

Detección de Noticias Falsas (Fake News) en Internet Utilizando Deep Learning

Mariana Esmeralda Centeno Reyes¹, Cristopher Jesús Chaire Rodríguez¹, Mario Isaac Fonseca Martínez¹, Diana Martínez Frías¹, Fernando Oviedo Paramo¹, Juan Carlos Gómez Carranza*

¹Departamento de Ingeniería Electrónica, División de Ingenierías Campus Irapuato-Salamanca, Universidad de Guanajuato {me.centenoreyes, cj.chairerodriguez, mi.fonsecamartinez, d.martinezfrias, f.oviedoparamo, jc.gomez}@ugto.mx

* Autor de correspondencia

Resumen

En este artículo se presenta un estudio de la detección automática de noticias falsas en Internet utilizando técnicas de deep learning. Esta detección es crucial para mantener la veracidad de la información que circula en medios digitales y evitar la desinformación. El estudio se dividió en dos fases; en la primera de ellas se utilizó un modelo de regresión logística que incluyó un proceso de limpieza de textos, una transformación de documentos en vectores de palabras utilizando el modelo FastText y finalmente una estandarización y normalización de los datos. En la segunda fase se emplearon modelos de clasificación con transformadores, incluyendo ALBERT, BERT, ELECTRA, RoBERTa y XLM-RoBERTa, los cuales utilizan arquitecturas de atención para entender el contexto y significado del texto, y posteriormente hacer una clasificación. Para la experimentación con los modelos se utilizó un conjunto de datos recolectado en Internet que incluye publicaciones de diversas redes sociales como Facebook, Instagram, Twitter, etc, que son noticias en inglés sobre el COVID-19. Este conjunto de datos está compuesto por 10,700 noticias, dividido en 5,600 noticias reales y 5,100 noticias falsas. La evaluación de los modelos de clasificación se realizó utilizando las métricas de accuracy, macro F1, recall, precisión y la matriz de confusión. De acuerdo con los experimentos, el modelo RoBERTa es el que muestra un mejor desempeño para la tarea en todas la métricas.

Palabras clave: Fake news, aprendizaje de máquina, deep learning, redes sociales, internet

1. Introducción

En México, cerca del 90% de las personas utilizan los medios electrónicos, como los portales de noticias y las redes sociales, para leer o ver noticias, debido a que estos medios son rápidos, económicos y fáciles de utilizar. Sin embargo, esta facilidad para distribuir la información ha llevado a la creación de noticias falsas, las cuales se han convertido en una preocupación mundial por sus repercusiones negativas en la sociedad, ya que éstas generan desinformación e ideas equivocadas entre la gente.

Durante el COVID-19 alrededor del 86% de los adultos en Estados Unidos utilizaron Internet para mantenerse informados. Medios como sitios web de noticias (por ejemplo, CNN, El Universal y Fox News) o redes sociales (por ejemplo, Facebook, Twitter y YouTube) son algunas de las principales plataformas que la gente utiliza para consumir noticias. Actualmente, el número de personas que utilizan estos medios como principal fuente de comunicación ha crecido exponencialmente. Si bien, estas plataformas puede ser un medio más fácil y rápido para divulgar información, también es un canal propenso a la desinformación y la propagación de noticias falsas.

La detección de noticias falsas es un proceso fundamental en la era digital, ya que proporciona una herramienta crítica para salvaguardar la veracidad de la información que circula en Internet. La propagación de noticias falsas puede tener consecuencias significativas, desde la desinformación del público hasta la influencia en decisiones políticas y de salud pública. Por ejemplo, en el ámbito de la salud, la detección de noticias falsas puede respaldar la correcta divulgación de información vital, asegurando que la población reciba datos precisos y confiables.

En este artículo se presenta un estudio sobre la detección automática de noticias falsas en Internet utilizando técnicas avanzadas de deep learning. El estudio se dividió en dos fases, considerando dos tipos de modelos de clasificación. La primera de ellas asociada con el uso de un modelo clásico de aprendizaje de máquina, la regresión logística; el cual es un modelo estadístico utilizado para predecir la probabilidad de un resultado binario basado en una o más variables independientes. La segunda fase, está asociada con modelos de deep learning llamados transformadores, tales como ALBERT, BERT, ELECTRA, RoBERTa y XLNet. Estos modelos utilizan la arquitectura de atención para comprender el contexto y el significado del texto, lo que los hace altamente efectivos para tareas de procesamiento de lenguaje natural, incluida la detección de noticias falsas.

Para experimentar con los modelos mencionados, se utilizó un conjunto de 10,700 publicaciones recolectadas de diversas redes sociales como Twitter, Facebook, Instagram, etc., que son noticias en inglés sobre el COVID-19. Las noticias están divididas en 5,600 noticias reales y 5,100 noticias falsas, separados en tres subconjuntos, uno de entrenamiento (6,420 noticias), otro de validación (2,140 noticias) y otro de prueba (2,140 noticias).

En la primera fase, con el modelo de regresión logística, se llevó a cabo un proceso de limpieza del conjunto de datos, en donde se eliminaron ligas (URLs), palabras vacías (stopwords), números, etc. Después, se transformó cada noticia a un vector de características utilizando los vectores de palabras del modelo FastText. Finalmente, se realizó una estandarización y normalización de los datos. En la segunda fase, con los transformadores, el procesamiento fue más sencillo ya que solo se excluyeron URLs del conjunto de datos.

Con los datos procesados se optimizaron los modelos con el subconjunto de entrenamiento y el subconjunto de validación, variando algunos parámetros como la penalidad o la regularización para la regresión logística; o el ajuste de los pesos en la red neuronal para los transformadores. Una vez obtenido un modelo optimizado, se entrenó un modelo final y se probó con el conjunto de prueba, midiendo su desempeño con las métricas de accuracy, las versiones macro de F1, recall y precisión, y la matriz de confusión. Estas métricas permitieron evaluar la efectividad de cada modelo en la clasificación de noticias falsas, proporcionando una comparación clara de su rendimiento. De acuerdo con los experimentos, el modelo RoBERTa es el que muestra un mejor desempeño en todas las métricas para la tarea de detección de noticias falsas.

El resto del artículo está organizado de la siguiente manera. En la Sección 2 se presentará una breve descripción de trabajos relacionados con la clasificación de noticias falsas en internet. En la Sección 3 se describirá detalladamente la metodología utilizada para abordar el estudio. La Sección 4 presentará los resultados obtenidos a partir de la experimentación con los modelos de aprendizaje. Finalmente, en la Sección 5 se presentarán las conclusiones del trabajo y se plantearán posibles direcciones para futuras investigaciones.

2. Trabajos Relacionados

La rápida difusión de información a través de Internet ha provocado un crecimiento exponencial de noticias falsas. Por lo tanto, la identificación automática de noticias falsas en medios en línea ha adquirido gran relevancia en los últimos años y se han desarrollado diversos enfoques para esta tarea.

Un enfoque común se basa en el análisis del contenido textual de las noticias. En [1], los investigadores examinaron la cantidad de errores gramaticales en noticias falsas, combinándolo con un clasificador Naïve Bayes para lograr un accuracy de 0.75. Otros estudios han explorado el uso de modelos de aprendizaje automático para analizar el contenido textual. En [2] y [3], las Máquinas de Vectores de Soporte (SVM) demostraron ser efectivas, alcanzando un accuracy de 0.92. Un enfoque alternativo, presentado en [4], emplea n-gramas para representar el contexto de las noticias y generar características para su clasificación combinado con una SVM de kernel lineal, obteniendo un accuracy de 0.92.

Los autores en [5] utilizaron vectores de palabras preentrenados (GloVe) como pesos en la capa de incrustación de modelos de aprendizaje profundo como Redes Neuronales Recurrentes (RNN) y Redes Neuronales Convolucionales (CNN). Una red de tipo *Long Short-Term Memory* (LSTM) Bidireccional (Bi-LSTM) sobresalió como el mejor enfoque, logrando un accuracy de 0.98. Un estudio similar [6] también

empleó vectores GloVe, pero propuso un modelo CNN profundo con múltiples capas ocultas para capturar características discriminatorias en las noticias falsas. La combinación de este modelo CNN con GloVe alcanzó un accuracy de 0.98.

Algunos estudios han utilizado el contenido textual y visual para identificar noticias falsas. En [7], los autores combinaron texto e imágenes de las noticias, utilizando una CNN para extraer características del texto y un modelo preentrenado VGG19 [8] para obtener características visuales de las imágenes. Este enfoque logró un accuracy de 0.82, una precisión de 0.84 y un valor F1 macro de 0.82.

El foro MEX-A3T, que forma parte del IberLEF (*Iberian Languages Evaluation Forum*), ha contribuido de manera significativa para el estudio del problema de las noticias falsas [9]. En este foro, los participantes exploraron diversos enfoques para resolver esta tarea, por ejemplo, el uso de modelos preentrenados como *Bidirectional Encoder Representations from Transformers* (BERT) junto con vectores de palabras y n-gramas de caracteres [10], con lo cual se obtuvo un F1 macro de 0.85 y un accuracy de 0.85 [11]. Otros estudios [12] utilizaron una combinación de representación binaria de texto con un modelo de bolsa de palabras y una SVM, alcanzando un accuracy de 0.81 y un F1 macro de 0.80. Los autores en [13] emplearon una representación de texto con *term frequency-inverse document frequency* (tf-idf) y la combinaron con diferentes modelos de aprendizaje de máquina, donde destacó el modelo SVM con los mejores resultados (F1 macro de 0.81 y precisión de 0.81).

A pesar de la variedad de enfoques en el campo de la identificación de noticias falsas, es necesario un mayor desarrollo e investigación para crear sistemas más robustos que filtren este tipo de noticias. Lo anterior debido a que este problema que aún está lejos de ser resuelto, debido a que es algo dinámico que modifica su forma de comunicación y que está creciendo exponencialmente, afectando a la sociedad y a las organizaciones de noticias.

3. Metodología

La Figura 3.1 muestra el esquema general de la metodología empleada en este artículo, la cual se divide en dos bloques: entrenamiento y prueba. A su vez, en cada bloque se tienen las dos fases comentadas anteriormente, una enfocada en el uso del modelo de regresión logística (LR) y otra enfocada al uso de los modelos de transformadores. Los pasos de procesamiento y transformación en el entrenamiento y la prueba son similares, ya que se requiere ajustar los datos para que puedan servir de entrada para la construcción y prueba de los modelos. En las siguientes subsecciones se describen a detalle los diferentes pasos mostrados en el esquema.

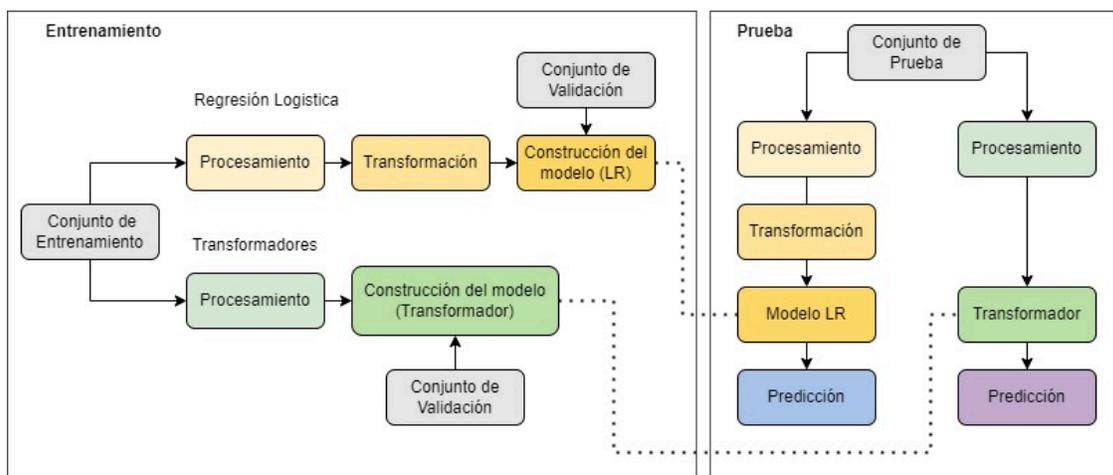


Figura 3.1. Esquema general de la metodología

3.1 Descripción del conjunto de datos

Para la experimentación se utilizó un conjunto de datos recopilado en Internet compuesto de 10,700 publicaciones de diferentes redes sociales, tales como Twitter, Facebook, Instagram, etc. que son noticias en inglés sobre el COVID-19. Este conjunto se dividió en tres subconjuntos: entrenamiento, validación y prueba. La Tabla 3.1 muestra la distribución de los datos en cada uno de los conjuntos, en cada una de las clases. La clase *fake* hace referencia a las noticias falsas. Los subconjuntos de validación y prueba tienen el mismo número de noticias, mientras que el subconjunto de entrenamiento es el mayor, para permitir la construcción de un modelo de clasificación robusto.

Subconjunto de datos	Número de noticias		Total
	Real	Fake	
Entrenamiento	3,360	3,060	6,420
Prueba	1,120	1,020	2,140
Validación	1,120	1,020	2,140
Total	5,600	5,100	10,700

Tabla 3.1. Distribución del conjunto de noticias

Para ilustrar el contenido del conjunto de datos, en la Tabla 3.2. se muestran algunos ejemplos de noticias en cada categoría.

Noticias reales	Noticias falsas
Our daily update is published. States reported 734k tests 39k new cases and 532 deaths. Current hospitalizations fell below 30k for the first time since June 22.	President Trump Asked What He Would Do If He Were To Catch The Coronavirus https://t.co/3MEWhusRZI #donaldtrump #coronavirus
States reported 1121 deaths a small rise from last Tuesday. Southern states reported 640 of those deaths. https://t.co/YASGRIT4ux	Alfalfa is the only cure for COVID-19
You can still fly the friendly skies without fear of COVID if airlines stay serious about safety. https://t.co/H8mGjwDl7G	A new alcohol-free sanitizer has been developed by the Dedan Kimathi University in Kenya.
Pregnant women with COVID have a 25% higher rate of premature births. https://t.co/g1hScYwGpr	Claim that the Jordanian Health minister says COVID-19 is a conspiracy in a video.

Tabla 3.2. Ejemplos de noticias reales y falsas

3.2 Procesamiento del conjunto de datos

En la fase correspondiente al modelo de LR, a los tres subconjuntos de datos (entrenamiento, prueba y validación), se les hizo un proceso de limpieza que consistió en transformar el texto de cada noticia a minúsculas; y en la eliminación signos de puntuación, ligas (URLs), palabras vacías (*stopwords*), números, palabras muy cortas (longitud < 3) y palabras muy largas (longitud > 25). Lo anterior debido a que tales elementos no proporcionan información relevante para que el modelo LR pueda aprender a discriminar con respecto a las dos categorías. En la Figura 3.2 se representa un ejemplo del proceso de limpieza de datos. En el lado izquierdo se tienen dos noticias (una real y una falsa) tal cual aparecen en el conjunto de datos. En el lado derecho tenemos ambas noticias transformadas a minúsculas y limpiadas de los elementos mencionados.

3.3 Proceso de Transformación

En la fase del modelo de LR, se transformó cada noticia de los subconjuntos de entrenamiento, validación y prueba a un vector de características. Este vector es el resultado del cálculo del promedio de los *vectores de palabras* de las palabras que conforman una noticia. Un *vector de palabra* es un vector que representa numéricamente el significado semántico de una palabra con base en las palabras que la rodean. Para obtener estos vectores se usó la colección de vectores de palabras obtenidos con el modelo FastText que fueron preentrenados con una colección de datos de Internet. Una vez obtenido el vector de características para cada noticia, se realizó una estandarización, removiendo la media y dividiendo por la desviación estándar en cada característica de los vectores; y posteriormente una normalización, dividiendo cada vector entre su norma euclidiana.

3.4 Construcción de los modelos

La Figura 3.4 muestra una representación visual de la forma de trabajar de los modelos de clasificación en cada fase:

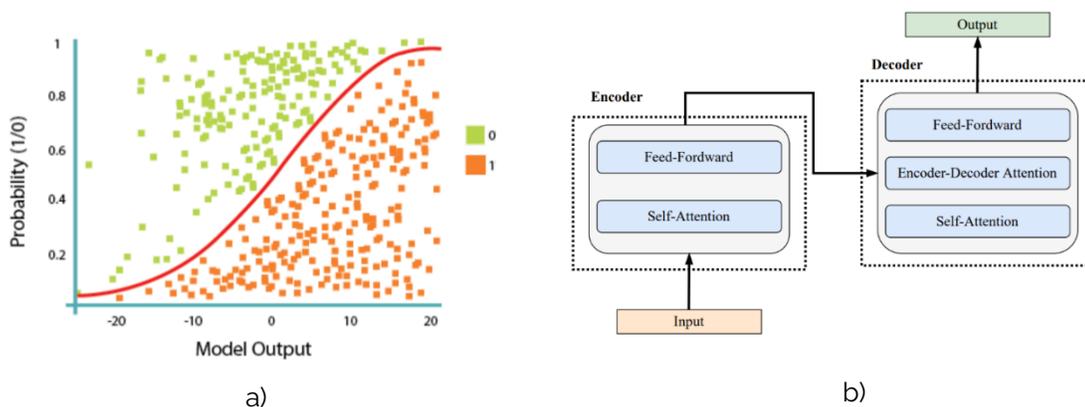


Figura 3.4. Representación de los modelos de clasificación a) Regresión Logística, b) Transformador

Regresión Logística

Para construir el modelo de LR se utilizó el módulo *Scikit-Learn* en Python. Este módulo permite el ajuste de diferentes hiperparámetros que influyen en la creación del modelo de LR. Como no es posible conocer de antemano los valores óptimos de los hiperparámetros, ya que cada conjunto de datos es diferente, se requiere hacer una búsqueda de los mejores valores para el conjunto de datos con el que se trabaja. Se realizó una experimentación variando los hiperparámetros de *penalty* y *C*; mientras que los hiperparámetros de *random_state*, *solver* y *max_iter* se dejaron constantes, con valores de 0, *liblinear* y 1000, respectivamente. La *penalty* se refiere a la regularización, en este caso se evaluaron los tipos L1 y L2. *C* controla la intensidad de la regularización, se probaron los valores 0.01, 0.1, 1, 10 y 100. Un valor pequeño de *C* conduce a una regularización más estricta y a un modelo más simple, mientras que un valor grande permite una regularización más laxa y un modelo más complejo. Ambos hiperparámetros son cruciales para evitar el sobreajuste, el cual es un problema potencial en los modelos de aprendizaje supervisado, se produce cuando un modelo es demasiado complejo y se ajusta demasiado a los datos de entrenamiento, lo que provoca predicciones precisas para los datos de entrenamiento, pero no para los datos nuevos.

Para elegir el mejor modelo, se utilizó el subconjunto de entrenamiento para construir modelos de LR para todas las combinaciones de los hiperparámetros, más el uso de estandarización y normalización. Cada modelo fue probado con el subconjunto de validación, midiendo su desempeño con la métrica macro F1. Se probaron en total 40 modelos y se seleccionó el que tuviera mayor macro F1.

Transformador

Al ser modelos que no requieren una transformación previa, la construcción de estos modelos fue más sencilla, ya que el modelo hace de manera automática gran parte del trabajo. Para generar el modelo se hizo usaron los módulos *ktrain*, *tensorflow* y *keras* en Python. Los modelos de transformadores utilizados fueron: ALBERT (“albert-base-v2”), BERT (“bert-base-uncased”), ELECTRA (“electra-base”), RoBERTa (“roberta-base”) y XLM-RoBERTa (“xlm-roberta-base”).

3.5 Predicción

Durante el proceso de entrenamiento, los modelos de clasificación aprenden a mapear las características encontradas en los vectores numéricos que representan las noticias a las categorías correspondientes. De esta forma, pueden posteriormente realizar predicciones sobre la categoría de una noticia desconocida. Para esto, una vez construidos los modelos, se utilizó el conjunto de prueba para probar qué también hacen la tarea de clasificación.

Considerando las noticias del conjunto de prueba, cada modelo puede clasificar a cada una con dos opciones: “real” (categoría positiva) o “fake” (categoría negativa). Con esto, se puede construir una matriz de confusión. Esta matriz relaciona las categorías verdaderas de las noticias con las categorías predichas por el modelo. En la Figura 3.5 se observa la representación visual de la matriz de confusión, en donde, TP corresponde al número de verdadero positivos (noticias reales clasificadas como reales), TN al número de verdaderos negativos (noticias fake clasificadas como fake), FP al número de falsos positivos (noticias fake clasificadas como reales) y FN al número de falsos negativos (noticias reales clasificadas como fake).

		Categoría predicha	
		Positivo	Negativo
Categoría Verdadera	Positivo	TP	FN
	Negativo	FP	TN

Figura 3.5. Representación de la matriz de confusión.

Utilizando la matriz de confusión se pueden calcular varias métricas que indiquen el desempeño de un modelo en la clasificación. En este trabajo se utilizaron las métricas de *accuracy*, *precisión*, *recall* y la métrica F1, la cual está basada en las métricas de precisión y recall. *Accuracy* mide la proporción total de noticias clasificadas correctamente entre todas las noticias. Toma valores entre 0 y 1, donde el 1 indica que todas las noticias han sido clasificadas correctamente y el 0 indica que todas han sido clasificadas incorrectamente. La *precisión* mide la proporción de noticias clasificadas correctamente, es decir, se centra en lo que el modelo dice y luego lo compara con la realidad; mientras que *recall* mide cuántas noticias positivas son correctamente clasificados. La métrica F1 es la media armónica de la precisión y el recall, también toma valores entre 0 (clasificación totalmente errónea) y 1 (clasificación perfecta). Estas métricas están expresadas por las siguientes ecuaciones:

$$(3.1) \quad Precision = \frac{TP}{TP + FP}$$

$$(3.2) \quad Recall = \frac{TP}{TP + FN}$$

$$(3.3) \quad F1 = 2 \frac{Precision * Recall}{Precision + Recall}$$

$$(3.4) \quad Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

En la versión macro de las métricas precisión, recall y F1, éstas se calculan para cada categoría de manera independiente y al final se calcula el promedio de las métricas independientes. Esto ayuda a mitigar el efecto en el caso de que haya alguna categoría con más cantidad de noticias.

Todos los códigos para el procesamiento de las noticias, el entrenamiento y la prueba de los modelos de clasificación se realizaron en Python utilizando las bibliotecas sklearn, pandas, numpy, ktrain, Keras y TensorFlow.

4. Resultados

En la Tabla 4.1 se muestran los resultados de los distintos modelos de clasificación en los subconjuntos de validación y prueba. Las columnas indican las diferentes métricas de evaluación utilizadas. En negritas aparece el valor más alto para cada columna/métrica.

Modelo	Subconjunto de Validación				Subconjunto de Prueba			
	Accuracy	Precisión	Recall	F1	Accuracy	Precisión	Recall	F1
AIBERT (base, v2)	0.965	0.965	0.965	0.965	0.965	0.965	0.965	0.965
BERT (base, uncased)	0.972	0.973	0.971	0.972	0.971	0.972	0.971	0.971
ELECTRA (base)	0.970	0.970	0.969	0.970	0.973	0.974	0.973	0.973
RoBERTa (base)	0.978	0.978	0.977	0.978	0.981	0.981	0.980	0.981
XLM-RoBERTa (base)	0.972	0.973	0.972	0.972	0.962	0.962	0.962	0.962
LR	0.891	0.891	0.890	0.890	0.886	0.891	0.890	0.890

Tabla 4.1. Resultados de evaluación con los subconjuntos de validación y prueba con los diferentes modelos de clasificación y las métricas de desempeño

Para el conjunto de validación, el modelo RoBERTa obtuvo los mejores resultados con una accuracy de 0.978, una precisión de 0.978, un recall de 0.977 y una F1 de 0.978. Le siguieron de cerca los modelos BERT y XLM-RoBERTa con valores similares aproximados de 0.972 para todas las métricas.

En el conjunto de prueba, nuevamente el modelo RoBERTa demostró el mejor desempeño con una accuracy de 0.981, una precisión de 0.981, un recall de 0.980 y una F1 de 0.981. En este caso el siguiente mejor modelo fue ELECTRA, con valores aproximados de 0.973 para todas las métricas.

El modelo de regresión logística (LR) mostró el menor desempeño tanto en el conjunto de validación como en el de prueba, con valores para sus métricas entre un 7% y 9% inferiores a los modelos de transformadores.

Los resultados indican claramente que los modelos basados en transformadores, especialmente RoBERTa, son altamente efectivos para la tarea de detección de noticias falsas, superando significativamente al modelo más tradicional LR.

En la Figura 4.1 se muestran las matrices de confusión para los subconjuntos de validación y prueba. En ambas matrices se observa que la mayor cantidad de errores que comete el modelo es al clasificar noticias que son falsas (fake) como reales.

		Validación		Prueba	
		Predicha		Predicha	
		Real	Fake	Real	Fake
Verdadera	Real	1110	10	1108	12
	Fake	38	982	29	991

Figura 4.1. Matrices de confusión para los subconjuntos de validación y prueba con el modelo RoBERTa

La Tabla 4.2 muestra algunos ejemplos de noticias en el conjunto de pruebas que son falsas, pero que el modelo RoBERTa clasifica como reales. El contenido de las noticias contiene palabras y oraciones que tiene sentido semántico y que es complicado de determinar si efectivamente son falsas. El modelo debe confundir el contexto de diversas palabras con aquellos que aparecen en noticias reales.

Noticias falsas mal clasificadas

*DNA Vaccine: injecting genetic material into the host so that host cells create proteins that are similar to those in the virus against which the host then creates antibodies

This year Election Night could be chaotic. In addition to coping with an unprecedented number of mail ballots COVID requires election officials to provide clean socially distanced in-person voting and grapple with a likely.

Tabla 4.2. Ejemplos de noticias falsas clasificadas como reales por el modelo RoBERTa

5. Conclusiones

En este trabajo se presentó un estudio sobre la detección automática de noticias falsas en Internet utilizando técnicas de deep learning. El estudio se dividió en dos fases: en la primera fase se empleó un modelo clásico de aprendizaje de máquina, la regresión logística; mientras que en la segunda fase se utilizaron modelos de transformadores, como ALBERT, BERT, ELECTRA, RoBERTa y XLM-RoBERTa. Para experimentar con estos modelos, se utilizó un conjunto de 10,700 publicaciones recolectadas de diversas redes sociales, que son noticias en inglés sobre el COVID-19. El conjunto estaba dividido en 5,600 noticias reales y 5,100 noticias falsas.

De acuerdo con los resultados de los experimentos, es posible concluir lo siguiente:

1. Los modelos de transformadores superaron en desempeño al modelo de regresión logística, indicando que las arquitecturas de atención son más adecuadas para esta tarea que la representación en vectores de palabras.
2. El modelo que produce un mejor desempeño en general para la detección de noticias falsas es el modelo RoBERTa. Otros modelos con buen desempeño son BERT, XLM-RoBERTa y ELECTRA.
3. El mejor modelo suele equivocarse más al clasificar noticias falsas como reales, debido quizá a la imposibilidad de distinguir de forma correcta el contexto de las palabras.

Algunas ideas para trabajos futuros incluyen la exploración de otras arquitecturas de redes neuronales de deep learning para representar las noticias, tales como GPT y T5; así como el uso de otros modelos de aprendizaje de máquina para la clasificación, como máquinas de vectores de soporte no lineales y ensamblajes de clasificadores. También se puede considerar usar más datos para construir los modelos de clasificación, lo que se espera mejore su desempeño. Finalmente, la combinación de características textuales y contextuales, así como la integración de datos multimodales (texto, imágenes, vídeos), es otra área para explorar en este problema.

Bibliografía/Referencias

- [1] Mykhailo Granik and Volodymyr Mesyura. Fake news detection using naive Bayes classifier. 2017 IEEE 1st Ukraine Conference on Electrical and Computer Engineering, UKRCON 2017 - Proceedings, pages 900-903, 2017.
- [2] Karishnu Poddar, Geraldine Bessie Amali D., and K.S. Umadevi. Comparison of Various Machine Learning Models for Accurate Detection of Fake News. In 2019 Innovations in Power and Advanced Computing Technologies (i-PACT), pages 1-5. IEEE, 3 2019.
- [3] Vasu Agarwal, H. Parveen Sultana, Srijan Malhotra, and Amitrajit Sarkar. Analysis of Classifiers for Fake News Detection. *Procedia Computer Science*, 165:377-383, 2019.
- [4] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. In Issa Traore, Isaac Woungang, and Ahmed Awad, editors, *First International Conference on Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments*, volume 10618 of *Lecture Notes in Computer Science*, pages 127-138. Springer International Publishing, Cham, 2017.
- [5] Pritika Bahad, Preeti Saxena, and Raj Kamal. Fake News Detection using Bi-directional LSTM-Recurrent Neural Network. *Procedia Computer Science*, 165:74-82, 2019.
- [6] Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, and Soumendu Sinha. FNDNet - A deep convolutional neural network for fake news detection. *Cognitive Systems Research*, 61:32-44, 6 2020.
- [7] Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 849-857, New York, NY, USA, 7 2018. ACM.
- [8] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1-14, 2015.
- [9] Mario Ezra Aragon, Horacio Jarqun-Vasquez, Manuel Montes-Y-Gomez, Hugo Jair Escalante, Luis Villaseñor-Pineda, Helena Gomez-Adorno, Juan Pablo Posadas-Duran, and Gemma Bel-Enguix. Overview of mex-a3t at iberlef 2020: Fake news and aggressiveness analysis in mexican Spanish. *CEUR Workshop Proceedings*, 2664(September):222-235, 2020.
- [10] Esau Villatoro-Tello, Gabriela Ramirez-De-La-Rosa, Sajit Kumar, Shantipriya Parida, and Petr Motticek. Idiap and UAM Participation at MEX-A3T Evaluation Campaign. *CEUR Workshop Proceedings*, 2664(January):252-257, 2020.
- [11] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 1(Mlm):4171-4186, 2019.
- [12] Diego Zazar-Gutierrez, Daniel Fajardo-Delgado, and Miguel A. Alvarez-Carmona. ITCG's participation at MEX-A3t 2020: Aggressive identification and fake news detection based on textual features for mexican Spanish. *CEUR Workshop Proceedings*, 2664:258-264, 2020.
- [13] Samuel Arce-Cardenas, Daniel Fajardo-Delgado, and Miguel Alvarez-Carmona. TecNM at MEX-A3t 2020: Fake news and aggressiveness analysis in mexican Spanish. *CEUR Workshop Proceedings*, 2664:265-272, 2020.