Text Mining

UNIVERSIDAD DE GRANADA

Asignatura: Tratamiento Inteligente de Datos

Índice

1. Introducción

Esta práctica trata sobre la clasificación de noticias como falsas o reales mediante técnicas de procesamiento de lenguaje natural (NLP) y minería de texto. El objetivo principal consiste en desarrollar un workflow completo que permita identificar automáticamente la veracidad de un artículo a partir de su contenido textual, facilitando la detección de desinformación en medios digitales.

Se utiliza el ISOT Fake News Dataset, compuesto por dos conjuntos de datos: uno de noticias reales obtenidas de Reuters y otro de noticias falsas extraídas de sitios web identificados como no confiables. Cada registro incluye el título, el texto completo, el tema de la noticia y la fecha de publicación. El dataset contiene aproximadamente 75.837 noticias, con 36.326 noticias reales y 40.773 noticias falsas.

En el análisis inicial consideramos la distribución de las clases y las características básicas del texto. La longitud de las noticias varía significativamente, con un promedio de 404 palabras y un rango que se extiende desde artículos bastante breves hasta noticias con más de 8.000 palabras. Por eso hace falta hacer un preprocesamiento textual que normalice los datos y permita extraer características relevantes para la clasificación.

Hemos usado técnicas de preprocesamiento de texto, incluyendo limpieza de URLs, puntuación, conversión a minúsculas, eliminación de palabras vacías y lematización. También se combina el título y el contenido de cada noticia para maximizar la información disponible para los modelos de aprendizaje automático.

La estructura del workflow considera dos enfoques principales de representación textual: TF-IDF, que transforma el texto en vectores basados en la frecuencia relativa de términos, y embeddings generados mediante modelos de transformers, que capturan la semántica del contenido. Ambos enfoques permiten entrenar clasificadores para distinguir entre noticias falsas y reales, evaluando su rendimiento con el uso de métricas de precisión, recall y F1-score.

El enlace al código es el siguiente:
https://colab.research.google.com/drive/19jZxcXnVU4MczSGIRsA4fCpEPsm957fX?usp=sharing

2. Análisis exploratorio de los datos (EDA)

El análisis exploratorio permite entender la estructura y las características del dataset antes de aplicar técnicas de aprendizaje automático. Se examinan la distribución de las clases, la longitud de los textos y otras estadísticas descriptivas relevantes.

2.1 Distribución de las clases

La variable objetivo *label* indica si una noticia es falsa (0) o real (1). La inspección inicial de la distribución muestra que las noticias falsas representan aproximadamente el 53,7 % del dataset, mientras que las noticias reales representan el 46,3 %. Esta distribución relativamente balanceada permite entrenar modelos de clasificación sin requerir técnicas de re-muestreo o balanceo adicional.

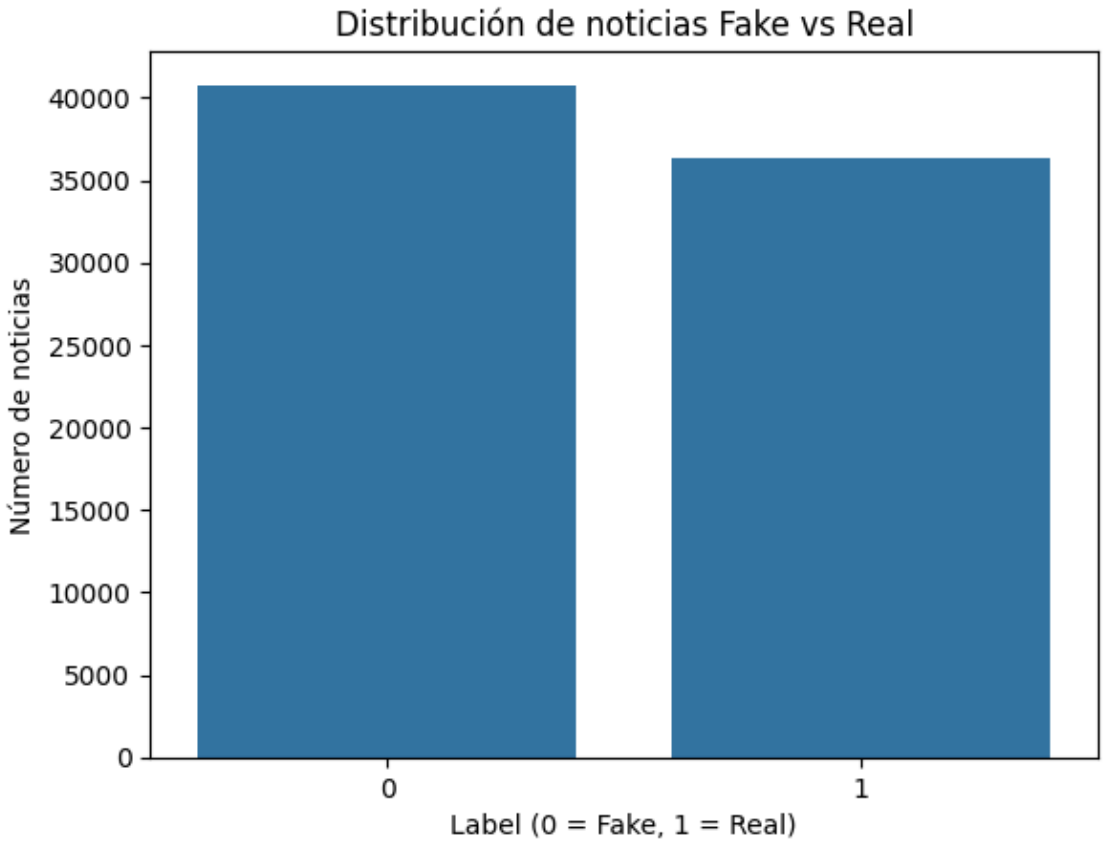


Imagen 1: Distribución de noticias

2.2 Estadísticas descriptivas del texto

Se analizan la longitud de los artículos y el número de palabras para identificar variabilidad, outliers y patrones relevantes.

- La longitud de los textos varía entre 1 y 8.135 palabras
- La media de palabras por noticia es aproximadamente 404, con desviación estándar de 367, indicando heterogeneidad en la extensión de los artículos
- El análisis percentil muestra que el 25 % de las noticias contienen menos de 191 palabras, el 50 % menos de 354 palabras y el 75 % menos de 512 palabras.

	text_length	word_count
count	77099.000000	77099.000000
mean	2460.158290	403.989325
std	2270.304599	367.384662
min	1.000000	0.000000
25%	1158.000000	191.000000
50%	2138.000000	354.000000
75%	3099.000000	512.000000
max	51794.000000	8135.000000

Imagen 2: Estadísticas descriptivas de la longitud y número de palabras

2.3 Distribución de la longitud de noticias

La visualización de la longitud de noticias permite identificar noticias extremadamente cortas o excesivamente largas que pueden influir en el entrenamiento de modelos. Se observa que la mayoría de los artículos se concentra entre 200 y 1.000 palabras, con una cola larga correspondiente a artículos muy extensos.

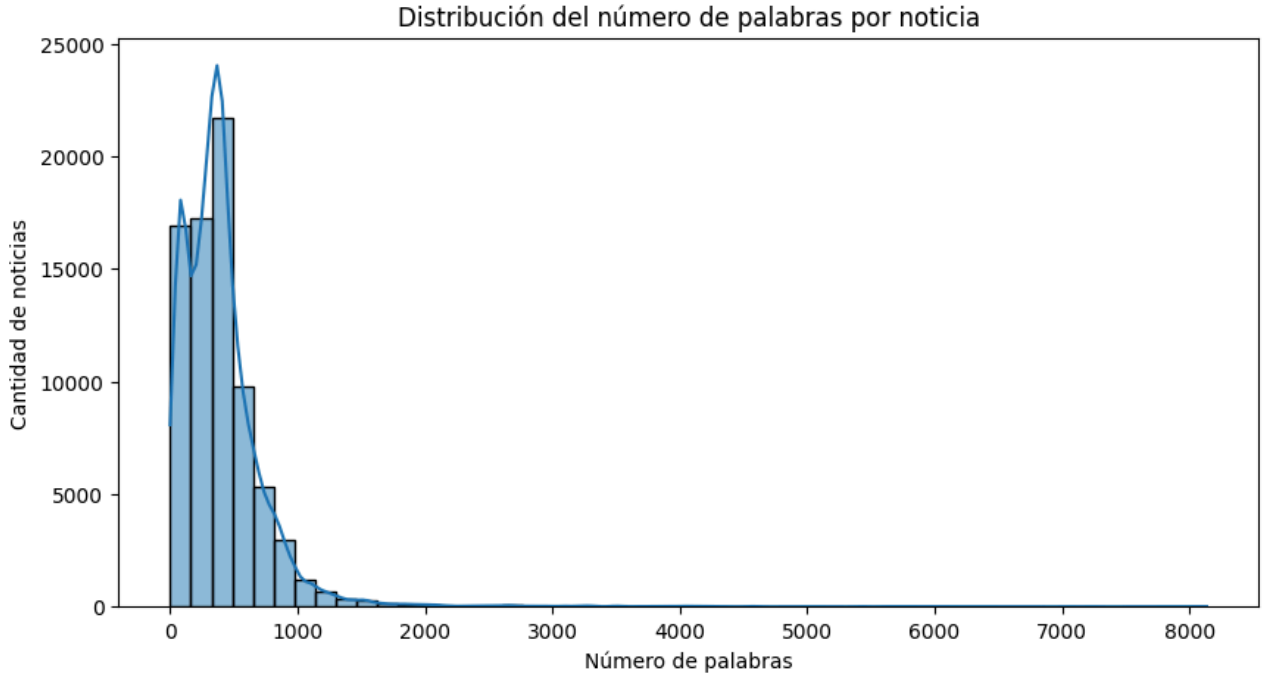


Imagen 3: Distribución del número de palabras por noticia

2.4 Conclusiones del EDA

El análisis muestra que el dataset tiene:

- Una distribución de clases relativamente equilibrada
- Noticias con longitudes variadas, lo que requiere normalización y preprocesamiento textual antes de entrenar los modelos
- Patrones de variabilidad que justifican la aplicación de técnicas de limpieza y transformación de texto, como lematización, eliminación de palabras vacías y combinación de título y contenido.

Estos hallazgos guían las decisiones en la fase de preprocesamiento y selección de características, asegurando que los modelos de clasificación reciban información homogénea y representativa de las noticias.

3. Preprocesamiento

El preprocesamiento de texto transforma los artículos en un formato adecuado para que los modelos de aprendizaje automático puedan analizarlos. Este paso combina limpieza, normalización y representación numérica de los textos.

3.1 Limpieza y normalización

Se hace una limpieza sistemática de los textos para eliminar elementos que no aportan información relevante:

1. Conversión a minúsculas para uniformizar todas las palabras
2. Eliminación de URLs mediante expresiones regulares, evitando que enlaces influyan en los modelos
3. Eliminación de puntuación y números, manteniendo únicamente caracteres alfabéticos y espacios
4. Tokenización y eliminación de stopwords, utilizando el corpus de NLTK
5. Lematización, transformando las palabras a su forma base para reducir la dimensionalidad y agrupar variantes léxicas.

Se combina el título y el contenido de cada noticia en un campo full_text para maximizar la información disponible.

	full_text	clean_text
0	Trump announces new hires to expand campaign operations (Reuters) - Presumptive Republican nominee D...	trump announces new hire expand campaign operation reuters presumptive republican nominee donald tru...
1	HAWAIIAN RESTAURANT Gets Hammered With Negative Reviews After BANNING Trump Supporters: “You cannot ...	hawaiian restaurant get hammered negative review banning trump supporter cannot eat honolulu caf get...
2	Cuba’s Raul Castro meets with U.S. Chamber of Commerce president HAVANA (Reuters) - The head of the	cuba raul castro meet u chamber commerce president havana reuters head u chamber commerce met cuban .
3	VA chief presses Congress to make it easier to fire workers for misconduct WASHINGTON (Reuters) - Ve...	va chief press congress make easier fire worker misconduct washington reuters veteran affair secreta...
4	Senator Grassley expresses reservations on two Trump judge nominees WASHINGTON (Reuters) - The Repub...	senator grassley express reservation two trump judge nominee washington reuters republican chairman ...

Tabla 1: Ejemplo resumido de textos originales y limpios

	full_text_len	clean_text_len
0	78	56
1	579	300
2	405	247
3	395	232
4	323	195

Imagen 4: Longitud de textos originales y limpios (número de palabras)

Estos ejemplos nos permiten observar cómo la limpieza y lematización simplifica y estandariza los textos.

3.2 Representación numérica: TF-IDF

Para convertir los textos en datos que los modelos puedan procesar, se utiliza TF-IDF (Term Frequency-Inverse Document Frequency). Este método asigna un peso a cada término según su frecuencia en un documento y su rareza en el corpus completo.

- Se limita el vocabulario a 10.000 términos más relevantes
- Cada noticia se representa como un vector de características numéricas de dimensión 10.000

```
tfidf = TfidfVectorizer(max_features=10000)
X_tfidf = tfidf.fit_transform(df['clean_text'])

y = df['label']

print("Tamaño TF-IDF:", X_tfidf.shape)

Tamaño TF-IDF: (75837, 10000)
```

Imagen 5: Dimensión de la matriz TF-IDF

3.3 Representación numérica: embeddings semánticos

Se utiliza el modelo all-MiniLM-L12-v2 de Sentence Transformers para generar embeddings que capturan la semántica del texto:

- Cada noticia se transforma en un vector de alta dimensión que refleja significado y contexto
- Esta representación permite que los modelos distingan patrones de significado más allá de coincidencias exactas de palabras.

3.4 Conclusiones del preprocesamiento

La combinación de limpieza, tokenización, eliminación de stopwords y lematización permite reducir el ruido presente en los textos y aumentar su representatividad. Estas técnicas nos sirven para que los modelos de clasificación reciban información homogénea, centrada en las palabras más relevantes de cada noticia.

El uso de TF-IDF nos sirve para capturar patrones de frecuencia de palabras clave que diferencian noticias falsas de reales, mientras que los embeddings proporcionan una representación más rica y semántica, capturando relaciones de significado entre las palabras y el contexto de las noticias.

4. Clasificación y resultados

El objetivo de esta fase consiste en entrenar modelos capaces de distinguir noticias falsas de reales, utilizando las representaciones numéricas generadas en el preprocesamiento. Se implementan dos enfoques principales: TF-IDF y embeddings semánticos.

4.1 Clasificación con TF-IDF

Se entrena un clasificador de Logistic Regression utilizando los vectores TF-IDF como entrada. La división de los datos considera 80 % para entrenamiento y 20 % para prueba, manteniendo la proporción de noticias falsas y reales mediante la opción stratify. Este enfoque permite que el modelo aprenda patrones de frecuencia de palabras que distinguen las noticias falsas de las reales, aprovechando las diferencias de vocabulario entre clases.

El modelo alcanza una accuracy de 0,9917, con métricas de precisión, recall y F1-score muy equilibradas entre las dos clases. La evaluación detallada mediante el classification report muestra valores cercanos a 0,99 para ambas categorías, lo que confirma la alta fiabilidad del modelo sobre este dataset.

Classification Report:

	precision	recall	f1-score	support
0	0.99	0.99	0.99	7903
1	0.99	0.99	0.99	7265
accuracy			0.99	15168
macro avg	0.99	0.99	0.99	15168
weighted avg	0.99	0.99	0.99	15168

Imagen 6: Classification report TF-IDF

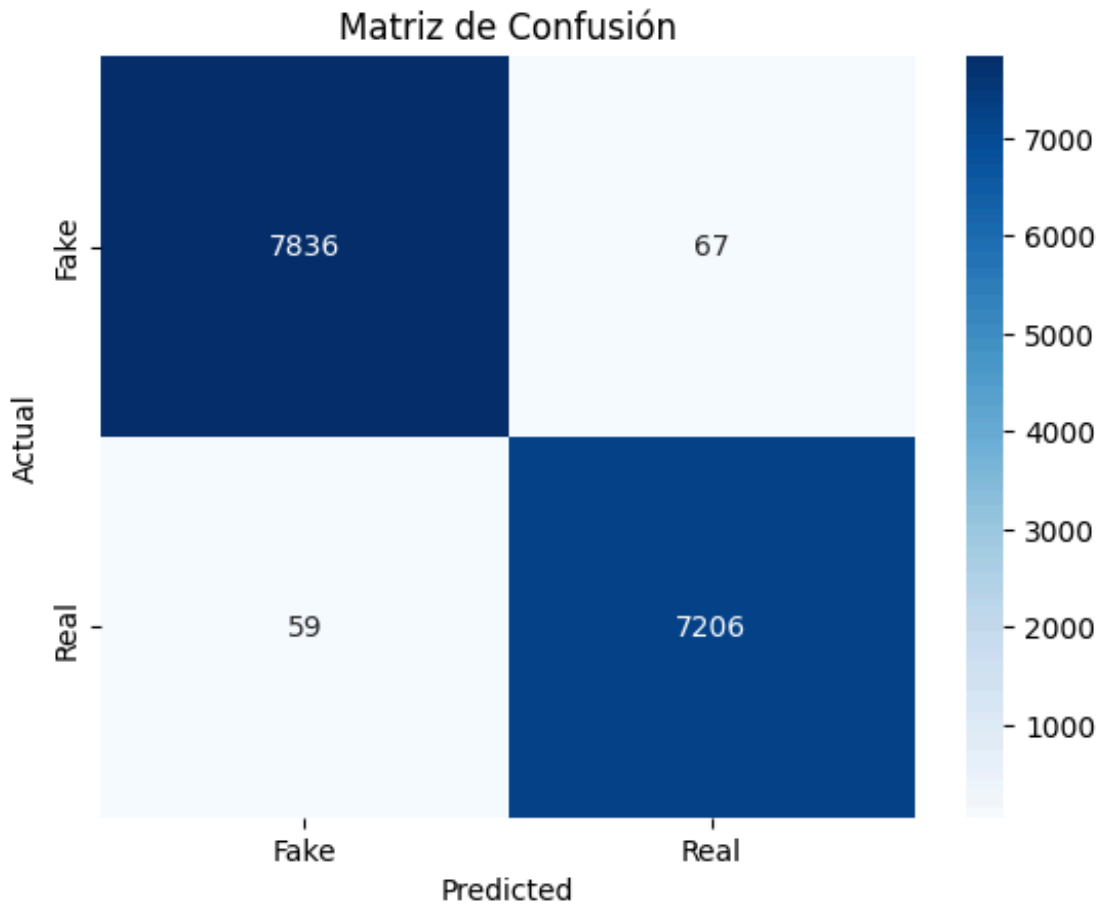


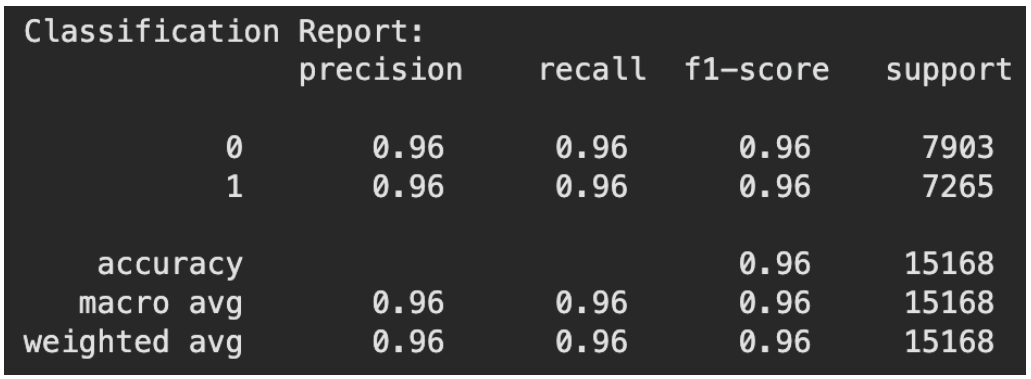
Imagen 7: Matriz de confusión TF-IDF

La validación cruzada con 5 folds ofrece resultados consistentes, con accuracies de [0,9921, 0,9910, 0,9912, 0,9912, 0,9916] y media de 0,9914. Esto confirma que el modelo generaliza correctamente y que los resultados no dependen de una partición específica del dataset.

4.2 Clasificación con embeddings

Se repite el proceso de clasificación utilizando embeddings generados por Sentence Transformers. Cada noticia se representa mediante un vector que captura su significado y contexto, permitiendo que el modelo reconozca similitudes semánticas incluso cuando las palabras exactas no coinciden. Logistic Regression se emplea de nuevo como clasificador, entrenando sobre los embeddings generados a partir de los textos limpios.

El modelo alcanza una accuracy de 0,9597, con métricas de precisión, recall y F1-score alrededor de 0,96 para ambas clases. La validación cruzada de 5 folds confirma estos resultados, con valores de [0,9610, 0,9591, 0,9626, 0,9629, 0,9602] y media de 0,9611. Esto demuestra que los embeddings son consistentes y robustos, aunque en este dataset concreto presentan un rendimiento ligeramente inferior al de TF-IDF.



```
Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.96      0.96      7903
           1       0.96      0.96      0.96      7265

    accuracy                           0.96     15168
   macro avg       0.96      0.96      0.96     15168
weighted avg       0.96      0.96      0.96     15168
```

Imagen 8: Classification report embeddings

4.3 Comparación de resultados

La comparación entre ambos métodos nos muestra que TF-IDF ofrece un rendimiento superior en el dataset ISOT, debido a que las diferencias de vocabulario entre noticias reales y falsas son pronunciadas. Los embeddings, por su parte, capturan la semántica y proporcionan una representación más rica de los textos, lo que los hace útiles en escenarios donde las palabras clave no bastan para diferenciar clases o cuando se aplican modelos a nuevos datasets más variados.

4.4 Conclusiones de la clasificación

El workflow implementado permite detectar noticias falsas con alta fiabilidad, demostrando que las técnicas de preprocesamiento y representación textual que hemos aplicado son adecuadas para esta tarea. El uso de TF-IDF es bastante efectivo cuando los textos tienen diferencias claras de vocabulario entre las clases, ya que captura patrones de frecuencia de palabras que distinguen noticias reales de falsas. Esto explica el rendimiento elevado que se observa en el dataset ISOT, con métricas de precisión, recall y F1-score cercanas a 0,99.

Por otra parte, los embeddings ofrecen una representación más rica y semántica de los textos, permitiendo que los modelos reconozcan similitudes de significado incluso cuando las palabras exactas no coinciden. Esta característica hace que los embeddings sean especialmente útiles en escenarios donde las palabras clave no bastan para diferenciar las clases, proporcionando mayor generalización a textos nuevos o a datasets distintos del empleado en esta práctica.