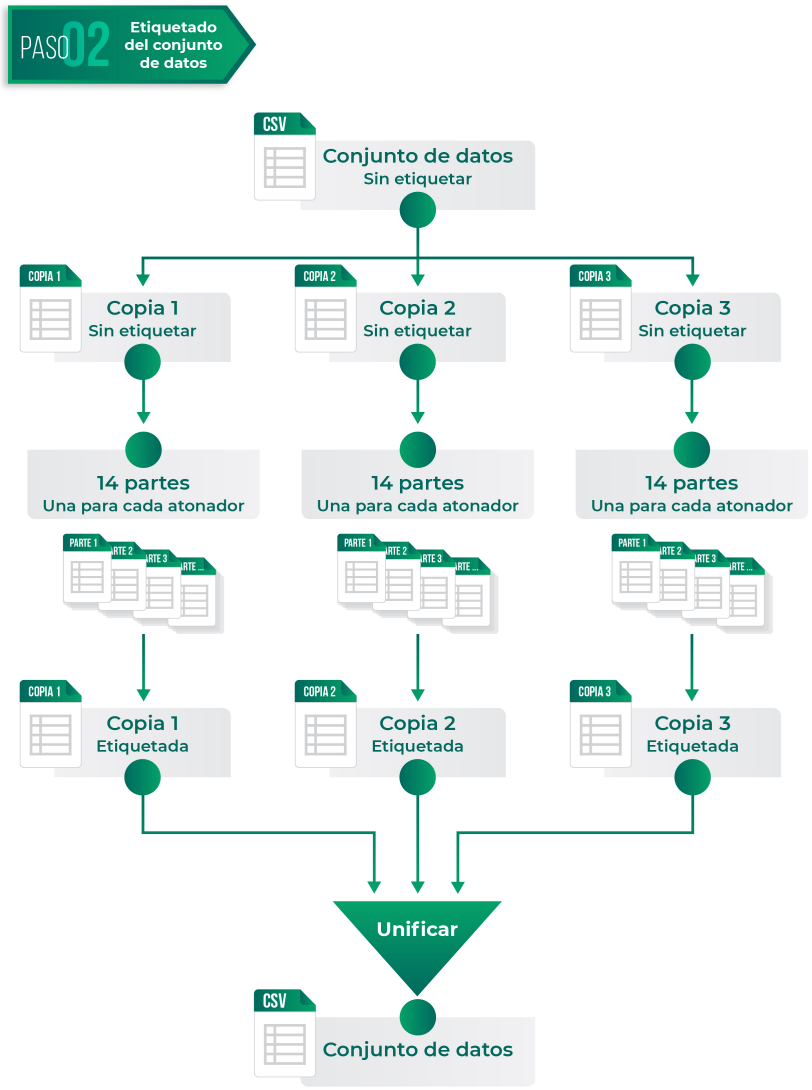


Figura 7: Proceso de etiquetado del conjunto de datos

- A partir del conjunto de datos original sin etiquetar, se crean tres copias iguales, llamados Copia 1, Copia 2, y Copia 3.
- Cada copia se divide de forma aleatoria en catorce partes iguales, una para cada anotador.

- A cada anotador se le asigna una parte de cada copia. Por ejemplo, al anotador 1 le corresponden la parte 1 de la Copia 1, la parte 1 de la Copia 2, y la parte 1 de la Copia 3. Se verifica que cada parte de cada anotador sea disjunta.

  En caso de que un mismo texto se encuentre duplicado, se intercambia una de sus apariciones con un texto de otro anotador para asegurar que las tres partes sean disjuntas. De esta forma ningún anotador evalúa un mismo titular más de una vez.

- Cada anotador etiqueta las partes que le hayan correspondido para así obtener la Copia 1 etiquetada, Copia 2 etiquetada, y Copia 3 etiquetada.

- Finalmente, las tres copias se unifican en una sola para obtener el conjunto de datos etiquetado.

El proceso anterior asegura que cada titular del conjunto de datos sea etiquetado por al menos tres anotadores diferentes. Para esto a cada anotador se le proporcionaron tres archivos en formato de hoja electrónica, correspondiente a cada parte de la Copia 1, Copia 2, y Copia 3. Así cada anotador emite un juicio para cada uno de los titulares.

En cada línea de la hoja de cálculo se presenta un titular que debe ser clasificado de acuerdo a las siguientes categorías:

- NO (Strong): se considera que el titular no es *clickbait*.
- NO (Weak): se considera, con reservas, que el titular no es *clickbait*.
- SI (Weak): se considera que el titular puede ser *clickbait* pero no con completa seguridad.
- SI (Strong): se considera que el titular sí es *clickbait*.

title	NO (STRONG)	NO (WEAK)	SI (WEAK)	SI (STRONG)	SCORE
Crean esta sencilla guía de moda masculina que ilustra lo que sí, lo que no, y lo que por favor no hay que ponerse				x	5
Crean una lámpara en forma del globo de "IT" y cuesta 37$			x		2.5
Cómo reutilizar y sacar partido a situaciones desafortunadas. 8 Ideas para decorar tu hogar reciclando			x		2.5
Después de que se burlaran de esta mujer en Tinder por su vestido, la marca ASOS puso su foto en su web			x		2.5
En Brasil ya han muerto 500 millones de abejas en 3 meses, y se cuestiona el futuro de nuestros alimentos	x				-5
El actor Tom Felton de Harry Potter se queja de envejecer a sus 32 años, y es troleado por su colega Matthew Lewis			x		2.5
El agua de los canales de Venecia se vuelve cristalina durante la cuarentena por el coronavirus		x			-2.5
WhatsApp compartirá los números telefónicos de los usuarios con Facebook	x				-5

Figura 8: Titulares etiquetados de acuerdo a categorías, tomado de Copia 1

De acuerdo a la categoría seleccionada por el anotador, se le asigna un puntaje a su elección. La categoría NO (Strong) equivale a un puntaje de -5, la categoría NO (Weak) equivale a un puntaje de -2.5, la categoría SI (Weak) equivale a 2.5, y la categoría SI (Strong) equivale a un puntaje de 5. En la figura 8 se muestra un extracto del conjunto de datos etiquetado de acuerdo a estas categorías.

El puntaje final de cada titular se obtiene al sumar los puntajes obtenidos en las tres copias etiquetadas del conjuntos de datos. De esta forma, si el puntaje es mayor a cero, el titular se clasifica con la clase *clickbait*. Si por otra parte el puntaje es menor a cero, el titular se clasifica con la clase *no-clickbait*. En caso de que el puntaje final sea igual a cero, el titular es evaluado por un nuevo anotador de acuerdo a las categorías antes mencionadas, y el puntaje que corresponde a su elección se suma al puntaje final.

La figura 9 muestra un extracto del conjunto de datos con su etiquetado final, donde la columna *label* indica si el titular es *clickbait*, representado con la etiqueta 1, o si el titular es *no-clickbait*, representado con la etiqueta 0.

title	Set 0 score	Set 1 score	Set 2 score	Tie breaker	Final score	label
Esta es Baddie Winkle, una abuela de 92 años con mucho estilo que lleva “desde 1928 robándote a tu hombre”	5	-2.5	5	0	7.5	1
La FIFA sancionó a Bolivia por alineación indebida de jugador	-2.5	-2.5	-5	0	-10	0
Un enfermero alemán confiesa haber asesinado a más de	2.5	2.5	5	0	10	1
18 Crímenes de cocina cometidos contra el pollo	5	5	5	0	15	1
Estas son las únicas 5 reglas que necesitas saber cuando	5	2.5	5	0	12.5	1
9 Cosas inesperadamente divertidas que puedes hacer con	5	5	5	0	15	1
15 Profundos y terribles pensamientos que tuvo el búho del	5	-5	5	0	5	1
Países del Caspio llegan a acuerdo sobre paz y desarrollo	-5	-2.5	-5	0	-12.5	0
Hallazgos romanos en el yacimiento de Bañugues	5	-5	-5	0	-5	0

Figura 9: Titulares con etiquetado final

6.3. Obtención de *word embeddings* y preprocesamiento del conjunto de datos

La presente investigación está basada en la hipótesis de que los titulares *clickbait* pueden ser identificados de acuerdo a la forma en que las palabras son utilizadas para generar curiosidad en el usuario.

Los *word embeddings* capturan significado sintáctico y semántico de las palabras en un idioma determinado, por ello se utilizó un corpus de *word embeddings* pre-entrenado en idioma español, obtenido de [4]. Este corpus corresponde a cerca de 1500 millones de palabras en español, entrenadas a partir de recursos de la web utilizando el algoritmo word2vec[5].

El preprocesamiento del corpus de *word embeddings* consiste en reemplazar los caracteres no alfa numéricos por espacios en blanco, reemplazar los dígitos por el token “DIGITO”, y reemplazar múltiples espacios en blanco por un único espacio en blanco. Consecuentemente, el mismo preprocesamiento se lleva a cabo con los titulares recolectados en este trabajo.

6.4. Construcción del modelo clasificador

En esta sección se presentan los métodos de clasificación utilizados en esta investigación. Se desarrollaron tres modelos, donde los dos primeros corres-

[5] https://code.google.com/archive/p/word2vec/

ponden a una línea base, utilizados para comparar los resultados obtenidos por el tercer modelo.

El primer modelo corresponde a una clasificación simple de acuerdo a la frecuencia de las palabras, mientras que el segundo modelo corresponde a un *Gaussian Naive Bayes*. Por último el tercer modelo corresponde a una Red Neuronal Recurrente tipo LSTM.

6.4.1. Línea base: Clasificación simple

Dado que no existe un estudio dedicado a la clasificación de titulares *clickbait* en idioma español, se realizó una clasificación sencilla del conjunto de datos, para así tener una línea base con la cual comparar el modelo desarrollado.

La idea general es clasificar un titular de acuerdo a las palabras más comunes del conjunto de datos perteneciente a la clase *clickbait*. Es decir, un titular se clasifica como *clickbait* si posee una palabra que aparece con frecuencia en el conjunto de datos etiquetado como *clickbait*. Los detalles de esta clasificación se muestran en la figura 10, y se implementan de acuerdo al siguiente algoritmo:

- **Pre-procesamiento**: El pre-procesamiento incluye eliminar caracteres no alfanuméricos, reemplazar los dígitos por el token "DIGITO", y remover *stopwords*. A diferencia de los demás modelos desarrollados, para este método se remueven los *stopwords* para evitar que las palabras más frecuentes de cada clase sean palabras que realmente no representan dicha clase.

- **Frecuencia de palabras:** Se obtienen las frecuencias de cada palabra dentro del vocabulario del conjunto de datos previamente etiquetados (figura 7). Después se seleccionan las dos palabras más frecuentes de

los titulares etiquetados como *clickbait*, y se almacenan como *palabra1* y *palabra2* (representadas en el gráfico como P1 y P2).

- **Clasificación:** Si un titular posee al menos una de las palabras obtenidas del punto anterior (*palabra1* o *palabra2*), el titular es clasificado como *clickbait*, de otra forma es clasificado como *no-clickbait*.

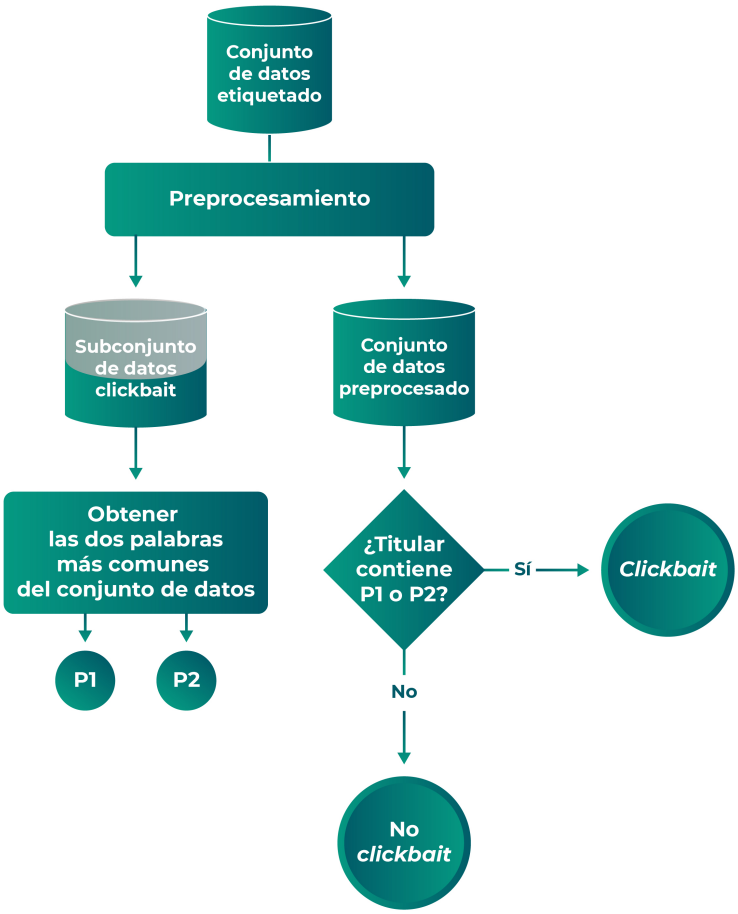


Figura 10: Algoritmo de clasificación simple (línea base)

6.4.2. Línea base: *Gaussian Naive Bayes*

Se realizó un modelo bayesiano ingenuo (*Naive Bayes*), para realizar la clasificación de acuerdo al teorema de Bayes. La idea general es clasificar un titular de acuerdo a la probabilidad de que este pertenezca a una clase, dado el conjunto de palabras que lo conforman.

Los detalles de esta clasificación se implementan de acuerdo al siguiente algoritmo:

- **Pre-procesamiento**: El pre-procesamiento incluye eliminar caracteres no alfanuméricos, reemplazar los dígitos por el token "DIGITO", y eliminar múltiples espacios en blanco por un único espacio en blanco.
- **Frecuencia de palabras (TF-IDF):** Se obtienen las frecuencias de cada palabra dentro del vocabulario del conjunto de datos. De forma que los pesos que representan cada palabra son utilizados como entrada de datos para el modelo.
- **Clasificación:** De acuerdo al teorema de Bayes, se calcula la probabilidad de que un titular pertenezca a una clase (denotada con la letra y), de acuerdo a las palabras que lo conforman (denotadas con la letra X). Es decir un titular se clasifica como *clickbait* si la probabilidad $P(clickbait|X)$ es mayor que la probabilidad $P(no - clickbait|X)$.

6.4.3. Red Neuronal Recurrente tipo LSTM

El modelo propuesto en esta investigación corresponde a una Red Neuronal Recurrente (RNN) de tipo *Long Short-Term Memory* (LSTM). La razón por la cual se propone una Red Neuronal Recurrente es porque estas brindan la posibilidad de representar las instancias de entrada como una secuencia de datos. Las redes neuronales tradicionales no brindan esta posibilidad, dado

que las entradas y salidas son representadas con un tamaño fijo no secuencial. Es por ello que las redes neuronales recurrentes han sido ampliamente adaptadas para modelar datos secuenciales, tal como discurso y texto [2].

Para esta investigación, representar las entradas como una secuencia de datos es importante dado que un titular es realmente una secuencia de palabras. Por ello, la forma en que una secuencia de palabras es utilizada nos ayuda a determinar si el titular es *clickbait* o no.

Por ejemplo, el titular "Los mejores 30 hombres del año en computación" se puede considerar *clickbait*. Sin embargo, el titular "Los hombres de 30 años son mejores en computación" no es *clickbait*. Con esto se demuestra que a pesar de utilizar prácticamente las mismas palabras, la diferencia principal radica en la secuencia de estas.

Otra de las razones por las cuales esta investigación utiliza una red neuronal recurrente, es porque estas han sido ampliamente utilizadas en la literatura para este tipo de tareas. Por ejemplo en [8], se utiliza una para auto generar texto *clickbait*.

Por ello, al utilizar una red neuronal recurrente, es posible preservar la información que brinda una secuencia de palabras, y así obtener una salida que corresponde al puntaje *clickbait* del titular. Sin embargo, las redes neuronales recurrentes no preservan de la mejor manera la relación de palabras que se encuentran muy alejadas entre sí [12]. Para resolver este problema se utiliza la variante LSTM, ya que su implementación de "compuertas" preserva de mejor manera la información de cada elemento en la secuencia.

La figura 11 muestra la arquitectura de la red neuronal desarrollada, así como el procesamiento que se le da a cada titular antes de ser insertado a la red. Este proceso de describe a continuación.

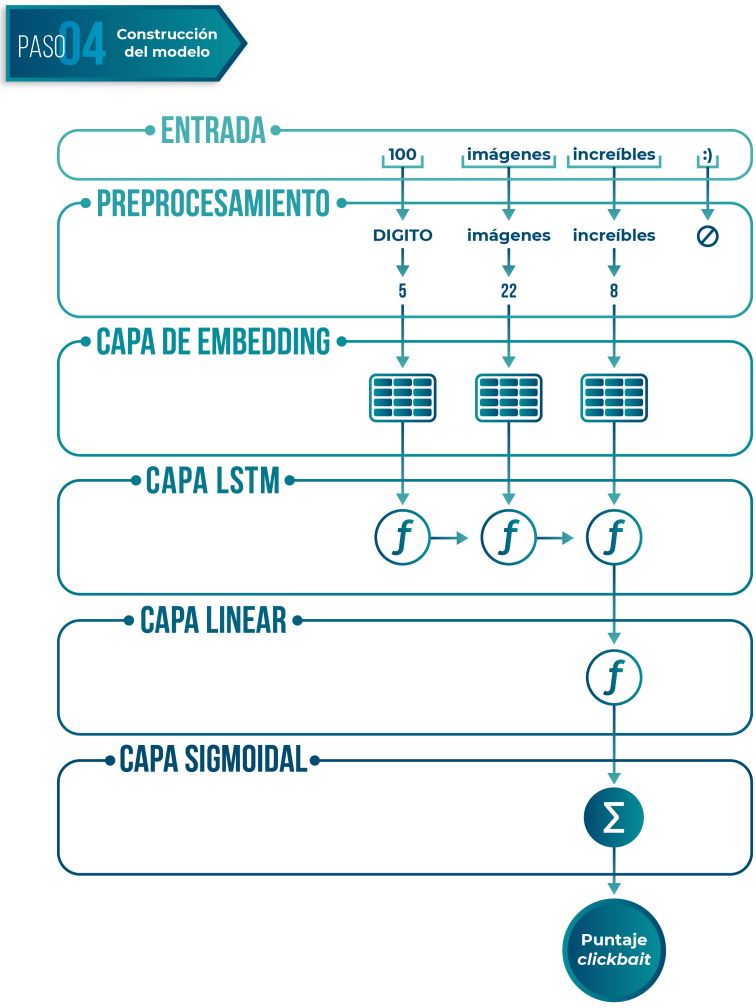


Figura 11: Arquitectura del modelo de red neuronal

- **Preprocesamiento:** Se realiza el mismo preprocesamiento que al corpus de *word embeddings* pre-entrenadas. Es decir, se eliminan caracteres no alfanuméricos, y se reemplazan los dígitos con el token "DIGITO". Como parte de esta fase también se asocia cada palabra del vocabulario con un número único que la identifique. De esta forma el titular es representado como una secuencia de números enteros.

- **Capa de *embedding*:** Se traduce cada palabra, identificada con un número único, a su respectiva representación vectorial (*word embedding*), de forma que la salida de esta capa sea una secuencia de *word embeddings*. Si una palabra no tiene un equivalente en el corpus de *word embeddings* pre-entrenados, entonces se utiliza un *word embedding* de únicamente ceros.

- **Capa LSTM:** Se procesa la secuencia de *word embeddings* como la entrada de una capa LSTM unidireccional. De la salida de esta capa se mantiene únicamente el resultado de la última celda para pasar a la siguiente capa. Esto porque es la última celda la que posee la información de toda la secuencia, dado que esta información fue transmitida a través del tiempo desde celdas anteriores por el *hidden state* de la LSTM.

- **Capa lineal:** Traduce la salida de la capa LSTM, cuyo tamaño está definido por el tamaño del *hidden state*, al tamaño de salida deseado de la red neuronal, en este caso uno.

- **Capa sigmoide:** Función de activación de la capa de salida. Esta salida representa una clasificación binaria; 0 para la clase *clickbait*, y 1 para la clase *no-clickbait*.

6.5. Entrenamiento y evaluación del modelo clasificador

El proceso de entrenamiento y evaluación del modelo clasificador se muestra en la figura 12. El conjunto de datos se divide aleatoriamente en tres grupos: el 80 % de instancias se utiliza como conjunto de entrenamiento, un 10 % se utiliza como conjunto de validación, y el restante 10 % se utiliza como conjunto de pruebas.

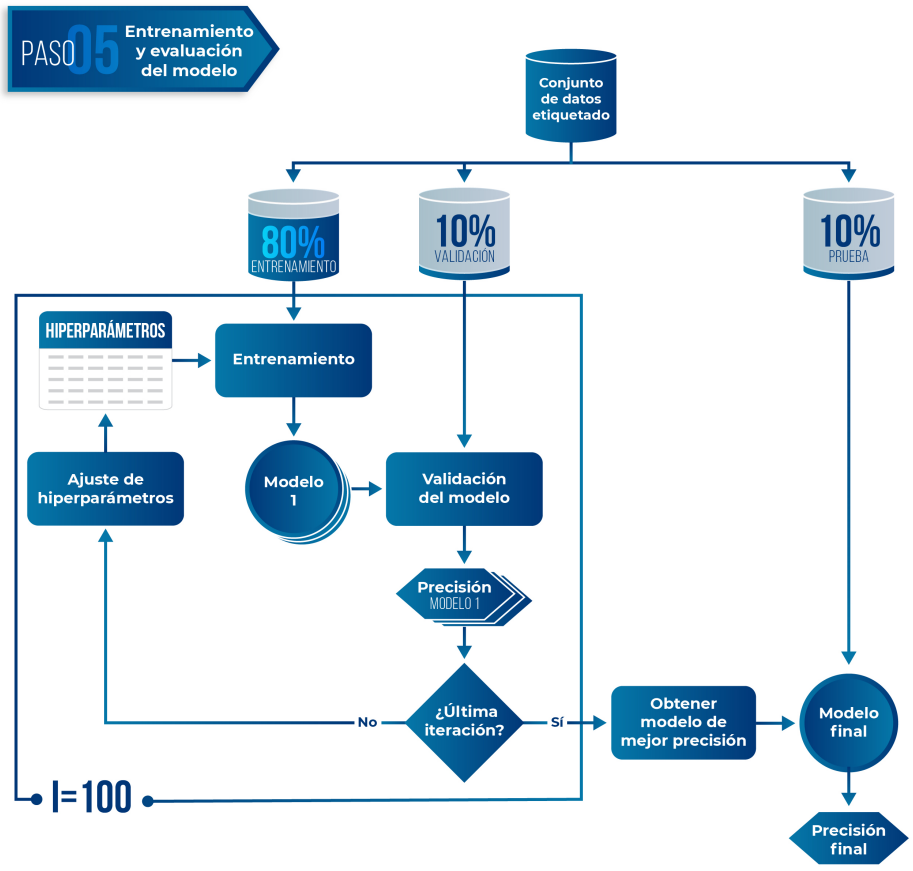


Figura 12: Proceso de entrenamiento y evaluación del modelo

Como se muestra en la figura 12, el proceso de entrenamiento consiste en entrenar el modelo a partir del conjunto de datos de entrenamiento y una serie de hiperparámetros, por ejemplo número de épocas, tasa de aprendizaje, y tamaño de baches.

Una vez que el proceso de entrenamiento termina, se obtiene el modelo entrenado, y este se utiliza para realizar el proceso de validación utilizando el conjunto de datos de validación. Este proceso consiste en realizar predicciones utilizando las instancias del conjunto de validación sobre el modelo entrenado, y así obtener la precisión del modelo durante la etapa de validación.

El proceso de validación se repite por cien iteraciones, realizando un ajuste de hiperparámetros en cada iteración. Al finalizar el ciclo de iteraciones, se obtiene, para cada iteración, la precisión del modelo dado una serie de hiperparámetros.

Seguidamente, los hiperparámetros que muestran una mejor precisión son seleccionados junto con el modelo entrenado, para proceder con la etapa de pruebas. Para esta etapa, se realizan predicciones utilizando el conjunto de datos de prueba, y se obtiene la precisión final del modelo.

7. Resultados

Esta sección brinda los resultados obtenidos de esta investigación mediante cuatro puntos: 1) características del conjunto de datos obtenidas mediante un análisis descriptivo, 2) descripción del comportamiento de los modelos con base a diferentes configuraciones de sus hiperparámetros, 3) métricas de desempeño obtenidas por los modelos desarrollados, y 4) análisis de la clasificación final para algunos de los titulares del conjunto de datos.

Tanto el conjunto de datos, como el modelo clasificador se encuentran disponibles en el siguiente enlace: `https://git.ucr.ac.cr/nlp/spanish_clickbait`.

7.1. Análisis del conjunto de datos

De acuerdo a los ciclos de ciencias del diseño presentado en la figura 4, se desarrolló un prototipo inicial para resolver el problema de investigación. Es por esto que el conjunto de datos pasó por dos diferentes fases: fase de prototipo y fase final.

Durante la fase de prototipo se obtuvieron un total de 5467 titulares, de los cuáles se conservaron 3192 instancias. Se conservó solamente un subcon-

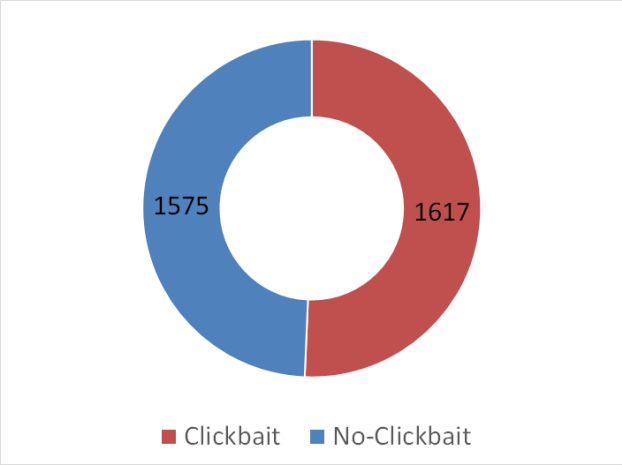


Figura 13: Distribución del conjunto de datos por clase

junto de las instancias para que así las dos clases, *clickbait* y *no-clickbait*, estuvieran balanceadas. En efecto, al realizar el prototipo se obtuvieron 1590 titulares *clickbait* y 1602 titulares *no-clickbait*.

Es importante mencionar que, durante la fase de prototipo, el etiquetado del conjunto de datos se llevó a cabo por una sola persona, mientras que durante la fase final, el etiquetado se llevó a cabo de acuerdo a lo descrito en la sección 6.2. Etiquetado del conjunto de datos.

Durante la fase final, se utilizó el conjunto de datos obtenido de la fase de prototipo. Por lo tanto, el conjunto de datos final se compone de 3192 titulares, de los cuales 1617 fueron etiquetados como *clickbait* y 1575 fueron etiquetados como *no-clickbait*, tal y como se muestra en la figura 13.

El conjunto de datos final fue analizado a través de una serie de estadísticas descriptivas, tomadas de [27], las cuales se pueden observar en el cuadro 2. La notación utilizada es la siguiente: T corresponde a la cantidad total de titulares, N a la cantidad total de palabras, V es la cantidad de palabras diferentes (vocabulario), p es la cantidad promedio de palabras por titular, y m la cantidad de veces que aparece una palabra en el corpus. Así, $V(m = 1, N)$

se refiere a la cantidad de palabras diferentes que aparecen solo una vez en el corpus.

Cuadro 2: Estadísticas descriptivas del conjunto de datos

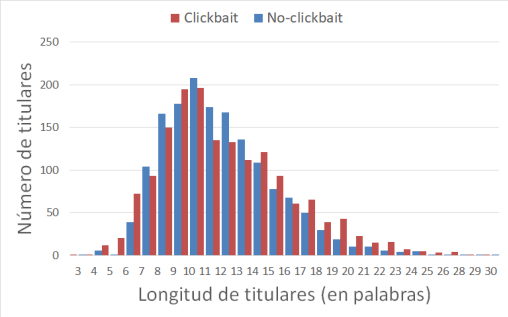
Variable	**Total**	**Clickbait**	**No-clickbait**
T	3192	1617	1575
N	38767	20638	18129
V	8762	5086	4927
p	12.14	12.76	11.51
V(m = 1, N)	5525	3393	3257

Algunas de estas características del conjunto de datos pueden ser visualizadas en la figura 14. En la figura 14a se puede observar la distribución de los titulares, tanto *clickbait* como *no-clickbait*, de acuerdo a su longitud en palabras. Este gráfico, junto con el cuadro 2, indica que los titulares *no-clickbait* son generalmente más cortos que los titulares *clickbait*.

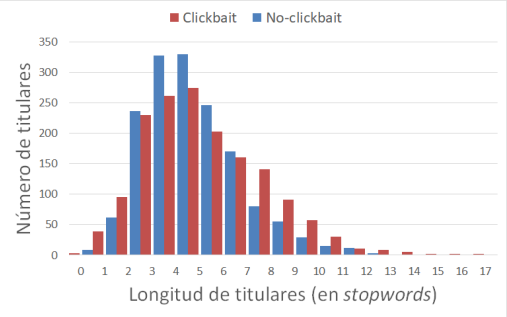
Mientras que la figura 14b muestra la distribución de los titulares de acuerdo a los *stopwords* utilizados en cada clase. Dada la diferencia notable en el empleo de los *stopwords*, se decidió no eliminarlos al momento de entrenar y evaluar los modelos de clasificación desarrollados.

Por otra parte las figuras 14c y 14d muestran las palabras más comunes utilizadas en los titulares de la clase *clickbait* y *no-clickbait*, respectivamente. Estas indican que los titulares *no-clickbait* hacen referencia más comúnmente a sustantivos propios, como por ejemplo nombres de países o personas. Mientras que los titulares *clickbait* utilizan palabras que apelan más a la curiosidad del lector, como por ejemplo “fotos” e “imágenes”.

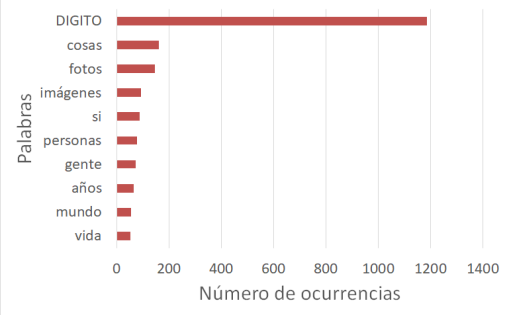
Otro de los análisis realizado al conjunto de datos, es la validación del etiquetado. Según el estudio efectuado en [5], existen algunos patrones o prácticas, que hacen a los titulares *clickbait* en cierto grado identificables sobre los



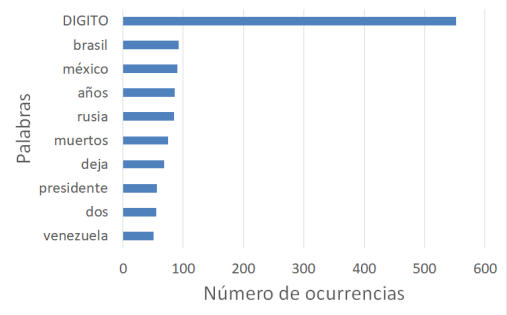
(a) Distribución de titulares de acuerdo a su número de palabras



(b) Distribución de titulares de acuerdo a su número de *stopwords*



(c) Palabras más comunes de la clase *clickbait*



(d) Palabras más comunes de la clase *no-clickbait*

Figura 14: Características del conjunto de datos

titulares *no-clickbait*. Estas características fueron tomadas en cuenta, como un tipo de lista de verificación, para revisar el conjunto de datos final, y se presentan a continuación.

- *Listicles*: Los titulares *clickbait* son frecuentemente presentados como *listicles*. Este el nombre que se le da a los artículos presentados en forma de lista (*list* + *article*), en los cuales por lo general se incluye un número como parte del titular. Por ejemplo “Las 10 mejores canciones de la historia”.

- *Stopwords*: Los *stopwords* son las palabras más utilizadas en un lenguaje, y según el estudio, los titulares *clickbait* emplean más *stopwords* que los titulares *no-clickbait*.

- Hipérboles: Las palabras que tienen una connotación muy negativa o muy positiva también son más frecuentes en titulares *clickbait* que en titulares *no-clickbait*.

- Signos de puntuación: Un mayor uso de signos de puntuación también es una de las características de los titulares *clickbait*. En particular su uso informal, como por ejemplo "...", "!!", o "!?".

- Jerga de Internet: Se refieren a palabras o abreviaciones utilizadas en Internet para captar la atención del usuario. Por ejemplo OMG, LOL, etc. Sin embargo, cabe destacar que en idioma español estos no son muy utilizados.

- Determinantes: El uso de determinantes, tales como "este", "aquel", y "estos", son más comunes en titulares *clickbait*. Esto se debe a que buscan apelar a la curiosidad que el lector siente por conocer a quién se refieren los determinantes empleados.

- Adverbios: Similar al uso de las hipérboles, los titulares *clickbait* utilizan con mayor frecuencia los adverbios para captar la atención del usuario.

Se llevó a cabo una verificación de la lista anterior con cinco titulares seleccionados al azar del conjunto de datos etiquetado. Como parte de esta verificación se excluyó la característica de "*Stopwords*", por necesitar una comparación con un titular *no-clickbait*, y la característica "Jerga de Internet" porque en idioma español no son muy utilizadas.

Como se puede observar en los resultados obtenidos en el cuadro 3, no todos los titulares cumplen con todas las características, pero todos cumplen con al menos una.

Cuadro 3: Verificación del etiquetado de titulares del conjunto de datos

Titular	*Listicle*	Hipérboles	Signos	Determinantes	Adverbios
26 Preguntas para los que aman a los gatos	✓	✗	✗	✗	✗
El test más difícil que vas a hacer sobre 2015	✗	✓	✗	✓	✓
23 Ocasiones en las que el mundo fue demasiado cruel	✓	✓	✗	✗	✓
Cuidado con acabar así el 2014	✗	✗	✗	✓	✗
El quiz de postres más difícil que harás en tu vida	✗	✓	✗	✓	✓

7.2. Entrenamiento e hiperparámetros

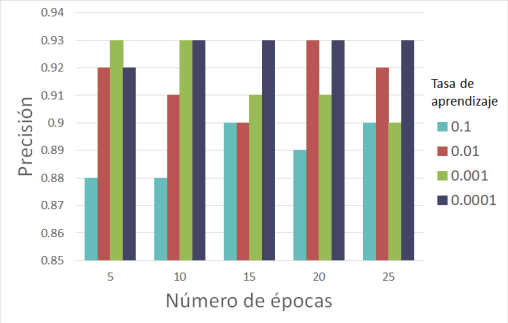
Esta sección presenta los resultados del entrenamiento y ajuste de hiperparámetros de la red neuronal recurrente tipo LSTM.

El entrenamiento se llevó a cabo de acuerdo al procedimiento mostrado en la figura 12. Después de realizar este procedimiento y partir de los resultados obtenidos, se seleccionó la mejor arquitectura de la red neuronal. Es decir, se realizó el proceso de selección de los hiperparámetros del modelo que mejor métrica de precisión reportan.

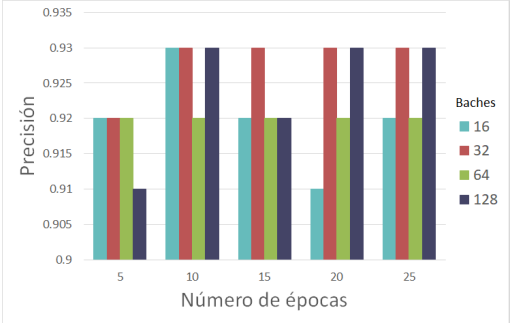
La figura 15 muestra el comportamiento del modelo bajo diferentes configuraciones de hiperparámetros en términos de la precisión obtenida. Es así como, la figura 15a muestra los resultados obtenidos variando la tasa de aprendizaje (*learning rate*) de la red neuronal. A partir de este experimento se concluyó que el mejor valor para este hiperparámetro fue de 0,0001.

Asimismo, la figura 15b muestra que el valor óptimo para el tamaño de los baches de la red neuronal es de 32. Por otra parte, como se observa en la figura 15c, el valor óptimo del *hidden size* es de 256, y el valor óptimo que corresponde al número de capas LSTM es de 1, como se puede ver en la figura 15d. Además, bajo esta configuración de hiperparámetros, el número

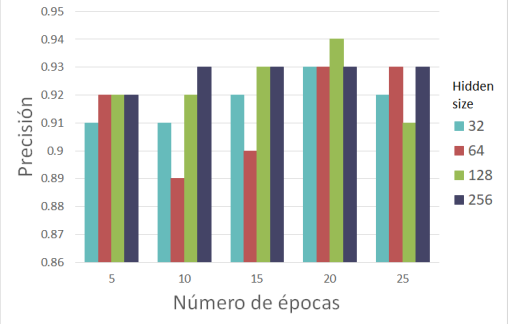
de épocas que mostró un mejor rendimiento es de 5 épocas.



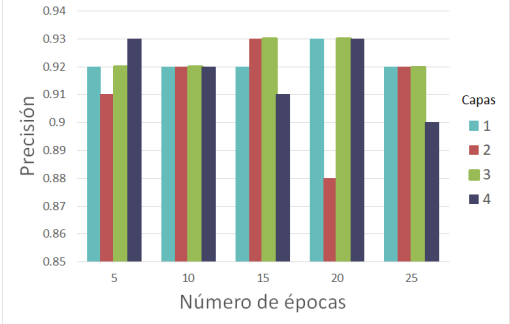
(a) Precisión obtenida con diferentes valores de tasa de aprendizaje



(b) Precisión obtenida con diferentes valores de tamaño de bache



(c) Precisión obtenida con diferentes valores de *hidden size*



(d) Precisión obtenida con diferentes valores de cantidad de capas LSTM

Figura 15: Resultados experimentales al usar diferentes valores de hiperparámetros para la RNN LSTM

El modelo obtuvo una mejor precisión utilizando los valores de hiperparámetros mostrados en el cuadro 4. Por otra parte, este modelo utiliza el optimizador Adam, y la función de perdida es *binary crossentropy loss*.

Cuadro 4: Valor de hiperparámetros de la RNN LSTM con mejor precisión

Hiperparámetro	**Valor**
Número de épocas	5
Tamaño de los baches	32
Tasa de aprendizaje	0.0001
Número de capas ocultas	1
Número de neuronas ocultas (*hidden size*)	256

7.3. Métricas de desempeño

En esta sección se presentan los resultados obtenidos de los experimentos realizados. Estos resultados corresponden a las métricas de desempeño de los modelos finales desarrollados.

7.3.1. Línea base: Clasificación simple

El modelo basal que se desarrolló consiste en la clasificación simple, tal como se explicó en la sección 6.4.1 de la metodología.

Al realizar la clasificación utilizando este algoritmo se obtuvo una precisión promedio de 69,00 %. Lo cual representa un rendimiento relativamente bueno, tomando en consideración la simplicidad del método de clasificación. El cuadro 5 muestra el reporte de clasificación generado por el módulo sklearn.metrics[6] para el método de clasificación simple.

7.3.2. Línea base: *Gaussain Naive Bayes*

Dado que en idioma español no existe un método de clasificación de titulares *clickbait*, y que el modelo de clasificación simple desarrollado en esta

[6] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

Cuadro 5: Reporte de clasificación: método de clasificación simple (*baseline*)

	Precisión	***Recall***	**Valor-F1**	**Soporte**
1	0.68	0.67	0.68	1575
0	0.68	0.69	0.69	1617
exactitud			0.68	3192
promedio	0.68	0.68	0.68	3192
promedio ponderado	0.68	0.68	0.68	3192

investigación es muy sencillo, se realizaron experimentos también con un modelo bayeseano ingenuo de tipo gaussiano. Este tipo de modelo ha sido ampliamente utilizado para resolver este clase de problemas, por ejemplo en [18] se utiliza para detección de *clickbait* en idioma inglés.

Con el modelo bayesiano realizado en esta investigación se obtuvo una precisión promedio de 82,00 %, como se muestra en el reporte de clasificación en el cuadro 6.

Cuadro 6: Reporte de clasificación: método *Gaussian Naive Bayes* (*baseline*)

	Precisión	***Recall***	**Valor-F1**	**Soporte**
1	0.82	0.82	0.82	162
0	0.82	0.83	0.82	158
exactitud			0.82	320
promedio	0.82	0.82	0.82	320
promedio ponderado	0.82	0.82	0.82	320

7.3.3. Red Neuronal Recurrente tipo LSTM

El modelo de red neuronal recurrente tipo LSTM obtuvo muy buenos resultados con una precisión promedio de 92,00 % utilizando el conjunto de datos de pruebas.

En resumen, este modelo se implementó utilizando las bibliotecas de PyTorch[7], con la arquitectura de red mostrada en la figura 11, utilizando el optimizador Adam, y la función de pérdida *Binary Cross Entropy Loss*. A su vez se utilizaron los hiperparámetros mostrados en el cuadro 4, y el proceso de entrenamiento y validación se realizó de acuerdo a la figura 12.

Como se puede observar en la figura 16, durante el proceso de entrenamiento y validación, el valor de la pérdida disminuye conforme pasan las épocas. Demostrando así, que el modelo efectivamente aprende sobre el tiempo, y no hay un sobreajuste. Esto debido a que la pérdida disminuye tanto para el conjunto de datos de entrenamiento, como para el conjunto de datos de validación.

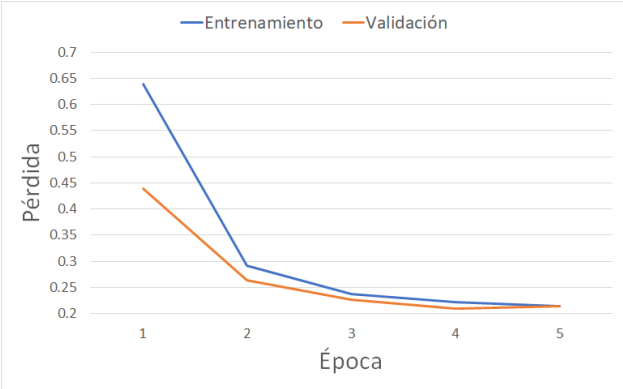


Figura 16: Valor de pérdida durante el entrenamiento y validación del modelo

[7] https://github.com/pytorch/pytorch

Finalmente, el reporte de clasificación para este modelo se muestra en el cuadro 7, mientras que la matriz de confusión se muestra en el cuadro 8.

Cuadro 7: Reporte de clasificación: método red neuronal recurrente LSTM

	Precisión	***Recall***	**Valor-F1**	**Soporte**
1	0.91	0.93	0.92	155
0	0.93	0.91	0.92	165
exactitud			0.92	320
promedio	0.92	0.92	0.92	320
promedio ponderado	0.92	0.92	0.92	320

Cuadro 8: Matriz de confusión para red neuronal recurrente LSTM

		Valor real		
		clickbait	*no-clickbait*	Total
Predicción	*clickbait*	143	15	158
	no-clickbait	12	150	162
	Total	155	165	

Al realizar la comparación final de los modelos implementados en esta investigación, se puede observar en el cuadro 9, que la red neuronal recurrente tipo LSTM es el modelo que reporta una mejor precisión. Seguidos por el modelo *Gaussian Naive Bayes*, utilizado como línea base, y finalmente el modelo de clasificación simple, utilizado también como línea base.

7.4. Análisis de titulares mal clasificados

Como puede observarse en la matriz de confusión (cuadro 8), el modelo de red neuronal recurrente clasificó erróneamente un total de 27 titulares.

Cuadro 9: Comparación de los modelos de acuerdo a métrica de precisión

Modelo	**Precisión**
Red Neuronal Recurrente tipo LSTM	0.92
Gaussain Naive Bayes	0.82
Clasificación simple	0.68

De los cuales 12 titulares *clickbait* fueron clasificados como *no-clickbait*, y 15 titulares *no-clickbait* fueron clasificados como *clickbait*.

En esta sección se analizan las características de algunos de los titulares mal clasificados con el objetivo de entender esta clasificación. El cuadro 10 muestra el análisis realizado a los titulares, tomando como resultado esperado la etiqueta manual que se le dio al titular en el conjunto de datos.

Cuadro 10: Análisis de titulares mal clasificados

(1) El ejército iraquí expulsa al Estado Islámico de Mosul	
Resultado obtenido: *no-clickbait*	**Resultado esperado:** *clickbait*
Análisis	
Este titular es informativo y no posee un enganche que sirva como método de atracción para el lector. Por otra parte, dado que se da en un contexto político y de relevancia internacional, es posible que pueda ser utilizado para atraer clics. Sin embargo, al no poseer ninguna de las características mencionadas en [5], es más probable que este titular sea considerado *no-clickbait* por la mayoría de lectores.	
Conclusión: Titular mal etiquetado por los anotadores.	
(2) Estrellan superauto único en el mundo quienes dicen ser cercanos a Peña Nieto	
Resultado obtenido: *clickbait*	**Resultado esperado:** *no-clickbait*

Análisis
Este titular es informativo, sin embargo utiliza palabras atrayentes al lector, como la frase "superauto único en el mundo". Este tipo de frases son utilizadas frecuentemente en titulares *clickbait*. Además el hecho de ligar el titular con una figura pública internacional, hace que el titular pueda ser considerado como *clickbait*.
Conclusión: El titular no fue realmente mal clasificado. Su categorización como *clickbait* o no, depende del contexto político y sociocultural del lector.

(3) Así es como se mantiene abierto uno de los últimos Videocentros de México	
Resultado obtenido: *no-clickbait*	**Resultado esperado:** *clickbait*
Análisis	
En [5] se demostró que los titulares *clickbait* utilizan más adverbios que los titulares *no-clickbait*. En este ejemplo, la utilización del adverbio demostrativo "así", induce la curiosidad del lector, por lo tanto puede ser considerado *clickbait*.	
Conclusión: Titular mal clasificado por el modelo.	

(4) IKEA responde a la aparición del bolso de Balenciaga de DIGITO que es igual que la bolsa de DIGITO centavos de IKEA	
Resultado obtenido: *clickbait*	**Resultado esperado:** *no-clickbait*
Análisis	

De acuerdo a las figuras 14c y 14d, la palabra más común, tanto para la clase *clickbait* como *no-clickbait*, es DIGITO. Sin embargo, para la clase *clickbait*, es mucho más recurrente que para la clase *no-clickbait*. Es probable que por esta razón el modelo clasificó este titular como *clickbait*.
Conclusión: Titular mal clasificado por el modelo.

8. Conclusiones y trabajo futuro

En esta investigación se presentó un modelo de red neuronal recurrente de tipo LSTM para clasificar titulares *clickbait* y *no-clickbait* en idioma español. Se desarrollaron además dos modelos, un clasificador simple y un *Gaussian Naive Bayes*, que fueron utilizados como líneas base de comparación.

El modelo de red neuronal recurrente tipo LSTM se implementó utilizando las bibliotecas de PyTorch, en conjunto con el optimizador Adam, y la función de pérdida *Binary Cross Entropy Loss*. De esta forma el modelo obtuvo una precisión promedio de 92,00 %, un *recall* de 92,00 %, y un valor de F1 de 92,00 %. Al comparar la métricas de precisión de los modelos de línea base con la red neuronal recurrente LSTM, esta los supera por un 10 %.

Es importante recalcar que los estudios obtenidos a través de la revisión de literatura reportan una precisión de hasta 99,00 %. Sin embargo, estos estudios aplican al idioma inglés y no al idioma español. Además se debe tomar en cuenta que el conjunto de datos es distinto, tanto en tamaño como en contenido, por lo cual no se espera que las métricas obtenidas en esta investigación sean comparables a las reportas en la revisión de literatura.

De acuerdo a los experimentos realizados, y las métricas obtenidas, es posible concluir que, para este caso de estudio en particular, los *word embeddings* de los titulares escritos en idioma español, junto con algoritmos de

aprendizaje máquina, son efectivamente útiles para detectar *clickbait*.

Este trabajo además ilustra que el uso de las redes neuronales recurrentes, y en particular las LSTM, realmente representa una mejora significativa para poder representar los datos de entrada como secuencias de información dependientes.

Sumado a lo anterior, esta investigación aporta avances en la detección automática de *clickbait* en idioma español. En efecto, los productos finales de la investigación incluyen el primer conjunto de datos *clickbait* de Costa Rica etiquetado en idioma español, el primer prototipo de detección automática de *clickbait* en idioma español, y la correspondiente investigación y análisis de resultados que pueden ser utilizados como insumo para futuras investigaciones. En particular, esta investigación puede ser utilizada como referencia para el desarrollo de identificación de desinformación, como la identificación de noticias falsas. Como trabajo futuro se propone utilizar *character embeddings*, en conjunto con los *word embeddings*. De esta forma sería posible tomar en cuenta el uso de los signos de puntuación como distintivo de los titulares *clickbait*, tal como se menciona en [5].

Una investigación futura podría también realizar una evaluación del desempeño de los modelos al utilizar diferentes tipos de *embedding*. Por ejemplo realizar comparaciones de *word embeddings*, *character embeddings*, y *document embeddings*, así como *embeddings* generadas a partir de los algoritmos word2vec, GloVe, etc.

Con respecto al conjunto de datos como tal, se propone aumentar el tamaño de la colección, para así obtener una cantidad suficientemente grande de titulares. Esto con el fin de poder generalizar de mejor manera cualquier tema que pueda ser usado para generar *clickbait*.

Además, contrario a utilizar *word embeddings* pre-entrenados, sería posible generar los *embeddings* automáticamente. Dado el tamaño de la colección utilizada en esta investigación, se decidió utilizar un conjunto externo pre-

entrenado de *embeddings*. Sin embargo, podría ser más representativo generar los *embeddings* de las palabras a partir del insumo de la red neuronal.

Relacionado también a la entrada de datos del modelo de red neuronal, se propone hacer uso de anotaciones lingüísticas del corpus, como por ejemplo de modalidad, morfológicas, y sintácticas. De forma que estas sean usadas como *features* en conjunto con los *word embeddings*.

En lo que respecta a la arquitectura de red neuronal utilizada, se propone como trabajo futuro realizar más experimentos con los hiperparámetros. Para que estos se seleccionen de manera automática, y así obtener una configuración de hiperparámetros y arquitectura de red neuronal que brinde mejores resultados.

Para finalizar, se concluye que los objetivos planteados en la investigación fueron alcanzados, y los productos resultantes pueden ser utilizados para investigaciones actuales referentes a procesamiento de lenguaje natural en idioma español. Particularmente aplicado a la identificación de textos tipo *clickbait*, desinformación, y noticias falsas.

Referencias

[1] A. Agrawal. «Clickbait detection using deep learning». En: *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*. Oct. de 2016, págs. 268-272. DOI: 10.1109/NGCT.2016.7877426.

[2] Ankesh Anand, Tanmoy Chakraborty y Noseong Park. «We Used Neural Networks to Detect Clickbaits: You Won't Believe What Happened Next!» En: *Advances in Information Retrieval*. Ed. por Joemon M Jose y col. Cham: Springer International Publishing, 2017, págs. 541-547. ISBN: 978-3-319-56608-5.

[3] U. Bhattacharjee. «Capsule Network on Social Media Text: An Application to Automatic Detection of Clickbaits». En: *2019 11th International Conference on Communication Systems Networks (COMSNETS)*. Ene. de 2019, págs. 473-476. DOI: 10.1109/COMSNETS.2019.8711379.

[4] Cristian Cardellino. *Spanish Billion Words Corpus and Embeddings*. Mar. de 2016. URL: https://crscardellino.github.io/SBWCE/.

[5] Abhijnan Chakraborty y col. «Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media». En: *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. ASONAM '16. Davis, California: IEEE Press, 2016, págs. 9-16. ISBN: 978-1-5090-2846-7. URL: http://dl.acm.org/citation.cfm?id=3192424.3192427.

[6] Claudia Ioana Coste, Darius Bufnea y Virginia Niculescu. «A New Language Independent Strategy for Clickbait Detection». En: *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. 2020, págs. 1-6. DOI: 10.23919/SoftCOM50211.2020.9238342.

[7] Manqing Dong y col. «Similarity-Aware Deep Attentive Model for Clickbait Detection». En: *Advances in Knowledge Discovery and Data Mining*. Ed. por Qiang Yang y col. Cham: Springer International Publishing, 2019, págs. 56-69. ISBN: 978-3-030-16145-3.

[8] Lars Eidnes. *Auto-Generating Clickbait With Recurrent Neural Networks*. 2016. URL: https://larseidnes.com/2015/10/13/auto-generating-clickbait-with-recurrent-neural-networks/.

[9] J. Fu y col. «A Convolutional Neural Network for Clickbait Detection». En: *2017 4th International Conference on Information Science and Control Engineering (ICISCE)*. Jul. de 2017, págs. 6-10. DOI: 10.1109/ICISCE.2017.11.

[10] Bhanuka Gamage y col. «Baitradar: A Multi-Model Clickbait Detection Algorithm Using Deep Learning». En: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, págs. 2665-2669. DOI: 10.1109/ICASSP39728.2021.9414424.

[11] A. Geçkil y col. «A Clickbait Detection Method on News Sites». En: *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. Ago. de 2018, págs. 932-937. DOI: 10.1109/ASONAM.2018.8508452.

[12] Sepp Hochreiter y Jürgen Schmidhuber. «Long Short-Term Memory». En: *Neural Computation* 9.8 (1997), págs. 1735-1780.

[13] Mini Jain y col. «Clickbait in Social Media: Detection and Analysis of the Bait». En: *2021 55th Annual Conference on Information Sciences and Systems (CISS)*. 2021, págs. 1-6. DOI: 10.1109/CISS50987.2021.9400293.

[14] Suhaib R. Khater y col. «Clickbait Detection». En: *Proceedings of the 7th International Conference on Software and Information Engineering*. ICSIE '18. Cairo, Egypt: ACM, 2018, págs. 111-115. ISBN: 978-1-4503-6469-0. DOI: 10.1145/3220267.3220287. URL: http://doi.acm.org/10.1145/3220267.3220287.

[15] Vaibhav Kumar y col. «Identifying Clickbait: A Multi-Strategy Approach Using Neural Networks». En: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. SIGIR '18. Ann Arbor, MI, USA: ACM, 2018, págs. 1225-1228. ISBN: 978-1-4503-5657-2. DOI: 10.1145/3209978.3210144. URL: http://doi.acm.org/10.1145/3209978.3210144.

[16] L. Liang, J. Zheng y J. Fu. «Sentence Classification with Transfer Network». En: *2018 International Conference on Information Systems and Computer Aided Education (ICISCAE)*. Jul. de 2018, págs. 379-382. DOI: 10.1109/ICISCAE.2018.8666915.

[17] Daniel López-Sánchez y col. «Hybridizing metric learning and case-based reasoning for adaptable clickbait detection». En: *Applied Intelligence* 48.9 (sep. de 2018), págs. 2967-2982. ISSN: 1573-7497. DOI: 10.1007/s10489-017-1109-7. URL: https://doi.org/10.1007/s10489-017-1109-7.

[18] S. Manjesh y col. «Clickbait Pattern Detection and Classification of News Headlines Using Natural Language Processing». En: *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*. Dic. de 2017, págs. 1-5. DOI: 10.1109/CSITSS.2017.8447715.

[19] Tomas Mikolov y col. «Distributed Representations of Words and Phrases and Their Compositionality». En: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Vo-*

lume 2. NIPS'13. Lake Tahoe, Nevada: Curran Associates Inc., 2013, págs. 3111-3119. URL: http://dl.acm.org/citation.cfm?id=2999792.2999959.

[20] Tomas Mikolov y col. *Efficient Estimation of Word Representations in Vector Space*. 2013. arXiv: 1301.3781 [cs.CL].

[21] Bilal Naeem y col. «A deep learning framework for clickbait detection on social area network using natural language cues». En: *Journal of Computational Social Science* 3.1 (abr. de 2020), págs. 231-243. ISSN: 2432-2725. DOI: 10.1007/s42001-020-00063-y. URL: https://doi.org/10.1007/s42001-020-00063-y.

[22] Amin Omidvar, Hui Jiang y Aijun An. «Using Neural Network for Identifying Clickbaits in Online News Media». En: *Information Management and Big Data*. Ed. por Juan Antonio Lossio-Ventura, Denisse Muñante y Hugo Alatrista-Salas. Cham: Springer International Publishing, 2019, págs. 220-232. ISBN: 978-3-030-11680-4.

[23] Deepanshu Pandey, Garimendra Verma y Sushama Nagpal. «Clickbait Detection Using Swarm Intelligence». En: *Advances in Signal Processing and Intelligent Recognition Systems*. Ed. por Sabu M. Thampi y col. Singapore: Springer Singapore, 2019, págs. 64-76. ISBN: 978-981-13-5758-9.

[24] S. Pandey y G. Kaur. «Curious to Click It?-Identifying Clickbait using Deep Learning and Evolutionary Algorithm». En: *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. Sep. de 2018, págs. 1481-1487. DOI: 10.1109/ICACCI.2018.8554873.

[25] Martin Potthast y col. «Clickbait Detection». En: *Advances in Information Retrieval*. Ed. por Nicola Ferro y col. Cham: Springer International Publishing, 2016, págs. 810-817. ISBN: 978-3-319-30671-1.

[26] Md Main Uddin Rony, Naeemul Hassan y Mohammad Yousuf. «Diving Deep into Clickbaits: Who Use Them to What Extents in Which Topics with What Effects?» En: *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*. ASONAM '17. Sydney, Australia: ACM, 2017, págs. 232-239. ISBN: 978-1-4503-4993-2. DOI: 10.1145/3110025.3110054. URL: http://doi.acm.org/10.1145/3110025.3110054.

[27] Marina Sokolova y Victoria Bobicev. «Corpus Statistics in Text Classification of Online Data». En: *CoRR* abs/1803.06390 (2018). arXiv: 1803.06390. URL: http://arxiv.org/abs/1803.06390.

[28] Vijay K. Vaishnavi y William Kuechler Jr. *Design Science Research Methods and Patterns: Innovating Information and Communication Technology*. 1st. USA: Auerbach Publications, 2007. ISBN: 1420059327.

[29] Agastya Vitadhani, Kalamullah Ramli y Prima Dewi Purnamasari. «Detection of Clickbait Thumbnails on YouTube Using Tesseract-OCR, Face Recognition, and Text Alteration». En: *2021 International Conference on Artificial Intelligence and Computer Science Technology (ICAICST)*. 2021, págs. 56-61. DOI: 10.1109/ICAICST53116.2021.9497811.

[30] Chuhan Wu y col. «Clickbait Detection with Style-Aware Title Modeling and Co-attention». En: *Chinese Computational Linguistics*. Ed. por Maosong Sun y col. Cham: Springer International Publishing, 2020, págs. 430-443. ISBN: 978-3-030-63031-7.

[31] Hai-Tao Zheng y col. «Boost Clickbait Detection Based on User Behavior Analysis». En: *Web and Big Data*. Ed. por Lei Chen y col. Cham: Springer International Publishing, 2017, págs. 73-80. ISBN: 978-3-319-63564-4.